OFF LINE

V. 10 - 11
1982 - 1983 (present)

Missing Pieces:

Vol.	No.	Date
V.	1	April 24
II	1-2	[illegible]

Missing items not ordered because availability or importance is doubtful

Comments:

Date:
L206

Computing Services Office

Off-Line

University of Illinois at Urbana-Champaign VOL. 10, NO. 1 January 1982

EDITOR: Lynn Bilger
PHONE: (217) 333-6236
150 Digital Computer Lab
1304 W. Springfield Ave.
Urbana, Illinois 61801

Page *Contents*

CSO DIRECTORY - STAFF AND SERVICES

Administrative

Director	George Badger	150	DCL	333-4103
Business Manager	Stanley Rankin	150	DCL	333-6530
Secretary	Joyce McCabe	150	DCL	333-1637

User Services and Hardware/Software Support

User Accounting		**1208**	**W Springfield**	**333-7752**
Distribution Office		**1208**	**W Springfield**	**333-6760**
Systems Consulting		**1208**	**W Springfield**	**333-6133**
Statistical Services Consulting		**65**	**Comm West**	**333-2170**
Text Processing Consulting		**207**	**Astronomy**	**333-7318**
Terminal Repair Service		**150**	**DCL**	**333-0969**
CYBER Dial-up Numbers	**CYBER 175**	**110-300**	**baud**	**333-4000**
	CYBER 175	**1200**	**baud**	**333-4001**
	CYBER 174	**110-300**	**baud**	**333-4004**
Asst Dir User Services	Robert Penka	173	DCL	333-4709
Asst Dir Systems and Operations	Sandra Moy	177	DCL	333-4703
Asst Dir Development	J. M. Randal	120	DCL	333-9772
Asst Dir Engineering and Hardware Selection	Cliff Carter	195	DCL	333-3723
Manager Statistical Services	Beth Richardson	91	Comm West	333-2170
Documentation	Lynn Bilger	139	Astronomy	333-6236
CYBER-IBM Operations	Jack Knott	194a	DCL	333-6562
UNIX Operations	Debbie Weller	123	DCL	333-8150
Telecommunications	Tom Kerkering	164	DCL	333-0816
Laboratory Support Project	Mike Gardner	164	DCL	333-7904
RJE Operations North	Rex Duzan	162	DCL	333-6285
RJE Operations South	Don McCabe	1208	W Springfield	333-2171 333-7752

RJE Sites (see Reference Guide RF-0.3 for operating hours)

Agriculture	M103	Turner Hall	333-8170
Chemistry	153	Noyes Lab	333-1728
Commerce West	70	Comm West	333-4500
CRH Snack Bar	120	Snack Bar	333-1851
DCL Routing Room	14	DCL	333-6203
Electrical Engineering	146	EEB	333-4936
Florida Ave Res Hall		FAR	333-2695
Illinois St Res Hall		ISR	333-0307
Mechanical Engineering	65	MEB	333-1430
Psychology	453	Psych Bldg.	333-7531
Social Science	202	Lincoln Hall	333-0309

OFF-LINE is the monthly newsletter of the Computing Services Office at the University of Illinois at Urbana-Champaign. Unless otherwise indicated, permission to reprint is freely granted, provided that the author, if named, and the Computing Services Office (CSO) are credited. Information in this issue is current as of December 24, 1981.

CSO operates a CDC CYBER 175 with 262K words of central memory and a CDC CYBER 174 with 196K words of central memory. The 175 and 174 run under the NOS Operating System and share 512K words of ECS. The 175 serves over 200 simultaneously active text and graphics terminals and the 174 serves over 100 simultaneously active terminals. CSO also operates an IBM 4341 with 4 million bytes of memory running HASP-OS/MVT under VM. In addition, CSO operates a DEC PDP-11/50 with 252K bytes of core driving a GSI CAT-8 phototypesetter and a DEC VAX 11/780 with 2 million bytes of core, both running under the UNIX Operating System.

Operating Hours (see HEARYE or SCHEDUL for exceptions):

	CYBERS 174/175	IBM
M-F	8 am to 6 am	8 am to 6 am
SAT	8 AM - Midnight	8 AM - 6 AM
SUN	Noon - 6 AM	Noon - 6 AM

POLICY

TREATMENT OF OVERHEAD CHARGES

As of January 1, 1982, all sponsored research accounts operating under the new overhead structure of "modified total direct cost," will be handled and flagged automatically through the UFAS accounting system. Charges made on a ledger 5 account (sponsored projects) will have an additional charge of 8.3% added by the UFAS accounting system against an object code of 5569. Accordingly, 5569 is no longer a legitimate object code which can be used by departments; it is restricted to use by the UFAS accounting system. This additional charge is for overhead charges associated with computing services. Computing service is not included in the direct cost to which the 42.4% overhead is applied.

Questions concerning this matter should be directed to the contracts office.

SYSTEM NOTES

NOVEMBER RELIABILITY REPORTS

CYBER 175

10 recoverable interruptions
13 non-recoverable interruptions
13 of these were for more than 15 minutes

MTBF = 30.6 hours
MTR = 42 minutes
Availability: 97.6% of scheduled uptime

Major cause of downtime was related to ECS errors.

CYBER 174

15 recoverable interruption
8 non-recoverable interruptions
8 of these were for more than 15 minutes

MTBF = 30.3 hours
MTR = 60 minutes
Availability: 96.4% of scheduled uptime

Major cause of downtime was related to ECS errors and communications equipment.

IBM 4341

26 interruptions
7 of these were for more than 15 minutes

MTBF = 27.3 hours
MTR = 26 minutes
Availability: 98.5% of scheduled uptime

Major cause of downtime related to disk problems.

CYBER

NEW USER-CONTRIBUTED SOFTWARE LIBRARY

CSO is now sponsoring a library of user-contributed software. The purpose of the library is to foster sharing of software between CYBER users. CSO is providing the disk space, the tools used to access the offerings in the library, and is making all library updates, but is not otherwise involved. This is a user-to-user exchange.

Updates to the library will be made on a periodic basis. CSO will decide which programs are entered into the library using the following guidelines:

- The program or package must be generally useful and not of interest primarily to a single class or research group. Games will not generally be accepted for inclusion in the library.

- The contributor of the software must provide a machine-readable description of it sufficient to allow a potential user to judge its usefulness and, hopefully, begin using it. Ideally, complete documentation should be in machine-readable form. If complete documentation is available only in printed form, a copy must be supplied for storage at the Systems Consulting Office (1208 W. Springfield) where it will be made available for inspection.

- The user must state the level of support given to the package. At a minimum, the contributor must agree to make any modifications required by installation of a new version of the operating system, must give his/her name, phone number and address, and must be willing to talk to users encountering difficulties using the program. The contributor's name, phone number and address will be made available to users of the program.

- The program must be stable and not undergoing continuing development. Repeated updates will not be made to an existing program in the library.

- Copies of all files used must be provided to CSO for inclusion in the library.

CSO will make broad judgements about the usefulness of the software, but will not make any tests for reliability or usability. As a consequence, CSO will not provide software maintenance or consulting help for the contents of the library. These functions remain the contributor's responsibility.

To list a directory of the library, enter the command:

```
SHARED.
```

The output generated will contain one line for each product in the library. Each line will contain a product name followed by a one-line description.

To obtain a description and on-line documentation for a specific product in the library, enter the command:

SHARED,*product name*/WRITEUP.

This command will produce a local file or files containing on-line documentation for the product named. The file and its attributes (e.g., ascii, carriage control) will be identified in the output produced by the command. Finally, to use one of the programs or packages in the library, enter the command:

SHARED,*product name*.

Then follow the instructions given in the on-line documentation.

Programs will be removed from the library if they are not being used enough to warrant their inclusion or if the contributor has stopped providing support. Advance warning will be given before any deletions are made. If you plan to use a program in the library on an ongoing basis, we recommend that you obtain a source copy from its contributor as a precaution against its eventual removal from the library. CSO will not maintain source copies of programs available in the library for archival or distribution purposes.

If you have software which you would like to contribute to the SHARED public library, please contact Kurt Hirchert, 173 DCL (333-6264).

REMINDER TO CLAIM PERMANENT FILES

At the end of each semester, CYBER permanent files potentially suffer a number of problems. If you expect your files to exist from semester to semester, you must keep a few things in mind:

- When you create a permanent file, the charge,project combination you are using at that time generates a unique project index (and an associated project index hash) which is stored with the file for accounting purposes. The file is said to "belong" to the project.

- Many charge,project numbers expire or run out of funds at the end of a semester.

A file which belongs to a charge,project which has expired or run out of funds must be claimed to an active account within three months or the file will be purged. Once a file is purged, we can only guarantee a tape backup of the file for an additional three months. Be sure to claim any files belonging to ps numbers which have expired or are out of funds to an active account.

The single command

DIRECT/CLAIM/PROJECT=*xxxx*.

provides a convenient way to claim all files belonging to an expired project. *xxxx* is the project

index hash which uniquely identifies the expired project. The command assigns all files belonging to the expired project to the project specified on the latest BILL or CHARGE command.

To use this command, you must first know the project index hash of your expired project. If you do not know it, you may obtain it by entering the command:

```
DIRECT/PROJECT/SUM.
```

The generated output will end with a series of lines of the form

xxxx CHARGED TO (*DEPT,PSnnn*) 150 FILES, 4498 INDIRECT PRUS.

where *xxxx* is the four-character project index hash and *(DEPT,PSnnnn)* is the project which it uniquely identifies. One line of this form will appear for every project to which at least one of your files belongs.

STATISTICAL SERVICES

FOSOL

The FOSOL statistical package is being considered for removal from the CYBER systems. The reasons for removal are lack of usage and lack of vendor support. If strong, convincing arguments for the support of FOSOL are not forthcoming, the package will be removed from the system on February 12, 1982. Comments on this matter should be directed to Beth Richardson, Box 86 Comm West (3-2171).

FEATURE ARTICLES

FORTRAN FUTURES

This article is part of a series on the work of the FORTRAN Standards Committee X3J3 in producing the next revision of the FORTRAN standard. The reader is reminded that the features described in this article are not a part of any FORTRAN compiler currently available, but rather are proposed requirements for FORTRAN processors produced in the late 1980's through the mid 1990's. Although every effort has been made to accurately describe the current position of X3J3 on these matters, the development of a revision to the FORTRAN standard is an evolutionary process, and these proposals may be subject to refinement, revision, or even retraction. Comments on these proposals may be given to Kurt Hirchert of the CSO Systems Consulting staff, who is a member of X3J3.

Traditionally, each subroutine and function in a FORTRAN program is equally accessible to all other subroutines and functions in that program. The only exception to this global

To obtain a description and on-line documentation for a specific product in the library, enter the command:

SHARED,*product name*/WRITEUP.

This command will produce a local file or files containing on-line documentation for the product named. The file and its attributes (e.g., ascii, carriage control) will be identified in the output produced by the command. Finally, to use one of the programs or packages in the library, enter the command:

SHARED,*product name*.

Then follow the instructions given in the on-line documentation.

Programs will be removed from the library if they are not being used enough to warrant their inclusion or if the contributor has stopped providing support. Advance warning will be given before any deletions are made. If you plan to use a program in the library on an ongoing basis, we recommend that you obtain a source copy from its contributor as a precaution against its eventual removal from the library. CSO will not maintain source copies of programs available in the library for archival or distribution purposes.

If you have software which you would like to contribute to the SHARED public library, please contact Kurt Hirchert, 173 DCL (333-6264).

REMINDER TO CLAIM PERMANENT FILES

At the end of each semester, CYBER permanent files potentially suffer a number of problems. If you expect your files to exist from semester to semester, you must keep a few things in mind:

- When you create a permanent file, the charge,project combination you are using at that time generates a unique project index (and an associated project index hash) which is stored with the file for accounting purposes. The file is said to "belong" to the project.

- Many charge,project numbers expire or run out of funds at the end of a semester.

A file which belongs to a charge,project which has expired or run out of funds must be claimed to an active account within three months or the file will be purged. Once a file is purged, we can only guarantee a tape backup of the file for an additional three months. Be sure to claim any files belonging to ps numbers which have expired or are out of funds to an active account.

The single command

DIRECT/CLAIM/PROJECT=*xxxx*.

provides a convenient way to claim all files belonging to an expired project. *xxxx* is the project

Although X3J3 has been working on the internal procedure proposals since the beginning of the current revision effort, there remain a number of unresolved issues. Should an internal procedure be allowed to have internal procedures of its own nested within it? Should it be permissible to pass an internal procedure to a subprogram such as an integration subroutine which expects a procedure dummy argument? Should some kind of shorthand notation be defined to provide an internal procedure access to all of the entities known to its host?

A topic which has proved closely related to the FORTRAN internal procedures facility is the development of an alternative to the use of COMMON for sharing data between procedures. Using COMMON normally requires either the tedious duplications of the specifications of each item in the common block in every procedure that references it or careful "mapping" of the common in each referencing procedure to insure that the items accessed are properly matched in type, length, and position. Furthermore, COMMON can be used to share only variables and arrays. On the other hand, in situations where the INHERIT statement is applicable, it provides access to a wide variety of FORTRAN entities simply by giving the names of the specific entities desired. It should come as no surprise that an extended form of the INHERIT statement has been adopted as a means of sharing data between procedures.

Again, a simple example will help illustrate the features of this facility:

```
SHARED DATA PLOTDEFS
PARAMETER (NUMCOORDS=2)
FORM COORDINATES
 REAL VALUE(NUMCOORDS)
END FORM
END
 .
 .
 .
SHARED DATA PLOTLOCS
INHERIT /PLOTDEFS/ COORDINATES
COORDINATES VIRTUAL_POSITION,ACTUAL_POSITION
SAVE VIRTUAL_POSITION,ACTUAL_POSITION
END
 .
 .
 .
SUBROUTINE MOVE(C)
INHERIT /PLOTLOCS/ COORDINATES,VIRTUAL_POSITION
COORDINATES C
VIRTUAL_POSITION=C
END
SUBROUTINE PEN(C)
INHERIT /PLOTLOCS/ COORDINATES
INHERIT /PLOTLOCS/ ACTUAL_POSITION
INHERIT /PLOTLOCS/ VIRTUAL_POSITION
COORDINATES C,TEMP
IF (C.NE.VIRTUAL_POSITION) THEN
 IF (VIRTUAL_POSITION.NE.ACTUAL_POSITION) THEN
  TEMP.VALUE=VIRTUAL_POSITION.VALUE-ACTUAL_POSITION.VALUE
  CALL RELATIVE_MOVE(TEMP)
 END IF
```

```
 TEMP.VALUE=C.VALUE-VIRTUAL_POSITION.VALUE
 CALL RELATIVE_PEN(TEMP)
END IF
ACTUAL_POSITION=C
VIRTUAL_POSITION=C
END
```

The data to be shared is specified in a special kind of subprogram which must be compiled before any of the routines that reference it. The compiler must then "remember" the attributes of the entities contained in the shared data subprogram so they can be referenced later. The routines using these entities specify them in an extended INHERIT statement which includes the name of the shared data subprogram in which the entities were originally specified. Unlike COMMON, one shared data subprogram may INHERIT entities from a previous one. Also unlike COMMON, the specification SAVE can be applied separately to the individual entities in the shared data subprogram, rather than apply to the collection as a whole.

As with the internal procedured facility, there are a number of unresolved issues relating to the shared data facility. Should an internal procedure be allowed to inherit directly from a shared data program or must it first be inherited into the host and then inherited into the internal procedure from there? Should a shared data subprogram be allowed to contain and share internal procedures? Should there be internal shared data subprograms to provide another means of sharing between internal procedures? Should adjustable (i.e., dynamically allocated) entities be allowed in a shared data subprogram? Should the extended INHERIT statement be extended still further to allow access to entities in executable subprograms as well as the nonexecutable shared data subprogram? If two different shared data subprograms have entities with the same names, should there be some means of resolving this name conflict, allowing both entities to be inherited into the same procedure? If so, what approach to name conflict resolution should be taken? Should there be any means of limiting which procedures can inherit from a given shared data subprogram? Should access modes such as READONLY which have been proposed for use with argument association also apply to inheritance? All of these issues are likely to be considered at forthcoming X3J3 meetings.

In spite of the issues which remain unresolved in the internal procedure and data sharing proposals, they promise to greatly simplify the job of managing the complexity of large FORTRAN programs.

DOCUMENTATION

NEW AND REVISED CSO DOCUMENTATION

Revised Manual

An Index to Software on the CYBER Systems

This manual has been updated to reflect the new software added to the CYBER systems since the previous publication date of September 1980. An appendix containing brief descriptions of unsupported software has been added.

On-line Documentation

On-line documentation has been added for the following:

XEDIT - an extended interactive text editor from the University of Minnesota. This writeup may be obtained and printed by entering:

```
WRITEUP,XEDIT.
PRINT,XEDDOC/CC/EJ.
```

The HELP program files are currently undergoing major revision with new HELP files being added for all entries in the software index. The new files should be on the system by the time of publication (or shortly thereafter).

Reference Guides

RF-0.1 Reference Guide List (Revised 12/22/81)
RF-0.6 Disk Policy (Revised 12/18/81)
RF-1.2 CYBER Rates (Revised 12/18/81)
RF-4.20 FLEX (New 12/03/81)
RF-4.21 FORSIM (New 10/05/81)
RF-4.22 GMET (New 12/03/81)
RF-4.23 ITPACK (New 12/22/81)
RF-4.24 NETWORK (New 12/04/81)
RF-4.25 PORT (New 12/04/81)
RF-4.26 REDUCE (New 10/27/81)
RF-4.27 FISHPAK (New 10/27/81)
RF-5.4 Easy Graphing (New 12/16/81)
RF-5.5 NCAR Graphics Software (Revised 12/22/81)
RF-5.6 ZETAVU (New 12/16/81)
RF-7.8 FETCH (Revised 11/30/81)
RF-11.2 IBM Rates/Charge Estimator (Revised 12/18/81)

MISCELLANEOUS

COMPUTER-RELATED DISCOUNTS

A number of computer-related discounts, available through the University, were published in the September issue of *OFF-LINE.* The following information on Tektronix equipment should be added to that list.

Tektronix has grouped similar types of products for quantity discount. Products are not combined for discount across group lines. The product groups are:

Group 62	4006-1, 4010, 4010-1, 4012, 4013, 4014, 4014-1, 4015-1, 4016-1, 4112, 4114
Group 63	4023, 4024, 4025, 4027
Group 64	4051, 4052, 4054
Group 65	4081
Group 66	4611, 4612, 4631, 4632, 4634, 4641, 4641-1, 4642, 4642-1, 4662, 4663
Group 67	4905, 4907, 4923, 4924, 4952, 4953, 4954, 4956
Group 70	611, 611-2, 613, 613-1, 618

A product option will receive the same discount as the product if ordered with the product. Options not ordered with the product are part numbered items and are not discounted.

Information Display Division products with a catalog price of $125.00 or more may qualify for discount on single orders as follows:

Quantity Ordered Per Product Group	***Discount***
1 - 3 units	4.5%
4 - 9 units	9.5%
10 units	14.5%

The prices that the University will be charged will be the prices in effect at the time of order placement or the prices that are in effect on the delivery date, whichever is the lesser.

SALES - EXCHANGES - HELP WANTED

1/4 RESEARCH ASSISTANTSHIP

1/4 Research Assistantship, beginning 1/21/82. Should have knowledge of FORTRAN, SPSS, and SOUPAC. For more information, call 333-6371.

APPLE BASIC PROGRAMMER

Wanted. Temporary student programmer. Must know Apple Basic. Applicant should have completed CS 221. Contact:

R. G. Rossman
Room 134
Physical Plant Service Building
333-8380

TERMINAL TABLES FOR SALE

For sale: White terminal tables, 32" wide x 24" deep, black trim. Price=$125.00 each. Contact:

Harold Lopeman
333-0304

OFF-LINE's Mailing List

If you wish to be placed on our mailing list for future issues of *OFF-LINE*, if you wish to be removed from the list, or if you wish to enter an address correction, please complete and return this page. (Current subscribers are kept on the mailing list until a specific request for removal is received, or until a mailing is returned as undeliverable.)

Check one:
- ☐ New subscriber
- ☐ Removal request
- ☐ Address correction

Name: _____

Address: _____

_____

_____

CAMPUS or Zip Code _____

(If address correction, give old address and zip code below.)

Comments:

RETURN TO:

OFF-LINE
150 Digital Computer Laboratory
University of Illinois at Urbana-Champaign
1304 West Springfield Avenue
Urbana, Illinois 61801

Computing
Services
Office

Off-Line

University of Illinois at Urbana-Champaign VOL. 10, NO. 2 February 1982

EDITOR: Lynn Bilger
PHONE: (217) 333-6236
150 Digital Computer Lab
1304 W. Springfield Ave.
Urbana, Illinois 61801

CSO DIRECTORY - STAFF AND SERVICES

Administrative

Director	George Badger	150	DCL	333-4103
Business Manager	Stanley Rankin	150	DCL	333-6530
Secretary	Joyce McCabe	150	DCL	333-1637

User Services and Hardware/Software Support

User Accounting		**1208**	**W Springfield**	**333-7752**
Distribution Office		**1208**	**W Springfield**	**333-6760**
Systems Consulting		**1208**	**W Springfield**	**333-6133**
Statistical Services Consulting		**65**	**Comm West**	**333-2170**
Text Processing Consulting		**207**	**Astronomy**	**333-7318**
Terminal Repair Service		**150**	**DCL**	**333-0969**
CYBER Dial-up Numbers	**CYBER 175**	**110-300**	**baud**	**333-4000**
	CYBER 175	**1200**	**baud**	**333-4001**
	CYBER 174	**110-300**	**baud**	**333-4004**
Asst Dir User Services	Robert Penka	173	DCL	333-4709
Asst Dir Systems and Operations	Sandra Moy	177	DCL	333-4703
Asst Dir Development	J. M. Randal	120	DCL	333-9772
Asst Dir Engineering and Hardware Selection	Cliff Carter	195	DCL	333-3723
Manager Statistical Services	Beth Richardson	91	Comm West	333-2171
Documentation	Lynn Bilger	139	Astronomy	333-6236
CYBER-IBM Operations	Jack Knott	194a	DCL	333-6562
UNIX Operations	Debbie Weller	123	DCL	333-8150
Telecommunications	Tom Kerkering	164	DCL	333-0816
Laboratory Support Project	Mike Gardner	164	DCL	333-7904
RJE Operations North	Rex Duzan	162	DCL	333-6285
RJE Operations South	Don McCabe	1208	W Springfield	333-2171 333-7752

RJE Sites (see Reference Guide RF-0.3 for operating hours)

Agriculture	M103	Turner Hall	333-8170
Chemistry	153	Noyes Lab	333-1728
Commerce West	70	Comm West	333-4500
CRH Snack Bar	120	Snack Bar	333-1851
DCL Routing Room	14	DCL	333-6203
Electrical Engineering	146	EEB	333-4936
Florida Ave Res Hall		FAR	333-2695
Illinois St Res Hall		ISR	333-0307
Mechanical Engineering	65	MEB	333-1430
Psychology	453	Psych Bldg.	333-7531
Social Science	202	Lincoln Hall	333-0309

OFF-LINE is the monthly newsletter of the Computing Services Office at the University of Illinois at Urbana-Champaign. Unless otherwise indicated, permission to reprint is freely granted, provided that the author, if named, and the Computing Services Office (CSO) are credited. Information in this issue is current as of January 25, 1982.

CSO operates a CDC CYBER 175 with 262K words of central memory and a CDC CYBER 174 with 196K words of central memory. The 175 and 174 run under the NOS Operating System and share 512K words of ECS. The 175 serves over 200 simultaneously active text and graphics terminals and the 174 serves over 100 simultaneously active terminals. CSO also operates an IBM 4341 with 4 million bytes of memory running HASP-OS/MVT under VM. In addition, CSO operates a DEC PDP-11/50 with 252K bytes of core driving a GSI CAT-8 phototypesetter and a DEC VAX 11/780 with 2 million bytes of core, both running under the UNIX Operating System.

Operating Hours (see HEARYE or SCHEDUL for exceptions):

	CYBERS 174/175	IBM
M-F	8 am to 6 am	8 am to 6 am
SAT	8 AM - Midnight	8 AM - 6 AM
SUN	Noon - 6 AM	Noon - 6 AM

POLICY

CSO SHORT COURSES - SPRING SEMESTER

CSO is offering the following short courses during the 1982 spring semester to acquaint people with our facilities and the Control Data Corporation's (CDC) CYBER 175 and CYBER 174 computer systems.

To register for a course,

> please come in person to Room 150 DCL, or
> telephone 333-6630.

Registration is free and limited to 30 people in some classes. If you find that all of the available classes on a topic are full, please leave your name on our waiting list. We will call you if someone drops a class.

> ***PLEASE NOTE THAT YOU MAY NOT REGISTER FOR A CLASS UNTIL THE WEEK PRECEDING THE WEEK THE CLASS IS TO BE TAUGHT!***

Reminders of upcoming classes will be published in subsequent issues of *OFF-LINE.*

The classes offered are as follows:

INTRODUCTION TO THE CYBER COMPUTER SYSTEM

This course is intended for the first-time CYBER computer user. The emphasis is on time-sharing usage with discussions on the ICE text editor and card batch usage. One class was held at the end of January (before the date of this issue); another class will be offered:

Feb 22,24,26, Mar 1,3,5	2 PM - 3 PM	192 LH

EASY GRAPHING

This is a discussion of a high-level interactive plotting program for X-Y plots, bar charts and pie charts. Its English-like commands require no programming experience to generate plots. Familiarity with the CYBER is assumed. One class is offered:

Mar 29, 31, Apr 2	2 PM - 3 PM	192 LH

GRAPHICS COMPATIBILITY SYSTEM (GCS)

This class on GCS will cover plots on various graphics devices. Familiarity with FORTRAN and the CYBER is assumed. One class is offered:

Feb 15, 17, 19	2 PM - 3 PM	192 LH

INTRODUCTION TO RNF

This is a beginning-level discussion of the RNF text formatter used in the preparation of letters, manuals, or other documents. Topics will include tabbing, margins, spacing and paragraphing. Familiarity with the CYBER is assumed. Three classes are offered:

Feb 8, 10, 12	2 PM - 3 PM	192 LH
Mar 22, 24, 26	2 PM - 3 PM	192 LH
Apr 19, 21, 23	2 PM - 3 PM	192 LH

CYBER MAGNETIC TAPES

This class will cover the use of magnetic tapes on the CYBER. Familiarity with the use of the CYBER will be assumed. One class is offered.

Apr 5, 7, 9	2 PM - 3 PM	192 LH

BMDP

BMDP is a statistical package oriented toward the biological, medical, and agricultural sciences. This is an introductory course covering: data preparation, elements of the BMDP language, running BMDP programs from cards or at a terminal, data editing, creation and use of BMDP save files, and examples of using BMDP for descriptive statistics, regression and analysis of variance. The course, consisting of three 2-hour lectures, will be offered once.

Feb 16, 23, Mar 2	7:30 - 9:30 PM	124 Burrill Hall

INTRODUCTION TO SAS (Statistical Analysis System)

This is an introductory course on the SAS statistical package. Course topics will include: preparation of data for input, the SAS 'DATA' step, programming statements for data manipulation - creating new variables, deleting observations or variables, etc., basic statistics, and the general linear model programs GLM, ANOVA, and VARCOMP. The course, consisting of three 2-hour lectures, will be offered once:

Feb 23, 25, Mar 2	7 PM - 9 PM	243 Commerce West

INTRODUCTION TO SOUPAC (Statistically Oriented Users Package)

A lecture/laboratory will be presented on the SOUPAC statistical package on the CYBER. Handouts and examples will be provided illustrating basic statistics, regression and analysis of variance. Enough general CYBER terminology will be given to facilitate using SOUPAC. In the lab session, the use of terminals will be demonstrated, and exercises will be provided on editing, running and modifying SOUPAC programs. Instructors will act as consultants for solving problems during the lab session. The course, consisting of two 3-hour lectures and one 3-hour lab will be offered once:

Feb 2, 4 (lecture)	6:30-9:30 PM	243 Commerce West
Feb 6 (lab)	9 AM - 12 noon	70 Commerce West

MANOVA PROGRAM IN SPSS

This course will cover the MANOVA (Multivariate Analysis of Variance and Covariance) program in the SPSS statistical package. Emphasis will be placed on how to use DESIGN statements. Prior knowledge of the SPSS language will be assumed. The course, consisting of one 2-hour lecture, will be offered once.

Mar 4	7 PM - 9 PM	243 Commerce West

INTRODUCTION TO SPSS

This is an introductory course on the SPSS (Statistical Package for the Social Sciences) package on the CYBER. The course contents will include: preparation of data for input, basic components of the SPSS language, running SPSS programs from cards or at a terminal, using SPSS to modify the data, obtaining basic statistics and crosstabulation tables. The course, consisting of 6 hours of lecture and one 2 1/2 hour lab, will be offered twice. The lab session will provide participants with supervised "hands-on" experience running SPSS programs at terminals, and discussion of the generated output.

Feb 9, 11, 16 (lecture)	7 PM - 9 PM	243 Commerce West
Feb 13 (lab)	9:00-11:30 AM	70 Commerce West
Mar 22, 24 (lecture)	6:30-9:30 PM	115 DCL
Mar 27 (lab)	9:00-11:30 AM	70 Commerce West

CSO SEMINARS

In addition to the normal offering of short courses provided by CSO, we are offering a number of one-hour seminars on selected topics this semester. There is **NO REGISTRATION** for these seminars, simply show up on the appropriate day.

The following seminars are all offered in Room 115 DCL between 12 noon and 1 PM on Tuesdays and Thursdays.

- February 2 (Tues) - Text Processing: An overview of the tools available for producing documents, theses, letters, or manuals.
- February 4 (Thurs) - Cross Assemblers: An overview of the various microcomputer cross assemblers available.
- February 9 (Tues) - CCL: CYBER Control Language, how to make effective use of various CYBER commands.
- February 11 (Thurs) - Tape File Backup: Learn how to backup your disk file to magnetic tape.

- February 16 (Tues) - NCAR: An overview of the capabilities and use of the NCAR graphics package. Capabilities include contour plots, world maps, and 3-D plots.

- February 18 (Thurs) - Foreign Tapes: How to deal with magnetic tapes from another computer site and tapes to be sent elsewhere.

- February 23 (Tues) - Differential Equations: An overview of software for ordinary differential equations.

- February 25 (Thurs) - FTN4 vs FTN5: Comparison of the FORTRAN compilers for the ANSI 1966 and ANSI 1977 standards.

- March 2 (Tues) - ACSL: An overview of the advanced continuous simulation language.

- March 9 (Tues) - Partial Differential Equations: An overview of software for partial differential equations.

- March 11 (Thurs) - FORTRAN Futures: A discussion of progress at the current X3J3 ANSI meetings.

- March 23 (Tues) - Simulation: An overview of discrete and continuous simulation software.

- March 25 (Thurs) - Subroutine Libraries: How to create and use your own libraries of subroutines, and some useful loader commands.

- March 30 (Tues) - Linear Programming: An overview of the software for linear programming including APEX and MPOS.

- April 1 (Thurs) - Tape File Backup: Same as February 11 presentation.

- April 6 (Tues) - Linear Equations: An overview of software for solving $Ax=b$ especially where A is big.

- April 13 (Tues) - Eigenvalues: Software for handling ordinary and generalized eigenvalue problems.

- April 15 (Thurs) - Foreign Tapes: Same as February 18 presentation.

- April 20 (Tues) - Symbolic Manipulation: An overview of the REDUCE, FORMAC, and ALTRAN packages for formal algebraic manipulation.

CSO VIDEOTAPES AVAILABLE

CSO has produced a series of eight videotapes which introduce the novice to the CYBER Systems. A viewing guide containing the major pictorials used in the series is available and can be used to facilitate note taking.

The title and a brief synopsis of each of the videotapes is given below. Running time is 10-15 minutes for each videotape.

Introduction to Computing at CSO

A brief look at the steps required to solve a problem using a computer and some of the hardware used.

Using a Terminal

A description of the physical operation of a terminal and some of the keys that have a special meaning to the CYBER.

Introduction to CYBER Time-Sharing

A tutorial for logging on and off the CYBER.

File Usage - Local File and Indirect Access to Permanent Files

An introduction to CYBER files and the commands used to manipulate them.

Introduction to ICE Text Editing

A tutorial on entering and modifying files with ICE.

Running a FORTRAN Program - Concepts

The concepts of compilation, loading and execution.

Running a FORTRAN Program - The PROGRAM Statement

The PROGRAM statement and its relationship to files accessed by the program.

Running a FORTRAN Program - Control Statement

The control statements used to compile, load and execute a FORTRAN program.

Anyone can view these videotapes by going to the Undergraduate Library in person to make a reservation for use of the videotape equipment. Ask for a copy of the viewing guide when you check out the videotape for viewing.

A copy (Betamax format) of these videotapes is available for loan from CSO to any instructor wishing to use them in class. They were effectively used in this environment several times last semester with the instructor stopping the playback equipment whenever he/she wished to elaborate further, or to answer question which arose from the class.

To borrow a videotape for classroom use and obtain copies of the viewing guide for classroom distribution, call Scott Lathrop, 333-6618. If you do not have access to the required videotape equipment, Betamax viewing equipment can be borrowed from the Office of Instructional Resources, 333-3690.

CYBER

NEW VERSION OF LISP AVAILABLE

The most recent version of UT LISP, LISP 5.1, is now available on the CYBER systems. UT LISP 5.1 will permanently replace UT LISP 4.1 at the end of this semester. UT LISP 5.1 corrects bugs and adds many new capabilities to LISP. Among the enhancements are:

- A residential LISP editor with most of the capabilities of the UCILISP/INTERLISP editor.
- A BREAK package that stops on errors detected by LISP or breaks in execution programmed by the user, and understands many useful commands for examining the environment, correcting the error, and continuing execution as though the error has not occurred.
- A package for creating files of LISP functions and other LISP expressions that can be read by LISP as a whole or in part.
- A reallocation package that allows you to dynamically expand the size of available space and the execution stack.
- A new top-level interface that permits multiple inputs per line as well as editing and reexecution of previous user inputs.
- CATCH/THROW and other useful functions.
- A built-in HELP facility with manual excerpts describing the LISP functions and system variables.
- Functions to define macros, read macros, and print macros, plus functions and system control variables in support of these.

- A file of predefined system macros of general use to the LISP programmer.
- New iteration functions SOME, EVERY, SUBSET, MAPCAN, plus two general mapping macros LOOP and FOR.
- Functions PPRIN1 and GCTIME (which reports garbage collection time).
- A completely updated manual describing UT LISP version 5.1.

To access the LISP 5.1 interpreter, use the command:

```
GRAB,LISP/FUTURE.
```

To access the LISP 5.1 interpreter, the LISP assembler, the LISP compiler, and the LISP predefined macros, use the command:

```
GRAB,LISPSYS/FUTURE.
```

A copy of the LISP 5.1 manual is available for inspection in the Systems Consulting Office, 1208 W. Springfield. Arrangements are currently being made to obtain copies of the manual for sale. Watch fu ture issues of *OFF-LINE* for further information about this.

FEATURE ARTICLES

FORTRAN FUTURES

This article is part of a series on the work of the FORTRAN Standards Committee X3J3 in producing the next revision of the FORTRAN standard. The reader is reminded that the features described in this article are not a part of any FORTRAN compiler currently available, but rather are proposed requirements for FORTRAN processors produced in the late 1980's through the mid 1990's. Although every effort has been made to accurately describe the current position of X3J3 on these matters, the development of a revision to the FORTRAN standard is an evolutionary process, and these proposals may be subject to refinement, revision, or even retraction. Comments on these proposals may be given to Kurt Hirchert of the CSO Systems Consulting staff, who is a member of X3J3.

The original 1966 FORTRAN standard provided a facility for limited manipulation of character data in the form of Hollerith data, i.e., character data stored in real, integer, or logical variables. The facility described in the standard was extremely limited and was suitable primarily for supplying FORMATs or output titles at execution time, but implementations based on that standard often extended the Hollerith data facility. Common extensions included a simpler syntax for specifying Hollerith constants, use of Hollerith data and constants in less restricted contexts, and the use of the character conversion capabilities associated with formatted input/output on Hollerith data as well as external files. Thus, when X3J3 began work on what was eventually to become the FORTRAN 77 standard, extending the Hollerith data facility was one obvious possibility. On the other hand, it was equally obvious that use of the Hollerith data facility was the source of many of the more vexing problems in transporting FORTRAN from machine to machine. Even if the Hollerith data was being used only to hold

titles or variable FORMATs, problems could result from the fact that integer variables on different machines had different capacities for holding characters. If any kind of character manipulation was involved, the problems could become even more serious. As a result of these problems, X3J3 decided to drop the concept of Hollerith data from the FORTRAN 77 standard and provide equivalent facilities in the form of a data type specifically intended to hold characters. (Because it was anticipated that many FORTRAN 77 compilers would include Hollerith data as an extension in order to provide continuing support for existing programs, the Hollerith data facility was described in a non-binding appendix of the FORTRAN 77 standard, and, as expected, most FORTRAN 77 compilers do include a Hollerith data facility.) The problem of varying character capacities was avoided by making the capacity of a character variable part of its specification. The various special functions previously attributed to Hollerith data, such as variable FORMATs, were transferred to the new data type. Additional operations and intrinsic functions were defined on the new type to do such things as concatenate character strings, extract substrings, and compare characters lexicographically. The formatted input/output statements were extended to allow their application to character strings as well as external files.

In its work on the next revision to the FORTRAN standard, X3J3 has adopted a number of proposals which extend the FORTRAN 77 character facility even further. Many of these extensions are actually simply the removal of restrictions applied in FORTRAN 77. For example, it was necessary to impose restrictions on the use of some character dummy arguments in order to insure that all temporary storage in a FORTRAN 77 subroutine could be allocated on a static basis. Since a limited form of dynamic storage allocation is already a part of the adopted proposals for the next revision, these restrictions are no longer necessary. Restrictions on character assignment statements involving overlap between the left and right hand sides were simply removed. The substring extraction mechanism was extended to apply to constants as well as variables and array elements. It will now be possible to initialize long strings in a DATA statement using implied DO loops and substring notation much as it has previously been possible to initialize arrays with implied DO loops and subscript notation. The lower limit on character string size has been lowered from one to zero, including specification of how a character constant of zero length is written and what forms of substringing yield a zero length result. (Although it is unlikely that anyone would want a character variable that can hold no characters, expressions of zero length are of use in handling boundary conditions in many kinds of character manipulation problems.)

Another area of character extension has been in the adoption of additional intrinsic functions for manipulating characters. A large number have been considered. The following have already been adopted:

- FORTRAN 77 included functions ICHAR and CHAR to allow conversion between single characters and integers, but the specific integer value used for a given character was processor dependent. Functions IACHAR and ACHAR have been adopted to provide a more portable version based on the ASCII character code.

- Functions have been adopted for trimming trailing blanks from a character string, for shifting trailing blanks to the beginning of a string, and for shifting leading blanks to the end.

- The REPEAT function takes an input string and an integer count, and concatenates that many copies of the input string to produce the output string.

- A function has been adopted which locates the first character in a string which is also in a second string. A related function locates the first character in a string which is not in a second string.

Yet another area of extension has been the adoption of a second, closely related character data type. In the FORTRAN 77 character data type, each character variable has a fixed size. If the value on the right side of a character assignment statement is of a different length than the variable on the left side, it is truncated or padded with blanks to reach the correct size. In the new data type, each variable will have a fixed maximum size and will be able to hold any character length up to that maximum. Thus, padding and truncation will not occur as a side effect of assignment to variables of the new type. Expressions and assignment involving mixtures of the two types will be allowed, but correct type matching will be required across subprogram boundaries. All of the character intrinsics will be made generic, so arguments of either character type will be accepted.

In a related area, X3J3 has noted that in application areas like industrial process control and telemetry data analysis, integer variables are often used to manipulate strings of bits in much the same way integer variables were used to manipulate characters in the form of Hollerith data. Unfortunately, many of the same portability issues also are similar. Consequently, X3J3 has adopted a bit string data type similar to the FORTRAN 77 character string data type as a means of supporting the portability of such applications. Whenever possible, the syntax and operations on character strings have been carried over to bit strings. Padding and truncation is similarly performed, although the frequent use of bit strings with numeric interpretations has mandated that it be done at the left hand end rather than the right hand end, and zero bits rather than blanks are used as the pad value. The form of a bit string constant is similar to the form of a character string constant, but there are variations that give the programmer the freedom to express bit constants in binary, octal, or hexadecimal. Similarly, there are three different edit descriptors for performing formatted input/output of bit strings in binary, octal, or hexadecimal. In addition to the string oriented operations similar to those defined on character strings, there are bit operations similar to those defined for logical variables. Intrinsics have been provided to translate between single bits and logical variables, and to translate between bit strings and integer variables. There are also a few "miscellaneous" bit intrinsics, such as an intrinsic function to count the number of one bits in a bit string.

MISCELLANEOUS

NECC-82 CONFERENCE

The 1982 National Educational Computing Conference, NECC-82, is to be held in Kansas City, Missouri, June 28-30, 1982, as previously announced in the September 1981 issue of *OFF-LINE*. Anyone desiring additional information about this conference is asked to contact:

E. Michael Staman
NECC-82 General Chairman
Campus Computing Services
University of Missouri-Columbia
305 Jesse Hall
Columbia, MO 65211

SALES - EXCHANGES - HELP WANTED

DATABASE MANAGER POSITION

DATABASE MANAGER, 1/2-time graduate research assistant position, using the administrative computer system and Model 204 database management software. Contact Dean Stapleton, Graduate College, 314 Administration Bldg., 333-6807 or 333-0073.

STUDENTS WANTED

STUDENTS WANTED -- to write programs for Campus Administration System. Can work hourly or on a contract basis. Please contact Sue Greenberg, Assistant Director for Development, 333-6289.

OFF-LINE's Mailing List

If you wish to be placed on our mailing list for future issues of *OFF-LINE*, if you wish to be removed from the list, or if you wish to enter an address correction, please complete and return this page. (Current subscribers are kept on the mailing list until a specific request for removal is received, or until a mailing is returned as undeliverable.)

Check one:
- ☐ New subscriber
- ☐ Removal request
- ☐ Address correction

Name: _____

Address: _____

_____

_____

CAMPUS or Zip Code _____

(If address correction, give old address and zip code below.)

Comments:

SEND TO:

OFF-LINE
150 Digital Computer Laboratory
University of Illinois at Urbana-Champaign
1304 West Springfield Avenue
Urbana, Illinois 61801

Computing Services Office

Off-Line

University of Illinois at Urbana-Champaign VOL. 10, NO. 3 March 1982

EDITOR: Lynn Bilger
PHONE: (217) 333-6236
150 Digital Computer Lab
1304 W. Springfield Ave.
Urbana, Illinois 61801

CSO DIRECTORY - STAFF AND SERVICES

Administrative

Director	George Badger	150	DCL	333-4103
Business Manager	Stanley Rankin	150	DCL	333-6530
Secretary	Joyce McCabe	150	DCL	333-1637

User Services and Hardware/Software Support

User Accounting		**1208**	**W Springfield**	**333-7752**
Distribution Office		**1208**	**W Springfield**	**333-6760**
Systems Consulting		**1208**	**W Springfield**	**333-6133**
Statistical Services Consulting		**65**	**Comm West**	**333-2170**
Text Processing Consulting		**207**	**Astronomy**	**333-7318**
Terminal Repair Service		**150**	**DCL**	**333-0969**
CYBER Dial-up Numbers	**CYBER 175**	**110-300**	**baud**	**333-4000**
	CYBER 175	**1200**	**baud**	**333-4001**
	CYBER 174	**110-300**	**baud**	**333-4004**
Asst Dir User Services	Robert Penka	173	DCL	333-4709
Asst Dir Systems and Operations	Sandra Moy	177	DCL	333-4703
Asst Dir Development	J. M. Randal	120	DCL	333-9772
Asst Dir Engineering and Hardware Selection	Cliff Carter	195	DCL	333-3723
Manager Statistical Services	Beth Richardson	91	Comm West	333-2171
Documentation	Lynn Bilger	139	Astronomy	333-6236
CYBER-IBM Operations	Jack Knott	194a	DCL	333-6562
UNIX Operations	Debbie Weller	123	DCL	333-8150
Telecommunications	Tom Kerkering	164	DCL	333-0816
Laboratory Support Project	Mike Gardner	164	DCL	333-7904
RJE Operations North	Rex Duzan	162	DCL	333-6285
RJE Operations South	Don McCabe	1208	W Springfield	333-2171 333-7752

RJE Sites (see Reference Guide RF-0.3 for operating hours)

Agriculture	M103	Turner Hall	333-8170
Chemistry	153	Noyes Lab	333-1728
Commerce West	70	Comm West	333-4500
CRH Snack Bar	120	Snack Bar	333-1851
DCL Routing Room	14	DCL	333-6203
Electrical Engineering	146	EEB	333-4936
Florida Ave Res Hall		FAR	333-2695
Illinois St Res Hall		ISR	333-0307
Mechanical Engineering	65	MEB	333-1430
Psychology	453	Psych Bldg.	333-7531
Social Science	202	Lincoln Hall	333-0309

OFF-LINE is the monthly newsletter of the Computing Services Office at the University of Illinois at Urbana-Champaign. Unless otherwise indicated, permission to reprint is freely granted, provided that the author, if named, and the Computing Services Office (CSO) are credited. Information in this issue is current as of February 19, 1982.

CSO operates a CDC CYBER 175 with 262K words of central memory and a CDC CYBER 174 with 196K words of central memory. The 175 and 174 run under the NOS Operating System and share 512K words of ECS. The 175 serves over 200 simultaneously active text and graphics terminals and the 174 serves over 100 simultaneously active terminals. CSO also operates an IBM 4341 with 4 million bytes of memory running HASP-OS/MVT under VM. In addition, CSO operates a DEC PDP-11/50 with 252K bytes of core driving a GSI CAT-8 phototypesetter and a DEC VAX 11/780 with 2 million bytes of core, both running under the UNIX Operating System.

Operating Hours (see HEARYE or SCHEDUL for exceptions):

	CYBERS 174/175	IBM
M-F	8 am to 6 am	8 am to 6 am
SAT	8 AM - Midnight	8 AM - 6 AM
SUN	Noon - 6 AM	Noon - 6 AM

POLICY

CSO SHORT COURSES - SPRING SEMESTER

This is a reminder of the short courses being offered by CSO during March and April of the spring semester to acquaint people with our facilities and the Control Data Corporation's (CDC) CYBER 175 and CYBER 174 computer systems.

To register for a course,

> please come in person to Room 150 DCL, or
> telephone 333-6630.

Registration is free and limited to 30 people in some classes. If you find that all of the available classes on a topic are full, please leave your name on our waiting list. We will call you if someone drops a class.

> ***PLEASE NOTE THAT YOU MAY NOT REGISTER FOR A CLASS UNTIL THE WEEK PRECEDING THE WEEK THE CLASS IS TO BE TAUGHT!***

The classes offered are as follows:

EASY GRAPHING

This is a discussion of a high-level interactive plotting program for X-Y plots, bar charts and pie charts. Its English-like commands require no programming experience to generate plots. Familiarity with the CYBER is assumed. One class is offered:

> Mar 29, 31, Apr 2 2 PM - 3 PM 192 LH

INTRODUCTION TO RNF

This is a beginning-level discussion of the RNF text formatter used in the preparation of letters, manuals, or other documents. Topics will include tabbing, margins, spacing and paragraphing. Familiarity with the CYBER is assumed. Two classes are offered:

> Mar 22, 24, 26 2 PM - 3 PM 192 LH
> Apr 19, 21, 23 2 PM - 3 PM 192 LH

CYBER MAGNETIC TAPES

This class will cover the use of magnetic tapes on the CYBER. Familiarity with the use of the CYBER will be assumed. One class is offered.

> Apr 5, 7, 9 2 PM - 3 PM 192 LH

INTRODUCTION TO SPSS

This is an introductory course on the SPSS (Statistical Package for the Social Sciences) package on the CYBER. The course contents will include: preparation of data for input, basic components of the SPSS language, running SPSS programs from cards or at a terminal, using SPSS to modify the data, obtaining basic statistics and crosstabulation tables. The course, consisting of 6 hours of lecture and one 2 1/2 hour lab, will be offered once. The lab session will provide participants with supervised "hands-on" experience running SPSS programs at terminals, and discussion of the generated output.

Mar 22, 24 (lecture) 6:30-9:30 PM 115 DCL
Mar 27 (lab) 9:00-11:30 AM 70 Commerce West

CSO SEMINARS

In addition to the normal offering of short courses provided by CSO, we are offering a number of one-hour seminars on selected topics this semester. There is **NO REGISTRATION** for these seminars, simply show up on the appropriate day.

The following seminars are all offered in Room 115 DCL between 12 noon and 1 PM on Tuesdays and Thursdays.

- March 9 (Tues) - Partial Differential Equations: An overview of software for partial differential equations.
- March 11 (Thurs) - FORTRAN Futures: A discussion of progress at the current X3J3 ANSI meetings.
- March 23 (Tues) - Simulation: An overview of discrete and continuous simulation software.
- March 25 (Thurs) - Subroutine Libraries: How to create and use your own libraries of subroutines, and some useful loader commands.
- March 30 (Tues) - Linear Programming: An overview of the software for linear programming including APEX and MPOS.
- April 1 (Thurs) - Tape File Backup: Learn how to backup your disk file to magnetic tape.
- April 6 (Tues) - Linear Equations: An overview of software for solving $Ax=b$ especially where A is big.
- April 13 (Tues) - Eigenvalues: Software for handling ordinary and generalized eigenvalue problems.

- April 15 (Thurs) - Foreign Tapes: How to deal with magnetic tapes from another computer site and tapes to be sent elsewhere.

- April 20 (Tues) - Symbolic Manipulation: An overview of the REDUCE, FORMAC, and ALTRAN packages for formal algebraic manipulation.

SAVING COPIES OF OUTPUT

Users are advised to save copies in disk files of print, punch and plot output produced by expensive or long-running jobs. These file copies would serve as insurance against possible loss or destruction of printed, punched, or plotted output, and can be easily purged once the output is safely in hand. This is especially important to time-sharing users since we ordinarily do not give refunds for lost time-sharing work. Batch users also can benefit since it is usually much more convenient to requeue a file for output than to regenerate the data in the file.

CYBER

THE CYBER FILE MIGRATION SYSTEM

Disks, like airline seats, are commodities which are much too expensive to leave unused. Whereas unused airline seats are defined as empty, unused disks can be either empty or full of unused files. There are two possible solutions to the problem. One is to absorb the cost of the unused space (and of course, pass the additional cost on to the users); the other is to deliberately overbook the space. Both the airlines and CSO find it more cost effective to do the latter.

We allow about 3,800,000 PRUs of permanent files to reside on our permanent file system. This has been determined, mostly by trial and error, to be the maximum number which still allows the system to provide reasonable response times during the afternoon peak. (The system needs free disk space to swap executing jobs from core, allocate local files, and store new or extended files.) At any given time on our system, about four times the number of PRUs are migrated as are resident on our disks.

Nightly, during the third shift, the operator runs a procedure to enforce the 3,800,000 PRU limit to insure that we can make it through the following day. The programs which run as part of this procedure read the entire permanent file catalog. They ignore any files which already have been migrated or have been accessed during the previous working day. Since this process occurs after midnight, any files accessed during that day will be on disk the next day (files accessed on Friday will be on disk on Monday since Saturday and Sunday are not working days).

For all files which have not been accessed that day or already migrated, the procedure assigns a priority according to the formula:

priority = size * # working days since access * (unclaimed)

Where *(unclaimed)* is normally 1. If, however, the file has been marked as "must be claimed" (i.e., the DIRECT program prefaces its name with a minus sign), then *(unclaimed)* takes as a value the number of days the file has been in the "must be claimed" state. This unclaimed factor causes disk files which have no one paying for them to be taken off disk faster than normal. Files are then chosen for migration in decreasing priority order until the disk space left is within acceptable limits. Normally, files with a priority of less than 2000 are chosen infrequently for migration, and files with a priority of less than 1000 are rarely chosen.

To help illustrate this (assuming that files of priority less than 1000 are never chosen), the following table shows the number of working days a file of a particular size should remain on disk if it is not accessed or not claimed.

Size (PRUs)	*Days Unaccessed*	*Days Unclaimed*
5	280	22
10	140	15
50	28	7
100	14	5
250	6	3
600	2	2
1000	1	1

If a file is chosen to be migrated, the actual freeing of its disk space can occur in two ways. If the file was previously migrated and was reloaded but not modified, then the version of the file on the migration tape is still current and the file can just be marked as migrated and its disk space freed. If the file has never been migrated, or has been reloaded and modified, then the current version of the file on disk must be written to a migration tape. After a number of files are written to a migration tape, and either the tape is full or enough space has been freed on the disks, the tape is rewound and read to insure its integrity. Only after this has occurred, will a file be marked as migrated and its disk space freed. To further insure reliability, the tape is again rewound and copied to another tape. This second tape is then stored in another campus building to insure that the files are still "saved" in the event the original tape is damaged.

After migration runs are completed for the night, the tapes are again mounted and copied to a Mass Storage System (MSS) which we have acquired from CDC. The MSS is sort of a "juke box" for cartridges of magnetic tape. Each cartridge contains about one quarter of a 1600 bpi reel of tape and the MSS holds 2000 cartridges. The device, although fairly cantankerous, can mount, locate and retrieve a file and return the cartridge in about a minute without operator intervention.

This is, no doubt, enough information about the removal of files. The important thing to most people is how to get a file back onto disk once it has been removed. When a RELOAD or a USE command is executed for a particular file, the request is added to a list of files to be reloaded. If there already is a request in the list for that particular file, the new request is

ignored. When the MSS is ready to reload another file, it scans the list and assigns a priority to each file in the list. The priority it assigns is the numnber 1 divided by the number of outstanding RELOAD requests for files belonging to the same user number. The MSS then attempts to reload the file with the highest priority. If the reload is successful, the file is back on disk and its entry in the list is deleted. If the reload from the MSS fails or cannot be done (this happens for about 5% of all reload requests), the file is marked as a MSS failure in the list. These failed reloads then are reloaded from tape.

A tape reload processor periodically scans the list and begins the reloading of files which have been marked as MSS failures. The tape processor issues a request to the operator to mount a migration tape and then reloads the files from it. To speed up the entire reloading process, after MSS failures have been reloaded from a tape, the tape processor will reload any other files in the list which are also on the mounted tape. Finally, if the tape processor completes reloading all failed files and finds requests that are more than 1/2 hour old, it will reload these files to help the MSS.

In conclusion:

- Migrated files are safe. The files reside on two tapes and, most likely, the MSS.
- The migration of a file, however, does not affect its deletion by the enforcement of CSO's disk policy (the purging of files unused for one year or unclaimed for 120 days).
- The response to a reload request for one file should be fairly quick; however, the response will be slower for requests of more than one file at the same time. If you issue a request to reload the same file repeatedly, it does not affect the speed of the reloading process, since these subsequent requests are ignored. The average time to reload a file from the MSS is less than five minutes, and the average time from tape is about 30 minutes.
- The type of operation (access or modification) performed on a file does not affect the likelihood of the file being migrated.
- The larger a file is, the more prone it is to be migrated. Also, taking into consideration the priority formula and the fact that files with a calculated priority of less than 1000 are never migrated, one can deduce that a validly claimed file of less than four PRUs should never be migrated. This is because 4 PRUs * 250 (approximate number of work days in a calendar year) equals 1000. Remember, however, that a file that has been unaccessed for 365 days (even if less than 100 PRUs) will be purged for nonuse.

SIR/DBMS SUPPORT FUNCTION ASSUMED BY SURVEY RESEARCH LABORATORY

A working agreement has been reached between the Computer Services Office and the Survey Research Laboratory under which the data processing staff of SRL will now be providing support and assistance for the Scientific Information Retrieval (SIR) data base management system. Richard Williams of the SRL data processing section will act as campus-wide coordinator for SIR/DBMS and will be responsible for installation, maintenance and support of the package as well as consulting services for users.

A new release of SIR/DBMS, Version I.IN, has been installed on the CYBER system and includes the following capabilities:

- SIR/DBMS can handle hierarchical data structures with extensive facilities for schema definition, data base loading and a sophisticated information retrieval programming language, all with syntax patterned after the SPSS package.

- SIR/DBMS enables the user to perform basic variable consistency edits during data base loading and extensive data transformation commands which can be executed during data loading, updating, or retrieval.

- SIR/DBMS interfaces directly with SPSS and BMDP on the CYBER, and provides facilities for producing raw data records which may be used with any set of analysis routines. In addition, users may generate new SIR data bases or subsets of data bases during retrieval.

- SIR/DBMS also has an extremely flexible report generator, and can produce complex hierarchical reports including reports with multiple levels of headings, subtotal and grand total computation and output.

- SIR/DBMS includes a full array of utilities for data base backup, dumping, record listing, and merging of multiple SIR data bases.

- A fairly powerful interactive mode is also available in SIR/DBMS which includes a text editor, storage of data base procedures, and the ability to escape to the NOS operating system for single commands.

- A new utility for converting data bases from the Version I.I format to the next generation of the package described below.

A new version of SIR/DBMS (V2.1) is expected during the month of March 1982 that will contain a number of significant enhancements. In particular, SIR/DBMS will then include:

- The capability to handle network data structures and perform full hierarchical retrievals.

- A structured retrieval programming language with greater flexibility in the handling of information retrieved from the data base.

- A sequential processing capability for large data bases residing on tape.

- A number of significant improvements to the interactive editor including the full escape to NOS, an ICE-like MODIFY command, and the provision for multiple editor workspaces.

- A full journal backup capability. During batch data input, users can create journal files which can then be used to update data bases damaged by hardware, software, or user error. In this way, updates or changes to the data base may be recovered in the event that full backups have been done infrequently.

- An export/import facility for conversion of data bases to SIR 2.1. The export procedure provided in release 1.1N will perform the necessary conversions of Version 1 data bases and SIR 2.1 will contain import procedures to recreate the data bases in the new format.

The CSO Distribution Office (1208 W. Springfield) will continue to stock the SIR/DBMS manuals for Version 1.1 until the end of February, 1982. After that time, only SIR Version 2 manuals will be available from the Distribution Office. In addition, an update package will be available which describes the extensions to SIR 2.1 on the CYBER. SIR, Inc. will continue to support the SIR/DBMS 1.1 for at least a year following the release of the new version.

To use the newest release of SIR 1.1, enter the following commands:

```
GRAB,SIR/SHARED.
SIR,parameter list.
```

Users seeking assistance or consulting for SIR/DBMS should contact Richard Williams of the Survey Research Laboratory at Room 3, 1005 W. Nevada, 333-3494. Up to four hours of consultation on SIR/DBMS will be provided without charge by the staff of SRL. In addition, non-credit workshops are being planned which will provide an in-depth introduction to SIR/DBMS and to some fundamental problems of data base management.

MATHEMATICAL SERVICES

TRANSACTIONS ON MATHEMATICAL SOFTWARE

We now have on tape all algorithms published in *Transactions on Mathematical Software* up to September 1981 (from algorithm 493, published in the first issue of March 1975, up to algorithm 577). The algorithms from the December 1981 issue have been ordered, but have not yet arrived. The directory of algorithms in file TOMS/UN=MATHLIB has been updated. To print this directory, enter:

```
GET,TOMS/UN=MATHLIB.
PRINT,TOMS.
```

The "Math Note" series described in the March 1981 issue of *OFF-LINE* includes a note summarizing all algorithms published in TOMS to date. To print this note (number 6), enter:

```
GET,MNOTES/UN=MATHLIB.
MNOTES,6.
PRINT,OUT/AS/CC/EJ.
```

(To print instructions on the use of MNOTES, substitute a 0 (zero) for 6 in the example above.)

The algorithms published in the September and December 1981 issues are as follows (the December algorithms have not arrived yet):

In the September issue:

#573: NL2SOL -- An Adaptive Nonlinear Least Squares Algorithm
#574: Shape-Preserving Osculatory Quadratic Splines
#575: Permutations for a Zero-Free Diagonal
#576: A FORTRAN Program for Solving Ax=b
#577: Algorithms for Incomplete Elliptic Integrals

In the December issue:

#578: Solution of Real Linear Equations in a Paged Virtual Store
#579: CPSC: Complex Power Series Coefficients
#580: QRUP: A Set of FORTRAN Routines for Updating QR Factorizations

The following articles which appeared in these issues may be of interest to some of our users:

In the September issue:

A Set of Library Routines for Solving Parabolic Equations in One Space Variable, by P. M. Dew and J. E. Walsh.

An Algorithm for Computing a Shape-Preserving Osculatory Quadratic Spline, by David F. McAllister and John A. Roulier (this paper describes algorithm 574).

An Adaptive Nonlinear Least-Squares Algorithm, by John E. Dennis Jr., David M. Gay and Roy E. Welsch (this paper describes algorithm 573, NL2SOL).

In the December issue:

The Design of the XMP Linear Programming Library, by Roy E. Marsten. (XMP is available on the CYBER.)

ME28: A Sparse Unsymmetric Linear Equation Solver for Complex Equations, by I. S. Duff. (We have ME28 in our CYBER HARWELL library under SHARED.)

If you are looking for software, *Transactions on Mathematical Software* is one of the journals to check, along with *Numerische Mathematik, BIT, Computer Journal,* and various publications of ACM (the Association for Computing Machinery) and SIAM (the Society for Industrial and Applied Mathematics).

QUADRATIC SPLINE ROUTINES ADDED TO UOILIB

Algorithm 574 from *Transactions on Mathematical Software*, September 1981, "Shape Preserving Osculatory Quadratic Splines" has been added to UOILIB.

A spline is a curve which is polynomial in pieces; it is composed of several curves, each a polynomial, whose coefficients are chosen so that the pieces "match up" with specified degrees of smoothness at certain points called the "knots" or nodes of the spline. Splines have long been used for purposes of interpolation, in particular the cubic spline, whose pieces are all cubics and which has two continuous derivatives everywhere. Good algorithms for higher degree splines now exist, however, and many were published by Carl de Boor in his book, *A Practical Guide to Splines;* these codes form the BSPLINE library (accessed on the CYBER via SHARED,BSPLINE.).

One problem of spline interpolation which is often difficult is getting the spline to "follow" the shape of the given data in a reasonable way. A cubic spline often "twists" around the data points it fits, and introduces changes of curvature or inflection points which the user may find undesirable, even though for most purposes of interpolation it is satisfactory. If the data is steadily decreasing and "obviously" concave downward, say, the user may strongly desire an interpolating curve which is also steadily decreasing and concave downward everywhere; This is difficult to guarantee with regular splines. The quadratic spline algorithm published in TOMS is intended to produce a curve which truly behaves the way the data does. Of course, the assumption is that the data is in fact correct, and has practically no errors, or errors so small as not to affect the "true" shape of the curve; if this is not the case (say there is some random noise in it), it should be smoothed by some technique before interpolation.

The names of the routines as published in TOMS were changed for use here to the following:

QSPLNS This routine, when called with given data points, estimates values for the derivatives at the data points, in such a way as not to force the introduction of spurious inflection points during interpolation. (Such spurious inflection points show up as an oscillation of the interpolated curve around the data; the curve "twists" around the data rather than following it smoothly.)

QSPLNI Given a set of data points, and a set of slopes at those points (the slopes can be calculated by QSPLNS or any technique that seems reasonable), QSPLNI interpolates a quadratic spline through the given data points, evaluating it at a set of specified points.

A writeup on both of these routines can be obtained from the MATH procedure as follows:

```
GRAB,MATH.
MATH,UOILIB,QSPLNI.
PRINT,DOC.
```

Of course, since the routines are in UOILIB, to use them you must enter the command

```
GRAB,UOILIB.
```

before running your program.

REDUCE FOR LISP 5.1

A version of the symbolic manipulation language REDUCE has been installed in FUTURE to run under the new version of LISP on the CYBER. The command

```
GRAB,REDUCE/F.
```

will access this version of REDUCE and the LISP 5.1 interpreter. It will become the default version of REDUCE at the end of the spring semester when the new LISP becomes the default.

Unfortunately, because of new features in LISP 5.1 which consume more memory, REDUCE has less memory for its own workspace, so, the same problem can take noticeably more time to run under the new version then the old. If (and only if) there is sufficient demand for it, these old versions can be kept in some accessible place, though they cannot be maintained or their performance warranted after the new versions become the system default.

IBM

NEW VERSION OF FORMAC

A new version of the FORMAC package for symbolic manipulation has been installed on the IBM. Called, FORMAC77, this version runs under the PL/1 optimizing compiler (the current FORMAC uses PL/1 level F). The new version will not replace the old version until the end of the spring semester. Until that time, the new version can be accessed as follows:

```
//PROCLIB DD DSN=SYS9.FORMAC.PROCLIB,DISP=SHR
// EXEC FORMAC
//FORMAC.SYSIN DD *
        (FORMAC program)
//GO.SYSIN DD *
        (data)
/*
```

This procedure (FORMAC) preprocesses, compiles, link edits, and executes a FORMAC77 program. The PROCLIB mentioned above contains several other procedures as well, analogous to the current FORMAC procedures:

FMAC	Preprocess and compile only
FMACLDGO	Preprocess, compile, load and execute
FMACLKED	Preprocess, compile and link edit to produce a load module which can be saved
GOFMAC	execute a saved FORMAC77 load module

The distribution materials for this version of FORMAC included a machine readable version of the language manual, which has been converted to upper/lower case and supplemented with several examples from previous FORMAC manuals, and information on local usage. This manual is still being worked on, but the information in it describing the FORMAC language is now complete. It can be obtained and printed as follows:

```
/*ID EJECT=YES,LINES=7000,TIME=1
// EXEC COPY
//SYSUT1 DD DSN=SYS1.FORMAC77.SAMPLES(MANUAL),DISP=SHR
//SYSUT2 DD SYSOUT=A,DCB=RECFM=FA
```

For those with access to the CYBER, the manual can also be printed from that system as follows:

```
ATTACH,FMACDOC/UN=MATHLIB.
PRINT,FMACDOC/AS/CC/EJ.
```

Five sample programs, which are discussed in the manual, have been placed in the dataset SYS1.FORMAC77.SAMPLES. Called EXAMPLE1, EXAMPLE2, EXAMPLE3, EXAMPLE4, and EXAMPLE5, they can be listed as follows:

```
// EXEC COPY
//SYSUT1 DD DSN=SYS1.FORMAC77.SAMPLES(EXAMPLEn),DISP=SHR
//SYSUT2 DD SYSOUT=A
```

Where n is 1, 2, 3, 4, or 5.

FORMAC is still under development by Knut Bahr in Europe, who adopted it some years ago after the IBM FORMAC project expired. Herman Knoble at Penn State is also working on it, and a CMS-usable version of it should be distributed through the IBM users group SHARE sometime later this year.

Although we have symbolic manipulation packages on the CYBER (REDUCE and ALTRAN), they suffer from the inherent memory limitations of the CDC machine. FORMAC, running on the IBM system with virtual memory, has the potential for solving much larger problems in algebra than either REDUCE or ALTRAN. (We also have ALTRAN on the IBM, but its capabilities are limited. For large problems which can be efficiently formulated using rational functions, ALTRAN can outstrip FORMAC.)

FEATURE ARTICLES

FORTRAN FUTURES

This article is part of a series on the work of the FORTRAN Standards Committee X3J3 in producing the next revision of the FORTRAN standard. The reader is reminded that the features described in this article are not a part of any FORTRAN compiler currently available, but rather are proposed requirements for FORTRAN processors produced in the late 1980's through the mid 1990's. Although every effort has been made to accurately describe the current position of X3J3 on these matters, the development of a revision to the FORTRAN standard is an evolutionary process, and these proposals may be subject to refinement, revision, or even retraction. Comments on these proposals may be given to Kurt Hirchert of the CSO Systems Consulting staff, who is a member of X3J3.

Although more numerical software is written in FORTRAN than any other language, the FORTRAN standard has provided relatively little assistance to the writer of such software, especially if the software is to be portable. For the past three years, X3J3 has had a closer liaison with the numerical software community and has adopted proposals to improve this situation.

Even the existing feature which has been of the greatest benefit in writing numerical software, the ability to use either single or double precision arithmetic, has been something of a mixed blessing. The problem has been that the definition of single and double precision has been based on storage properties rather than numerical properties, so the selection of single and double precision has been machine dependent. On machines with shorter word lengths, nearly all numerical software is written in double precision, but on machines with

longer word lengths, single precision is more commonly used. The FORTRAN 77 standard improved the situation by standardizing the IMPLICIT statement and introducing generic intrinsic functions which applied equally to both types, but many problems remained in converting a program between single and double precision, such as the conversion between single and double precision constants.

In the coming revision of the FORTRAN standard, these problems are being addressed through a major reworking of the concept of precision in FORTRAN. The DOUBLE PRECISION data type will cease to be a separate data type, except in supporting existing programs. Instead, there is the recognition that a FORTRAN processor may have several machine data types corresponding to the FORTRAN REAL data type. The programmer will be able to select among these machine data types on the basis of minimum requirements for precision and exponent range. Thus the programmer's numerical requirements will be used by the compiler to determine whether the machine single precision or double precision (or even quad precision) will be used. In the simplest case, the programmer will specify general numeric requirements to apply to all real variables and constants not given more specific requirements. (In the trivial case, the compiler will provide defaults for even these requirements.) For those few programmers who need variables with differing precision requirements in a single program, this capability will also be available. Constants will have the same default precision properties as variables, but, analogous to the current handling of double precision constants, it will be possible to associate other precision properties with constants written using other letters preceding the exponent part of the constant.

There are two interesting side effects of this approach to precision control. First, in contrast to the existing FORTRAN standard which does not provide for a DOUBLE COMPLEX type, the new proposals call for a complex data type corresponding to each real data type. Second, as an aid in writing library functions and subroutines which are usable no matter what precision requirements are specified, the concept of generic routines has been extended from the intrinsic functions to include user-written routines.

The second major area of numerical extensions in the coming FORTRAN standard will rarely be used directly by the typical FORTRAN programmer, but will often be used indirectly because of its benefit to the writers of numerical packages such as IMSL and EISPACK. These are intrinsics which return information about the numerical properties of the processor. These include intrinsics to return the machine radix for each real and integer data type, the number of digits in the representation, and the minimum and maximum exponents available in a real representation, as well as numbers which can be derived from these values, such as the minimum and maximum representable values and the machine "epsilon". In addition to these "constant" functions, more variable intrinsics include functions to obtain a real number's closest neighbor (on either the positive or negative side) and functions to assist in scaling real values by powers of the machine radix (for maximum accuracy).

X3J3 hopes that these facilities, along with other facilities adopted for more general use, will greatly enhance the ability of the writers of numerical software to write programs which are easily and efficiently transported between machines.

Recent Actions by X3J3

The 81st meeting of X3J3 was held February 8-12, 1982 in La Jolla, California. Actions taken at the meeting include the following:

- A facility was adopted to allow users to write functions returning array values.
- Several other aspects of previously passed array proposals were clarified, including the adoption of syntax relating array selection and character substrings.
- A proposal was adopted allowing block constructs such as the block-IF to be named as an aid in locating nesting errors.
- Several details in previously passed control structure facilities were clarified.
- The internal procedure facility was generalized to allow internal procedures to be passed to dummy procedure arguments and to allow internal procedures to have internal procedures nested within them.
- Aspects of the SHARED DATA proposal adopted at the last meeting were clarified.
- The procedure interface was further extended with a specification statement to indicate which arguments could be omitted and another specification statement to indicate whether a dummy argument was only to be used as a value or whether it could be modified.
- X3J3 agreed, in principle, to allow keyword argument specification on at least some of the intrinsic functions, but deferred the selection of specific keywords for this purpose to a later date.
- X3J3 reviewed the draft standard for industrial real-time use of FORTRAN and found it in conformance with the FORTRAN 77 standard. The draft standard was thus recommended for adoption at the international level.
- A number of clarifications were adopted on the use of the varying length character data type. A few of the proposed clarifications were referred back to subcommittee for additional work.
- A preliminary definition of the "core" of the new revision was adopted. FORTRAN 77 features not in "core" will continue to be supported by the next revision of the standard, but will be marked for eventual removal from language in the revision following it (i.e., in the late 1990's or early 2000's). New features not in "core" will be optional and may not be implemented in some FORTRAN processors, such as those developed for smaller computers.
- A special "engineering" or "metric" variation on the E edit descriptor was adopted. Output using this edit descriptor is similar to that produced using the E descriptor, but the exponent is forced to be a multiple of 3 for easier reference to the standard metric units.

- A presentation was made to the committee on the possible language implications of the proposed IEEE standard on microcomputer floating point hardware.

- The committee established liaison to consider the language implications of the open systems interconnection standards.

- A presentation was made on a framework for adding error/exception/event handling to FORTRAN.

- Further proposals were considered on adding a macro expansion facility to the FORTRAN standard, but no action was taken on them because of ambiguity in the proposed text.

- There was extensive discussion on the way in which the committee should incorporate the proposals passed thus far into the existing language to produce a first draft of the next revision of the standard. In a separate discussion, the rationale behind the development of certain language extension modules was considered, and their relation to the "core" document was explored.

- Possibilities for the specific syntax of the name-directed input/output (NAMELIST) facility were discussed.

- Means of determining the exact number of values obtained by a READ statement were considered. Related proposals would also make the handling of end-of-file and errors by the FORTRAN processor more predictable.

- Requests for interpretations of the FORTRAN 77 standard were considered.

- Issues relating to the extension of the SAVE and DATA statements to recursive procedures were discussed.

- A proposal for the syntax of array valued constants was considered.

SALES - EXCHANGES - HELP WANTED

RESEARCH APPLICATIONS PROGRAMMER

A full-time, 12-month position is available for a Research Application Programmer (rank open, depending upon the level of education and experience). Duties and responsibilities are to assist the scientific staff in data storage, retrieval and analysis of biological and bibliographic information.

Applicant should have a Bachelor's or Master's degree and knowledge of scientific programming and modern data management techniques. Preferred applicants will have a degree in computer science, have experience with computer systems of the kind available at the University of Illinois, and be able to provide samples of data application coded in FORTRAN or a similar language.

Salary is commensurate with qualifications and experience. Application deadline is April 1, 1982, or until position is filled. Applicants should send a letter of application and a resume, including names and addresses of 4 references to:

Dr. William Luckmann
Illinois Natural History Survey
172 Natural Resources Building
University of Illinois
607 E. Peabody Drive
Champaign, Illinois 61820

THE UNIVERSITY OF ILLINOIS IS AN AFFIRMATIVE ACTION/EQUAL OPPORTUNITY

OFF-LINE's Mailing List

If you wish to be placed on our mailing list for future issues of *OFF-LINE*, if you wish to be removed from the list, or if you wish to enter an address correction, please complete and return this page. (Current subscribers are kept on the mailing list until a specific request for removal is received, or until a mailing is returned as undeliverable.)

Check one:
- ☐ New subscriber
- ☐ Removal request
- ☐ Address correction

Name: _____

Address: _____

_____

_____

CAMPUS or Zip Code _____

(If address correction, give old address and zip code below.)

Comments:

RETURN TO:

OFF-LINE
150 Digital Computer Laboratory
University of Illinois at Urbana-Champaign
1304 West Springfield Avenue
Urbana, Illinois 61801

Computing Services Office

Off-Line

University of Illinois at Urbana-Champaign VOL. 10, NO. 5 May 1982

EDITOR: Lynn Bilger
PHONE: (217) 333-6236
150 Digital Computer Lab
1304 W. Springfield Ave.
Urbana, Illinois 61801

Page *Contents*

CSO DIRECTORY - STAFF AND SERVICES

Administrative

Director	George Badger	150	DCL	333-4103
Business Manager	Stanley Rankin	150	DCL	333-6530
Secretary	Joyce McCabe	150	DCL	333-1637

User Services and Hardware/Software Support

User Accounting		**1208**	**W Springfield**	**333-7752**
Distribution Office		**1208**	**W Springfield**	**333-6760**
Systems Consulting		**1208**	**W Springfield**	**333-6133**
Statistical Services Consulting		**65**	**Comm West**	**333-2170**
Text Processing Consulting		**118**	**DCL**	**333-7318**
Terminal Repair Service		**150**	**DCL**	**333-0969**
CYBER Dial-up Numbers	**CYBER 175**	**110-300**	**baud**	**333-4000**
	CYBER 175	**1200**	**baud**	**333-4001**
	CYBER 174	**110-300**	**baud**	**333-4004**
Asst Dir User Services	Robert Penka	173	DCL	333-4709
Asst Dir Systems and Operations	Sandra Moy	177	DCL	333-4703
Asst Dir Development	J. M. Randal	120	DCL	333-9772
Asst Dir Engineering and Hardware Selection	Cliff Carter	195	DCL	333-3723
Manager Accounting Services	Gary Bouck	1208	W Springfield	333-7752
Manager Statistical Services	Beth Richardson	150	DCL	
Documentation	Lynn Bilger	139	Astronomy	333-6236
CYBER-IBM Operations	Jack Knott	194a	DCL	333-6562
UNIX Operations	Debbie Hudson	123	DCL	333-8150
Telecommunications	Tom Kerkering	18	DCL	333-0816
Laboratory Support Project	Mike Gardner	18	DCL	333-7904
RJE Operations North	Rex Duzan	162	DCL	333-6285
RJE Operations South	Don McCabe	1208	W Springfield	333-2171 333-7752

RJE Sites (see Reference Guide RF-0.3 for operating hours)

Agriculture	M103	Turner Hall	333-8170
Chemistry	153	Noyes Lab	333-1728
Commerce West	70	Comm West	333-4500
CRH Snack Bar	120	Snack Bar	333-1851
DCL Routing Room	14	DCL	333-6203
Electrical Engineering	146	EEB	333-4936
Florida Ave Res Hall		FAR	333-2695
Illinois St Res Hall		ISR	333-0307
Mechanical Engineering	65	MEB	333-1430
Psychology	453	Psych Bldg.	333-7531
Social Science	202	Lincoln Hall	333-0309

OFF-LINE is the monthly newsletter of the Computing Services Office at the University of Illinois at Urbana-Champaign. Unless otherwise indicated, permission to reprint is freely granted, provided that the author, if named, and the Computing Services Office (CSO) are credited. Information in this issue is current as of April 28, 1982.

CSO operates a CDC CYBER 175 with 262K words of central memory and a CDC CYBER 174 with 196K words of central memory. The 175 and 174 run under the NOS Operating System and share 512K words of ECS. The 175 serves over 200 simultaneously active text and graphics terminals and the 174 serves over 100 simultaneously active terminals. CSO also operates an IBM 4341 with 4 million bytes of memory running HASP-OS/MVT under VM. In addition, CSO operates a DEC PDP-11/50 with 252K bytes of core driving a GSI CAT-8 phototypesetter and a DEC VAX 11/780 with 2 million bytes of core, both running under the UNIX Operating System.

Operating Hours (see HEARYE or SCHEDUL for exceptions):

	CYBERS 174/175	IBM
M-F	8 am to 6 am	8 am to 6 am
SAT	8 AM - Midnight	8 AM - 6 AM
SUN	Noon - 6 AM	Noon - 6 AM

POLICY

MANUALS TO GO ON SALE

As indicated in the April issue of *OFF-LINE*, CSO will begin selling some of the manuals formerly distributed free. Six manuals were removed from the list of free documentation on the basis of cost to CSO. Most of the manuals which will now be sold are elementary in nature. Some have been distributed at the rate of more than 3000 copies per year.

We have made arrangements with the Illini Union Bookstore, 715 South Wright Street, to sell these manuals individually. The manuals will be available in the Reference Section of the bookstore beginning June 10, 1982 (the start of summer session registration).

Because of space and manpower limitations, the CSO Distribution Center will only sell the *RNF Documentation - Tutorial, Macros and Reference* manual individually. The other five manuals will be shrink wrapped and sold only as a unit. Users are reminded that the Distribution Center handles no cash; manuals must be paid for by check.

The manuals affected and their prices are as follows:

	Illini Union Bookstore	*Distribution Center*
RNF Documentation - Tutorial, Macros and Reference	$4.00	$4.00
Introduction to the CYBER Systems	$2.00	*
A Tutorial Guide to the ICE Text Editor	$1.25	package
ICE Reference Manual	$3.25	price =
An Index to Software on the CYBER	$3.25	$11.00
CYBER FORTRAN Debugging	$1.25	*

Instructors who expect to recommend any of these manuals for use by their students are asked to inform the Illini Union Bookstore of that fact and the number of students who can be expected to buy the manual. This will ensure an adequate supply of manuals on the shelf.

PLOTTER AND DIABLO PRINTERS MOVED

The default plotter and the Diablo printers have been moved from DCL to other RJE sites. The new locations are more accessible to individuals restricted to a wheelchair or whose mobility is otherwise impaired.

One Diablo printer has been moved to the Mechanical Engineering RJE (Room 65 ME) for the convenience of users on the northern end of the campus. The second Diablo printer has been moved to the Lincoln Hall RJE (Room 202 LH) for those users on the southern end of the campus.

When you queue a job for printing on a Diablo printer, you are now asked to name the location (ME or LH) of the Diablo printer you wish to use.

The default plotter has been moved to 65 Mechanical Engineering. Other than retrieving plot output at the new location, no user changes are required by this move. The special plotters are still at DCL, and plot output from the special plotters should now be picked up in Room 123 DCL.

CYBER

KRANNERT EVENTS ARE NOW ON-LINE

Krannert Center announces that users of the CYBER systems may now obtain a listing of "This Week at Krannert" by entering the command:

```
GET,EVENTS/UN=3KRANNE
TYPE,EVENTS.
```

If you wish to print a copy of this file, simply replace the TYPE command with the PRINT command.

Also, messages may now be sent to Krannert Center via TELL,3KRANNE.

DO YOU USE HEARYE?

News of current or planned changes and/or problems is broadcast to the user community through HEARYE bulletins. The bulletins can be read by any user signed onto either CYBER system and are posted in printed form at the RJE sites.

Since HEARYE bulletins are available on-line, they represent the fastest method available to us for contacting our user community. There are no time delays introduced by reproducing or distributing the message. The system does, however, assume that users will take an active role, regularly checking for news bulletins.

In special circumstances we issue a banner at signon advising users to "TYPE HEARYE FOR NEWS." However, since we sometimes issue bulletins daily, we do this only in emergency situations so as not to dilute the impact of the banner.

Experience has taught us that many users never read HEARYE bulletins that directly affect them. The purpose of this article is to tell you about some of the useful forms of the HEARYE command. Secondly, we want to tell you how to package its use so that invoking it is convenient and automatic.

In the simplest case, you can read news bulletins by entering the command:

```
HEARYE.
```

All news bulletins will begin listing at your terminal, starting with the most recent and continuing until you press the S key or enter STOP to terminate the program. This form of the command requires only one step, but it gives you no control over what is listed. If you want to be more selective, enter the command:

```
HEARYE/F.
```

This command will list only headlines which accompany the news bulletins. Each headline will be preceded by the date on which the bulletin was issued and a keyword. The keyword can be specified on the HEARYE command to selectively retrieve the text of the news bulletin. For example, if the keyword RATES has been assigned to a headline which looks interesting, the command

```
HEARYE,RATES
```

can be used to retrieve the full text of the bulletin.

Finally, when ***/S=—nDAYS*** is appended to the command, all news bulletins except those issued in the last ***n*** days are ignored. For example:

```
HEARYE/S=—3DAYS
```

will list only news bulletins issued during the last three days. You can also use ***/S=—nDAYS*** together with either of the two alternate forms of the command noted above. For example:

```
HEARYE/F/S=—3DAYS
HEARYE/S=—3DAYS,RATES
```

To make checking for new HEARYE bulletins automatic, we suggest that you create a procedure file which is invoked at every signon. For those of you not familiar with the concept, a procedure file is a list of system commands stored in a file which can be executed as a group with a single command.

For example, suppose that you have created a permanent indirect access file named PROCFIL which contains the following record:

```
.PROC,START.
$BILL,chrg.proj.
$HEARYE/F/S=—3DAYS.
$REVERT.
```

Note that these lines must be placed in a separate section of file PROCFIL. (In the simplest case this means that the file contains only these lines.) Note also that the name PROCFIL was chosen purposefully. It must be used as the file name if the remainder of this article is to apply to your procedure file.

You can then invoke this procedure file at signon instead of issuing the usual BILL or CHARGE command. Invoking the procedure will result in the execution of the required BILL command followed by a HEARYE command which will list the headlines of all news bulletins issued during the last three days. If you regularly enter other commands at signon, you can add these to the procedure file as well.

To invoke the procedure file, enter the command:

```
BEGIN,START.
```

If you're at a time-sharing terminal, you can abbreviate this to

```
-START.
```

For the ultimate in abbreviation, if the above procedure file is the first or only record in the file PROCFIL, you can invoke it at signon on a time-sharing terminal by entering a dash, i.e.,

```
-
```

when responding to the prompt RECOVER/CHARGE.

Procedure files are a feature of the CYBER Control Language (CCL). CCL provides more elaborate facilities that you might want to investigate. You can, for instance, have more than one procedure in a file PROCFIL and can pass parameters to procedures at the time they are invoked. There are CCL statements which can make use of passed parameters to govern the sequence of statements executed. CCL is documented in the *NOS Reference Manual, Volume I,* Chapter 4.

MATHEMATICAL SERVICES

TRANSACTIONS ON MATHEMATICAL SOFTWARE

We now have on tape all algorithms published in *Transactions on Mathematical Software* up to December 1981 (from algorithm 493, published in the first issue of March 1975, up to algorithm 580). The algorithms from the March 1982 issue have been ordered, but have not yet arrived. The directory of algorithms in file TOMS/UN=MATHLIB has been updated. To print this directory, enter:

```
GET,TOMS/UN=MATHLIB.
PRINT,TOMS.
```

The "Math Note" series described in the March 1981 issue of *OFF-LINE* includes a note summarizing all algorithms published in TOMS to date.

To print this note (number 6), enter:

```
GET,MNOTES/UN=MATHLIB.
MNOTES,6.
PRINT,OUT/AS/CC/EJ.
```

(To print instructions on the use of MNOTES, substitute a 0 (zero) for 6 in the example above.)

The algorithms published in the December 1981 and March 1982 issues are as follows (the March algorithms have not arrived yet):

In the December issue:

#578: Solution of Real Linear Equations in a Paged Virtual Store
#579: CPSC: Complex Power Series Coefficients
#580: QRUP: A Set of FORTRAN Routines for Updating QR Factorizations

In the March 1982 issue:

#581: An Improved Algorithm for Computing the Singular Value Decomposition

The following articles which appeared in these issues may be of interest to some of our users:

In the December 1981 issue:

The Design of the XMP Linear Programming Library, by Roy E. Marsten (We have XMP on the CYBER.)

ME28: A Sparse Unsymmetric Linear Equation Solver for Complex Equations, by I. S. Duff (We have ME28 in our CYBER HARWELL library, under SHARED.)

In the March 1982 issue:

An Evaluation of Mathematical Software that Solves Systems of Nonlinear Equations, by K. L. Hiebert

LSQR: An Algorithm for Sparse Linear Equations and Sparse Least Squares, by Christopher C. Paige and Michael A. Saunders

An Improved Algorithm for Computing the Singular Value Decomposition, by Tony F. Chan (This describes Algorithm 580.)

If you are looking for software, *Transactions on Mathematical Software* is one of the journals to check, along with *Numerische Mathematik, BIT, Computer Journal,* and various publications of ACM (the Association for Computing Machinery) and SIAM (the Society for Industrial and Applied Mathematics).

ACSL VERSION 7A

The newest version of the ACSL package for continuous simulation has been obtained and installed under FUTURE on the CYBER systems. It will become the default version after June 1, 1982. Until then, it can be accessed by the command:

```
GRAB,ACSL/F.
```

The current version will be accessible after June 1, 1982 via the PAST command (GRAB,ACSL/P.) and will remain so until the fall semester begins.

The new version, designated 7A by the vendors Mitchell and Gauthier, includes the features listed below. A newsletter from Mitchell and Gauthier detailing these changes can be viewed in the Systems Consulting Office (1208 W. Springfield), or borrowed from Stan Kerr. A new manual has been prepared and has been ordered.

New features are:

- Terminal width control and high volume data flag

  New system variables have been added (accessible via SET run-time commands) for additional control over high-volume output (from PLOT and PRINT commands). They allow the user to require that such output be displayed on the terminal, and to control the assumed line width of the terminal.

- New PLOT subcommands and options

  New subcommands of PLOT allow specification of logarithmic axes and additional title information for each axis. A new system symbol (STRPLT) gives the option of producing stacked strip chart plots.

- Printer plot default grid

  The default grid separation in the Y direction on printer plots is now 10 instead of 20.

- Recovery from carriage return

  In the current ACSL, entering an empty line (just a carriage return) at run-time causes ACSL to terminate. In the new version, the user is prompted, instead, with the message "ENTER ACSL COMMAND".

- Linear analysis capability

  A facility for linear analysis, including detailed eigenvalue analysis of a system, has been added. (This facility has actually been in our current version for some time, but cannot be used due to an error in installation.) The facility is accessed via the run-time command ANALYZ, with various subcommands to control the exact type of analysis to be performed.

  The ANALYZ facility requires noticeably more memory to use, so it has been made

an option of the ACSLGO procedure. If the user wishes to use it, ACSLGO must be called with ANALYZ=YES. For example:

```
ACSLGO,PLOT=TEKTRON,ANALYZ=YES.
```

Questions or comments about the new version should be directed to Stan Kerr (333-4715, or TELL,UN=MATHLIB).

AMOSLIB, A LIBRARY OF SPECIAL FUNCTIONS

A library of special function routines, called AMOSLIB after its principal author Donald Amos of Sandia National Laboratories, has been installed on the CYBER systems. It includes Bessel functions I, J, K, Y, ber, bei, ker, kei, Airy functions Ai and Bi, the complex error function, iterated coerror function, inverse coerror function, log gamma function, incomplete gamma distribution, incomplete beta functions, sine and cosine integrals, and several statistical distributions. The library is accessed by the command:

```
GRAB,AMOSLIB.
```

Following this, you should compile and run a FORTRAN program which calls the AMOSLIB routines.

A short writeup, which includes a complete catalog of routines, can be obtained and printed as follows:

```
WRITEUP,AMOSLIB.
PRINT,AMOSDOC/AS/CC/EJ.
```

Reference Guide RF-4.28 contains a short description of AMOSLIB, and information on it is now in the system HELP files. Source and writeups of individual AMOSLIB routines can be obtained via the MATH procedure (see Reference Guide RF-4.12 MATH, or the system HELP file on MATH), as follows:

- To get a catalog of available routines (basically the same as that obtained through WRITEUP), enter:

```
GRAB,MATH.
MATH,AMOSLIB.
PRINT,DOC.
```

- To get source and a writeup for a specific routine, *xxx*, enter:

```
GRAB,MATH.
MATH,AMOSLIB,xxx.
```

  The source will be placed in local file SOURCE, the writeup in local file DOC.

The MATH procedure does not have a provision for getting a complete set of writeups for all routines; however, the following set of commands can be used to achieve this:

```
ATTACH,AMOSOPL/UN=MATHLIB.
MODIFY,P=AMOSOPL,Z,F,C=DOC,LO=E./*DEFINE,DOC
PRINT,DOC.
```

This will print about 3400 lines.

Questions about and problems with AMOSLIB should be directed to Stan Kerr (333-4715, or TELL,UN=MATHLIB).

IBM

SPIRES NOW AVAILABLE

SPIRES, a data base management system developed by Stanford University, has been acquired by CSO. The system is now running in a test environment on the IBM 4341 under CMS, an IBM time-sharing system.

We are waiting for a tape containing the next release of SPIRES before making the system available to the user community. SPIRES will be supported by dial-up and dedicated ports on the IBM system only.

The system will be limited in availability to hard money accounts, and to only those willing to make a commitment to fund the necessary disk space for an extended period. It will be necessary for CSO to purchase additional disks to meet these requirements, and a charge of $50.00 per million characters per year will be made to cover this cost. CMS usage will have a terminal connect time charge of $3.00 per hour and a CPU charge of $80.00 per hour.

The following excerpt from the preface of the SPIRES manual describes it in general terms:

> "SPIRES (Stanford Public Information REtrieval System) was developed to meet the need for a generalized data base management system -- one that could serve a bibliographic data base, a medical data base, or any of a variety of administrative and research applications. Since development costs are spread over its many users, a generalized system brings data base services within the reach of smaller applications.
>
> SPIRES differs substantially from most commercially available data base systems in other respects, too. Most systems are designed primarily for batch and online access from programs written in COBOL, PL/I, or Assembler language, and have only rudimentary self-contained command language capability for inquiry and searching. In contrast, SPIRES provides a rich command language for searching and updating both online and in batch and uses a host language interface as an auxiliary tool.

Another important SPIRES capability missing in most systems is file definition and creation by the users; SPIRES users may develop and administer their own data base applications without the constraints found in systems requiring centralized administration.

Finally, SPIRES is designed to handle bibliographic and text applications, as well as those oriented toward administration and management. The wide range of applications found in a university environment demands a system that employs generalized data structures and manipulative capabilities. At the same time, SPIRES, with its high-level command interface and its generality, may be less CPU-efficient than a set of host language programs carefully designed and optimized for a specific application. However, the rising costs of development and maintenance, as well as decreasing CPU costs, make SPIRES an attractive alternative, especially when professional programmers are not available or their time is limited, or when the size and scope of an application rule out developing a large, specialized system."

We expect to announce the availability of SPIRES to the user community by July 1, 1982. A demonstration for interested people will be arranged at that time.

FEATURE ARTICLES

FORTRAN FUTURES

This article is part of a series on the work of the FORTRAN Standards Committee X3J3 in producing the next revision of the FORTRAN standard. The reader is reminded that the features described in this article are not a part of any FORTRAN compiler currently available, but rather are proposed requirements for FORTRAN processors produced in the late 1980's through the mid 1990's. Although every effort has been made to accurately describe the current position of X3J3 on these matters, the development of a revision to the FORTRAN standard is an evolutionary process, and these proposals may be subject to refinement, revision, or even retraction. Comments on these proposals may be given to Kurt Hirchert of the CSO Systems Consulting staff, who is a member of X3J3.

One of the ways in which the 1978 FORTRAN standard differed significantly from the 1966 standard was in the area of input/output operations. Implementations of the 1966 standard frequently contained extensions in this area, and many of them were standardized in the 1978 standard. The next revision will contain additional extensions in this area, including both refinements of the material added in the 1978 standard and material that was omitted from the 1978 standard because time limitations did not allow for its adequate development.

One problem discovered in interpreting the 1978 standard was that people with systems where files are always read from the beginning tend to assume that such positioning is an intrinsic part of connecting a FORTRAN unit to a file (whether by preconnection or by the OPEN statement), while people with systems that allow prepositioning of files tend to assume that positioning is implementation dependent unless a REWIND statement is issued. This difference in assumptions can result in portability problems, so an additional file connection specifier has been added to the OPEN statement to make such assumptions explicit. In addition to values indicating positioning at the beginning and processor dependent positioning (to

allow prepositioning outside FORTRAN on systems with that capability), a value has also been defined which specifies that the file is to be positioned such that an immediately following WRITE statement has the effect of adding records to the end of the file. The latter specifier thus allows one to avoid having to read through an entire file in order to extend it.

An additional file connection specifier has been added to specify which operations are to be permitted on the file (i.e., reading, writing, or both). When the allowed operations are limited, FORTRAN processors should be able to produce better and more timely error messages and in some cases may also be able to process files more efficiently.

Another change stemming from interpretation of the 1978 standard relates to the detection of the end of file on a READ. The 1978 standard has both the END= specifier commonly implemented as an extension of the 1966 standard and a new IOSTAT= specifier, but both are described in terms of encountering something called an endfile record, which is guaranteed to be present only if an ENDFILE statement has been executed. In the next revision, changes will be made to the REWIND, BACKSPACE, and CLOSE statements (including implicit CLOSEs) to guarantee that any file produced by a FORTRAN program will have an endfile record.

One of the features omitted from the 1978 standard because of a lack of time to develop it was NAMELIST. This feature is a frequently implemented FORTRAN extension, but each vendor's version of NAMELIST tends to be slightly different. X3J3 is therefore developing a simplified name-directed input/output facility that should provide the useful functionality of NAMELIST without adopting any vendor's particular idiosyncrasies.

One of the more useful additions of the 1978 standard was the addition of list-directed output. The handling of character data under list-directed input/output was inconsistent, since input character values had to be enclosed in apostrophes, but output character values were written without apostrophes. This facility will be more symmetric in the next revision, since there will be a specifier to indicate whether output values are to be enclosed in apostrophes (or quotes) and it will be possible to read words not enclosed in apostrophes under list-directed input. (These rules will also apply to name-directed input/output.)

There have also been some improvements in the ways data can be formatted. As a convenience, it will now be possible to put a repetition factor on the "/" editing descriptor. Thus, it will be possible to convert a FORMAT like

```
100 FORMAT(A20/////////I10)
```

to

```
100 FORMAT(A20,10/,I10)
```

instead of

```
100 FORMAT((A20,10(/),I10))
```

Also, a variant on the E format descriptor has been adopted in which the exponent is always a multiple of three and the value of the "fraction" is more widely varied. This form of output is more in keeping with standards for publishing data expressed in metric units.

SALES - EXCHANGES - HELP WANTED

PROGRAMMER WANTED

Should have considerable experience in handling large data sets on magnetic tape. FORTRAN experience required. Experience with the IBM computer desirable. Work will include writing, sorting, and merging programs and documentation of tape files. Would need to be available for approximately 100 hours of work during summer 1982. Send credentials to Dr. K. Parkay, 261 Bevier Hall, 905 S. Goodwin Ave., Urbana, IL 61801; or telephone 333-2958, or leave a message at 333-2412.

PROGRAMMER/ANALYST

Programmer/analyst needed: 10 hours per week during the summer; half-time graduate or undergraduate assistantship beginning in the fall term. Applicants should be familiar with PASCAL, PL/1, and/or FORTRAN. Contact Susan Bonzi, Library Research Center, 333-1980.

HALF-TIME RESEARCH ASSISTANT OR HOURLY PROGRAMMER

For preparation of data extracts, statistical programming, and management of large social science datasets and number of files on tapes and disk. May involve cleaning and creating variables from a new dataset on tape. Must document computer work carefully. Knowledge of SAS, SPSS, and FORTRAN preferred.

(1) 20/hours/week on average, Summer 1982, and/or 10-20 hours/week, Academic Year 1982-83, depending upon your availability.

(2) A second job could be of limited term and begin immediately for an individual who knows SAS.

Economist in FACE. Contact Prof. A. Beller, 333-7257 0r 333-2412 (leave a message).

TYPIST NEEDED FOR DATA ENTRY

Needed: One or more persons with good typing skills and experience in data entry into computer data base Person(s) will enter data into Computing Services Office IBM computer; therefore, knowledge/experience with CMS editing procedures will be helpful or willingness to learn.

Computing
Services
Office

Off-Line

University of Illinois at Urbana-Champaign VOL. 10, NO. 6 June 1982

EDITOR: Lynn Bilger
PHONE: (217) 333-6236
150 Digital Computer Lab
1304 W. Springfield Ave.
Urbana, Illinois 61801

CSO DIRECTORY - STAFF AND SERVICES

Administrative

Director	George Badger	150	DCL	333-4103
Business Manager	Stanley Rankin	150	DCL	333-6530
Secretary	Joyce McCabe	150	DCL	333-1637

User Services and Hardware/Software Support

User Accounting		**1208**	**W Springfield**	**333-7752**
Distribution Office		**1208**	**W Springfield**	**333-6760**
Systems Consulting		**1208**	**W Springfield**	**333-6133**
Statistical Services Consulting		**65**	**Comm West**	**333-2170**
Text Processing Consulting		**118**	**DCL**	**333-7318**
Terminal Repair Service		**150**	**DCL**	**333-0969**
CYBER Dial-up Numbers	**CYBER 175**	**110-300**	**baud**	**333-4000**
	CYBER 175	**1200**	**baud**	**333-4001**
	CYBER 174	**110-300**	**baud**	**333-4004**
Asst Dir User Services	Robert Penka	173	DCL	333-4709
Asst Dir Systems and Operations	Sandra Moy	177	DCL	333-4703
Asst Dir Development	J. M. Randal	175	DCL	333-9772
Asst Dir Engineering and Hardware Selection	Cliff Carter	195	DCL	333-3723
Manager Accounting Services	Gary Bouck	1208	W Springfield	333-7752
Manager Statistical Services	Beth Richardson	189	DCL	333-6276
Documentation	Lynn Bilger	120	DCL	333-6236
CYBER-IBM Operations	Jack Knott	194a	DCL	333-6562
UNIX Operations	Debbie Hudson	123	DCL	333-8150
Telecommunications	Tom Kerkering	18	DCL	333-0816
Laboratory Support Project	Mike Gardner	18	DCL	333-7904
RJE Operations North	Rex Duzan	162	DCL	333-6285
RJE Operations South	Don McCabe	1208	W Springfield	333-2171 333-7752

RJE Sites (see Reference Guide RF-0.3 for operating hours)

Agriculture	M103	Turner Hall	333-8170
Chemistry	153	Noyes Lab	333-1728
Commerce West	70	Comm West	333-4500
CRH Snack Bar	120	Snack Bar	333-1851
DCL Routing Room	14	DCL	333-6203
Electrical Engineering	146	EEB	333-4936
Florida Ave Res Hall		FAR	333-2695
Illinois St Res Hall		ISR	333-0307
Mechanical Engineering	65	MEB	333-1430
Psychology	453	Psych Bldg.	333-7531
Social Science	202	Lincoln Hall	333-0309

OFF-LINE is the monthly newsletter of the Computing Services Office at the University of Illinois at Urbana-Champaign. Unless otherwise indicated, permission to reprint is freely granted, provided that the author, if named, and the Computing Services Office (CSO) are credited. Information in this issue is current as of May 26, 1982.

CSO operates a CDC CYBER 175 with 262K words of central memory and a CDC CYBER 174 with 196K words of central memory. The 175 and 174 run under the NOS Operating System and share 512K words of ECS. The 175 serves over 200 simultaneously active text and graphics terminals and the 174 serves over 100 simultaneously active terminals. CSO also operates an IBM 4341 with 4 million bytes of memory running HASP-OS/MVT under VM. In addition, CSO operates a DEC VAX 11/780 with 4 million bytes of memory, running under the UNIX Operating System and driving a GSI CAT-8 phototypesetter.

Operating Hours (see HEARYE or SCHEDUL for exceptions):

	CYBERS 174/175	IBM
M-F	8 am to 6 am	8 am to 6 am
SAT	8 AM - Midnight	8 AM - 6 AM
SUN	Noon - 6 AM	Noon - 6 AM

POLICY

CSO SHORT COURSES - SUMMER SEMESTER

CSO is offering the following short courses during the summer semester 1982 to acquaint people with our facilities and the Control Data Corporation's (CDC) CYBER 175 and CYBER 174 computer systems.

To register for a course:

> please come in person to Room 150 DCL, or
> phone 333-6630.

Registration is free and limited to 30 people in some classes. If you find that all of the available classes on a topic are full, please leave your name on our waiting list. We will call you if someone drops a class.

> ***PLEASE NOTE THAT YOU MAY NOT REGISTER FOR A CLASS UNTIL THE WEEK PRECEDING THE WEEK THE CLASS IS TO BE TAUGHT!***

As announced in the April and May issues of *OFF-LINE,* some of the manuals formerly distributed free by CSO are now being sold. The following manuals are required for some of the courses listed below. These manuals may be purchased at the Illini Union Bookstore (Reference Section), 715 South Wright Street.

- Introduction to the CYBER Systems $2.00
- A Tutorial Guide to the ICE Text Editor $1.25
- RNF Documentation - Tutorial, Macros and Reference $4.00

The classes offered are as follows:

INTRODUCTION TO THE CYBER COMPUTER SYSTEM

This course is intended for the first-time CYBER computer user. The emphasis is on time-sharing usage and the ICE text editor; batch usage will also be discussed. Required manuals: *Introduction to the CYBER Systems* and *A Tutorial Guide to the ICE Text Editor.* Four sessions are offered:

June 21, 22, 23, 24, 25	12 noon - 1 PM	115 DCL
June 21, 22, 23, 24, 25	3 PM - 4 PM	115 DCL
June 28, 29, 30, July 1, 2	3 PM - 4 PM	115 DCL
July 12, 13, 14, 15, 16	12 noon - 1 PM	115 DCL

INTRODUCTION TO STATISTICAL COMPUTING

The purpose of this course is to equip the novice computer user or the newcomer to this campus with the basic knowledge necessary to utilize the CSO computing resources to perform statistical analyses. Topics covered include: an overview of the CSO system and the statistical software available, file concepts, program and data entry using the ICE text editor, a general overview of the four major packages available (SPSS®, SAS, SOUPAC, and BMDP), and how to run basic statistics using each of these packages. Required manuals: *Introduction to the CYBER Systems* and *A Tutorial Guide to the ICE Text Editor.* The course will consist of three 2-hour lectures and two 2-hour labs. One session is offered:

July 9, 12, 14 (Lectures)	1 PM - 3 PM	160 Commerce West
July 10, 17 (Labs)	9 AM - 11 AM	70 Commerce West

BASIC CONCEPTS IN COMPUTER INFORMATION PROCESSING

An introductory survey of computer information processing concepts and terminology, intended especially for those thinking of acquiring a small computer or word processing equipment. The needs of UI nonacademic and clerical personnel have been central to the planning of this course. (Offered jointly with the campus Office of Administrative Information Systems and Services.) One session is offered:

June 21,22,23,24	3 PM - 4 PM	156 Admin Bldg

INTRODUCTION TO RNF

This is a beginning-level discussion of the RNF text formatter used in the preparation of letters, manuals, or other documents. Topics will include tabbing, margins, spacing, paragraphing, and justification. Familiarity with the CYBER and ICE is assumed. Required manual: *RNF Documentation - Tutorial, Macros and Reference.* Two sessions are offered:

June 28, 30, July 2	12 noon - 1 PM	115 DCL
July 12, 14, 16	12 noon - 1 PM	201 Astronomy

USE OF MANAGE

How an instructor or project director allocates funds and computer resources to individual users. Assumes familiarity with simple use of the CYBER. One session is offered:

June 22	12 noon - 1 PM	237 DCL

INTRODUCTION TO SPSS® (Statistical Package for the Social Sciences)

This is an introductory course on the SPSS statistical package on the CYBER. Topics to be covered include: preparation of data for input, basic components of the SPSS® language, running programs from cards or at a terminal, using SPSS® to modify the data, obtaining basic

statistics and crosstabulations. Assumes: Introduction to Statistical Computing or equivalent knowledge. One session is offered:

July 19, 21	3 PM - 5 PM	241 Commerce West
A 2-hour lab session to be arranged		70 Commerce West

INTRODUCTION TO BMDP (Biomedical Statistical Package)

BMDP, a widely-used package of statistical programs, is oriented toward the biological, medical, agricultural and veterinary sciences. This is an introductory course on the use of BMDP on the CYBER. Topics covered include: data preparation, elements of the BMDP language, running BMDP programs from cards or at a terminal, data editing, creation and use of system files, and examples of using BMDP for descriptive statistics, regression and analysis of variance. The course consists of two lectures and a lab session. Assumes: Introduction to Statistical Computing or equivalent knowledge. One session is offered:

July 26, 28	3 PM - 5 PM	241 Commerce West
A 2-hour lab session to be arranged		70 Commerce West

EASY GRAPHING

Elementary presentation of a high-level interactive plotting program for X-Y plots, bar and pie charts. Its English-like commands require no programming experience to generate plots. Assumes familiarity with CYBER. One session is offered:

July 12, 13, 14	12 noon - 1 PM	239 DCL

GRAPHICS COMPATIBILITY SYSTEM (GCS)

Discusses producing plots on various graphics devices. Familiarity with the CYBER and FORTRAN is assumed. One session is offered:

July 19, 20, 21, 22	3 PM - 4 PM	239 DCL

USING MAGNETIC TAPES

Covers the use of magnetic tapes on the CYBER. Assumes familiarity with the CYBER. One session is offered:

June 28, 30, July 2	12 noon - 1 PM	239 DCL

MATHEMATICAL LIBRARIES

An overview of the various mathematical packages and libraries available on the CYBER systems. Assumes a working knowledge of FORTRAN and familiarity with the CYBER. One session is offered:

July 6, 7, 8	12 noon - 1 PM	237 DCL

ORDINARY LEAST SQUARES REGRESSION IN SOUPAC

A brief presentation on the use of the OLS procedure in the SOUPAC statistical package. Assumes an elementary knowledge of simple regression and SOUPAC. One session is offered:

June 22	12 noon - 1 PM	226 Commerce West

CONTROL LANGUAGE FOR USING SAS (Statistical Analysis System)

A brief presentation of the IBM Job Control Language needed to run SAS jobs, and of the most convenient CYBER Control Language commands for running SAS jobs from time-sharing terminals. Assumes: Introduction to Statistical Computing or the equivalent. One session is offered:

June 29	12 noon - 1 PM	226 Commerce West

CSO VIDEOTAPES AVAILABLE

CSO has produced a series of eight videotapes which introduce the novice to the CYBER Systems. A viewing guide containing the major pictorials used in the series is available and can be used to facilitate note taking.

The title and a brief synopsis of each of the videotapes is given below. Running time is 10-15 minutes for each videotape.

Introduction to Computing at CSO

A brief look at the steps required to solve a problem using a computer and some of the hardware used.

Using a Terminal

A description of the physical operation of a terminal and some of the keys that have a special meaning to the CYBER.

Introduction to CYBER Time-Sharing

A tutorial for logging on and off the CYBER.

File Usage - Local File and Indirect Access to Permanent Files

An introduction to CYBER files and the commands used to manipulate them.

Introduction to ICE Text Editing

A tutorial on entering and modifying files with ICE.

Running a FORTRAN Program - Concepts

The concepts of compilation, loading and execution.

Running a FORTRAN Program - The PROGRAM Statement

The PROGRAM statement and its relationship to files accessed by the program.

Running a FORTRAN Program - Control Statement

The control statements used to compile, load and execute a FORTRAN program.

Anyone can view these videotapes by going to the Undergraduate Library in person to make a reservation for use of the videotape equipment. Ask for a copy of the viewing guide when you check out the videotape for viewing.

CYBER

NEW VERSION OF EASY GRAPHING

A new version of the popular Easy Graphing program is available via:

```
GRAB,EZGRAPH/FUTURE.
```

This version incorporates a number of long desired, additional features. The new features include:

- A complete on-line HELP facility. The command

  ```
  HELP
  ```

  will list all available commands and other useful information such as command syntax.

HELP is also available for each command. For example,

```
HELP,ENTER
```

will give information on the ENTER command similar to that found in the *Users Reference Guide.*

- A new command, **LINE**, allows you to select the line style for each curve, or to reset the default line style. In addition to the six standard line styles, a new style, number **0** (zero), specifies that no line be drawn. When used with a symbol, scatter plots can now be produced. For example,

```
LINE 1,0
SYMBOL 1,1
```

plots a triangle at each data point, but no line connecting them.

- A new symbol, number **7**, is a very small marker to provide a "point" symbol for scatter plots.

- The command **SAVE**, with no file name specified, will write the "save" information to the terminal.

- When listing the values of a variable with the **LIST,var** command, one can now terminate the listing at the end of any page by typing **Q** in response to the prompt

```
"TYPE <CR> TO CONTINUE OR Q TO QUIT"
```

- Because of the LINE command, LIST can no longer be abbreviated to LI, but can be abbreviated to LIS.

- For users who own Tektronix 4662 plotters, Easy Graphing now should support these plotters better, in block mode.

- There is no longer a limit of four files. The number of files can increase dynamically until your memory limit (Field Length or FL) is encountered.

- There are now three optional parameters on the Easy Graphing control card. The default integer storage area size is 600 words and the real storage area size is 200 words. These values may be increased, though not decreased, with the **I=** and **R=** control statement options. For example,

```
EZGRAPH,I=1000,R=1000
```

requests that 1000 words be allocated for the integer storage area, and 1000 words for the real storage area.

The third parameter, **C=**, specifies the minimum amount of dynamic memory to reserve. This parameter is normally not necessary, as Easy Graphing will simply

acquire dynamic memory until your limit is reached. Dynamic memory is allocated for each file used. The default value is 3072 words, which is enough for four files. This value should be increased by about 1000 words for each file you plan to use beyond four files. For example, if you plan to use eight files, use the command:

```
EZGRAPH,I=1000,C=7072
```

- The new command, **MEMORY**, will report back the amount of unused memory in the integer and real storage areas.
- The command 4014 has been renamed to **TEK4014**.

This version of Easy Graphing will remain as the /FUTURE version until the installation of the new operating system planned for this fall.

NEW ZETA PLOTTER SOFTWARE

Two products from ZETA are available on the CYBERS. The Functional Subroutine library includes subroutines to perform the following:

- Arrows and dimension lines
- Logarithmic plots
- Hatching (shading) a polygonal area
- Smoothing a line

The TYPSET subroutines are used to plot high quality, graphics arts characters. In all, 19 alphabets and 11 symbol sets can be accessed in a manner similar to the standard ZETA calls to SYMBOL and NUMBER.

The Functional Subroutines have been incorporated into the standard ZETA library and are accessed with the command:

```
GRAB,ZETA.
```

The TYPSET subroutines are available in a separate library which is accessed with the command:

```
GRAB,TYPSET.
```

(Note: This command also accesses the ZETA library since TYPSET must be used in conjunction with the standard ZETA routines.) Enter one of the above GRAB commands before executing your program.

Both manuals are available for inspection in the Systems Consulting Office, 1208 W. Springfield. Copies will be available for purchase at the Distribution Office.

RUN A BATCH JOB - SAVE A SERVICE UNIT

Are you aware that it costs less to run a job in batch than in time-sharing? The cost reduction varies, depending on the amount of central memory used. Compared to its cost when run as a time-sharing job, a job requiring a small amount of core will cost about 10% less when run as a batch job. A job which uses the maximum possible core will cost about 40% less.

For many users, "batch" and "time-sharing" are elusive terms. Understanding is clouded by the fact that there is a "batch subsystem" available to time-sharing jobs and that a batch job can be submitted from a time-sharing job. We'll try to clear away some of the confusion.

The term "batch job" refers to a complete jobstream - job card, signon sequence, system commands, and necessary input - which is presented to the system to be processed at its convenience. The commands in the jobstream are executed in sequence. Input which would normally come from a terminal is read from the jobstream. Output which would normally go to a terminal goes, instead, to a file which is printed or routed to the FETCH queue (user's choice). The system uses batch jobs to fill idle moments, shunting them aside when a user at a time-sharing terminal enters a command. This accounts for the reduced cost.

A batch job can be submitted either from a card reader or from a time-sharing terminal. Since most batch jobs originate from a time-sharing terminal, we'll treat only this mode of submission.

To submit a batch job from a time-sharing terminal you must first create a file (we'll assume you've named it TINJOB) which contains the jobstream. Then you present the contents of the file to the system to be run as a batch job with the command:

```
SENDJOB,TINJOB.
```

In the simplest case, file TINJOB would contain, in sequence, two formatting directives interpreted by SENDJOB, a job statement, and system commands. For example:

```
/JOB
/NOSEQ
WIZARD.                          Job statement
SIGNON(123456789)                Signon
MYCODE.                          Password
BILL,CSO,PS3445.                 Accounting statement
PRINT/RJE=location.              Print output at 'location'
GET,TINMAN.                      Make program file local
FTN,I=TINMAN.                    Compile program (with listing)
LGO.                             Load and execute
```

This job would compile and execute the program found in file TINMAN and print the output

at the location specified on the PRINT command.

It is also possible to include program source and execution time input in the file along with the system commands. Each group of input must reside in a separate section of the file. The sections must be arranged in the same order in which they will be read by the job. An example will clarify this. Suppose you build a file containing the following lines (note that a line containing the characters /EOR has been entered in the file to separate sections):

```
/JOB
/NOSEQ
WIZARD.
SIGNON(123456789)
MYCODE.
BILL,CSO,PS3445.
PRINT/FETCH.                                    Route output to FETCH queue
FTN.                                            Compile
LGO.                                            Load and Execute
/EOR
      PROGRAM WIZARD(INPUT,OUTPUT,TAPE5=INPUT,TAPE6=OUTPUT)
      DO 10 K=1,3
      READ 100,I
100   FORMAT(I3)
      I = I * 4
10    PRINT 200,I
200   FORMAT(I4)
      END
/EOR
100
200
300
```

This file contains all the required system commands in the first section, the program source in the second, and the input required by the program in the third. When the job is processed, the system will read the first section and place it in a buffer from which the commands will be retrieved and executed. The jobstream is left positioned at the start of the second section. Since the FTN command doesn't name the file containing the program source, the system assumes a file name of INPUT. For batch jobs the file INPUT is equivalenced to the jobstream. The compiler reads the jobstream until it encounters an end-of-section, leaving the jobstream positioned at the start of the third section. Similarly, when the executing program reads from file input, the request is satisfied from the third section of the jobstream.

There are three pitfalls you should look out for. First, every command must begin in column 1. Second, every system command and the password in a batch job must end with a period or right parenthesis. Finally, the signon command must be preceded by a job statement. In its simplest form (used in the above examples), the job statement consists of 1 to 7 alphanumeric characters starting in column 1, followed by a period.

For more information about batch jobs, see the *NOS Reference Manual, Volume 1* and the description of the SENDJOB command contained in the *CSO Utilities Manual.*

NUMERICAL SERVICES

LSODE, A DIFFERENTIAL EQUATIONS LIBRARY

A library of routines for solving ordinary differential equations, known collectively as LSODE, has been installed on the CYBER systems. These routines were developed by Alan Hindmarsh and others at Lawrence Livermore Laboratory. The package is accessed on the CYBER systems by the command:

```
GRAB,LSODE.
```

This places the library into your local file space, and adds it to your "global library set"; when you then compile and run a FORTRAN program which calls routines in the library, the library is automatically searched for the routines being called, and they are loaded with your program.

The library consists of five principal routines for solving initial value problems (note that boundary value problems are not covered). They are briefly described below:

LSODE is the "basic" routine for solving a system of ordinary differential equations. It offers abilities very similar to those of routine DGEAR in the IMSL library. You can select whether a "stiff" or "nonstiff" method is to be used. (A stiff system of equations, roughly speaking, is one in which some components change very rapidly with respect to others; a nonstiff method, if given a stiff system to solve, will take very tiny integration steps in an effort to track the rapid variations, even when they are no longer significant.) There are options for selecting the technique used to handle the Jacobian of the system, which is the matrix of partial derivatives of the equations with respect to the unknowns.

LSODI is a variation of LSODE for handling linear implicit systems. Most differential solvers require you to formulate the system in the form

$$\frac{dy}{dt}=f(t,y)$$

where y is a vector of unknown functions and t is the independent variable. However, some systems are given in the form

$$f(t,y,\frac{dy}{dt})=0$$

and the equation cannot be solved for dy/dt in terms of t and y. Such systems are called implicit systems. LSODI attempts to solve a linear implicit system, with the form

$$A(t,y)*\frac{dy}{dt}=G(t,y)$$

where $A(t,y)$ is a square matrix which may at times be singular, depending on t and y. In this case, you must either supply the initial vector of values for dy/dt, or (if A is non-singular or invertible) let LSODI find the initial dy/dt. As with LSODE, you must select which type of integration method (stiff or nonstiff) is to be used.

LSODA is a variant of LSODE which automatically selects the type of integration method (stiff or nonstiff) dynamically, as the system is being integrated, depending on its behavior. You still have the option of selecting how the Jacobian is handled.

LSODAR is a variant of LSODA which not only selects the integration method, but also attempts to find roots of specified functions as the equations are being integrated. This might be used, for example, if you want the roots of a certain function, but it is defined by a differential equation.

LSODES is a variant of LSODE for large sparse systems, that is, systems consisting of a large number of differential equations with very little coupling between them, so that the Jacobian matrix has mostly zero entries. The Yale Sparse Matrix Package, which has been installed separately as a set of routines in UOILIB, is used to handle the sparse Jacobians.

A writeup on the LSODE library can be obtained and printed as follows:

```
WRITEUP,LSODE.
PRINT,LSODDOC/AS/CC/EJ/RJE=xxx
```

Where *xxx* is one of the RJE printing sites (see Reference Guide RF-0.3 Job Entry Sites (RJE) for a complete list of site codes). This prints about 50 pages, and consists of a full and detailed writeup on the basic routine LSODE, plus short descriptions for each of the other routines. (Including full writeups on all the routines would make it at least 200 pages long, which seemed unwarranted.) Full writeups on any one routine can be obtained via the MATH procedure, by entering

```
GRAB,MATH.
MATH,LSODE,routine
```

where *routine* is one of the routine names (LSODE, LSODI, LSODA, LSODAR, LSODES). This will give you two local files: SOURCE, containing complete source of the given routine and its associated routines; and DOC, containing a detailed writeup on the particular routine. The detailed writeups for each routine are generally about 20 pages long; the writeup consists of a brief description and overview, an illustrative example program, a detailed description of all the parameters of the routine and their meanings, and a description of various optional facilities which the routine may provide.

A full set of detailed writeups on all five LSODE routines can be obtained and printed as follows:

```
GRAB,MATH.
R.
MATH,LSODE,LSODE.
MATH,LSODE,LSODI.
MATH,LSODE,LSODA.
MATH,LSODE,LSODAR.
MATH,LSODE,LSODES.
PRINT,DOC/SEG/EJ/RJE=xxx.
```

The SAMPLES procedure has been updated to include sample programs for each of the LSODE routines. To obtain a list of the names of available samples, enter

```
GRAB,SAMPLES.
SAMPLES,LSODE.
```

and, to obtain a particular sample, enter

```
GRAB,SAMPLES.
SAMPLES,LSODE,sss
```

where *sss* is the name of the sample.

If you have any questions about the LSODE routines, please contact Stan Kerr (333-4715 or TELL,UN=MATHLIB on the CYBER systems).

ROUTINE LSODE IN UOILIB TO BE REMOVED

With the advent of the LSODE library, described in the above article, the subroutine named LSODE in the UOILIB library is no longer needed. It will remain in the library until July 1, at which time it will be removed.

FEATURE ARTICLES

FORTRAN FUTURES

This article is part of a series on the work of the FORTRAN Standards Committee X3J3 in producing the next revision of the FORTRAN standard. The reader is reminded that the features described in this article are not a part of any FORTRAN compiler currently available, but rather are proposed requirements for FORTRAN processors produced in the late 1980's through the mid 1990's. Although every effort has been made to accurately describe the current position of X3J3 on these matters, the development of a revision to the FORTRAN standard is an evolutionary process, and these proposals may be subject to refinement, revision, or even retraction. Comments on these proposals may be given to Kurt Hirchert of the CSO Systems Consulting staff, who is a member of X3J3.

In a language revision effort such as the one X3J3 has undertaken, one can usually classify most of the specific changes and additions as parts of collections of related changes. Almost inevitably, however, there will be changes which can only be described as miscellaneous. The topic of this article will be a few such miscellaneous changes which have been adopted by X3J3 for the next FORTRAN revision.

The FORTRAN 77 standard included an extension of the DATA statement to allow the use of implied DO loops to indicate which elements of an array are to be initialized. For the next revision, X3J3 has adopted a proposal to further extend the DATA statement to allow the index of an implied DO to also be used in character substring expressions, thus allowing character variables to be more easily initialized on a piecemeal basis. For example:

```
CHARACTER*120 STARS
DATA(STARS(I:I),I=1,120)/120*'*'/
```

Another evolutionary change comes in the area of default association of data types with variable names. In the original (1966) FORTRAN standard, the default association was always based on the rule that names beginning with I through N were integer and all others real. In FORTRAN 77, the IMPLICIT statement was standardized to allow the programmer to control the default rule. For the next revision, X3J3 has adopted the statement form

```
IMPLICIT NONE
```

as an indication that no default type association is to take place. In effect, this requires that the type of all variables, arrays, functions, etc. be specified explicitly so that the processor may treat references to unspecified names (such as might result from typographical errors) as compile-time errors rather than references to implicitly specified variables, arrays, functions, etc. which may not cause errors until load-time or run-time.

Perhaps the most significant of the miscellaneous changes described in this article is the relaxing of the restriction against the recursive use of FORTRAN functions and subroutines. The major barrier to inclusion of recursion in earlier versions of FORTRAN has been the fact that FORTRAN is normally implemented using only static storage, but the requirement to store information about a number of recursive instances of procedure which is unknown at compile-time tends to force at least limited dynamic storage allocation in a recursive environment. In the next revision, such dynamic storage is already required to support the array processing facilities, so this is no longer a barrier. For the most part, recursion has required little

in the way of new semantic rules, only clarification and extensions of the existing rules. For example, the SAVE statement, which in FORTRAN 77 indicates which variables and arrays are to retain their values in order to communicate among successive executions of a procedure, extends naturally to also provide communication among recursive executions of a procedure. Although recursion as a language feature now will impose little or no additional burden on FORTRAN run-time support, the execution of recursive routines may still be more expensive than the execution of their non-recursive counterparts on many implementations, so recursive procedure reference is limited to those procedures in which the keyword RECURSIVE appears as part of the FUNCTION or SUBROUTINE statement at the beginning of the procedure.

Recent Actions by X3J3

The 82nd meeting of X3J3 was held May 10-14, 1982 in Minneapolis, Minnesota. Actions taken at the meeting include the following:

- An "entity oriented" specification system was adopted as an alternative to the current FORTRAN specification system. This new system allows all the attributes of a variable, array, function, etc. to be specified in a single statement instead of requiring a separate statement for each attribute.

- The data structuring facility was extended to include variant data structure. Variant data structures are data structures in which the nature of some of the fields in the structure are dependent on the values stored in other fields.

- The committee adopted official responses to several requests for interpretation of the FORTRAN 77 standard.

- The committee officially revised the expected date of completion for the next revision to 1988.

- It was decided that the first attempt at describing the entire language permitted by the next revision would be made by merging the proposals already adopted into the text of the FORTRAN 77 standard.

- There was extensive discussion of the possible strategies for resolving several categories of name conflicts that may occur when writing programs using the features already adopted for the next revision.

- Tutorials were presented on the current views of the task group investigating means of adding error/exception/event handling to FORTRAN.

- Two extensions of the block-DO statement were considered. These proposals were prepared too late to be formally acted upon at this meeting, but informal votes suggest that they will likely be adopted at the August meeting.

- There was extensive discussion of possible means of reconciling the existing rules in FORTRAN for procedure argument association with the extensions which have been introduced, particularly in the area of array processing.

- There was extensive discussion on the feasibility and desirability of using arrays of single characters to replace the concept of character strings in FORTRAN. The replacement of varying length character strings by a structure containing a fixed length string (or array) and an integer to hold the length was the subject of similar discussions.

- Attempts to choose specific syntax for some of the previously adopted functions were unsuccessful because of a serious disagreement on the basic approach to be used in developing such syntax.

DOCUMENTATION

NEW AND REVISED REFERENCE GUIDES

The following Reference Guides are new:

RF-0.3a RJE Information for Handicapped

This guide provides information about which sites are accessible to handicapped persons; directions for wheelchair access routes, and hours of operation.

RF-0.11 LocalNet

This document describes a network called LocalNet that has been installed by CSO to provide a new method for connecting terminals on campus to CSO's computers via the campus coaxial cable.

RF-7.33 F45

This document contains information about F45 which is the FORTRAN Extended Version 4 to FORTRAN Version 5 Conversion Aid Program. It converts programs working under FORTRAN Extended Version 4 (FTN) into programs which will work under FORTRAN Version 5 (FTN5).

The following Reference Guides have been revised:

RF-0.1 Reference Guide List (5/21/82)
RF-0.2 Documentation List (5/21/82)
RF-0.9 TELENET (4/28/82)
RF-5.4 Easy Graphing (4/12/82)

OFF-LINE's Mailing List

If you wish to be placed on our mailing list for future issues of *OFF-LINE*, if you wish to be removed from the list, or if you wish to enter an address correction, please complete and return this page. (Current subscribers are kept on the mailing list until a specific request for removal is received, or until a mailing is returned as undeliverable.)

Check one:
- ☐ New subscriber
- ☐ Removal request
- ☐ Address correction

Name: _____

Address: _____

_____

_____

CAMPUS or Zip Code _____

(If address correction, give old address and zip code below.)

Comments:

SEND TO:

OFF-LINE
150 Digital Computer Laboratory
University of Illinois at Urbana-Champaign
1304 West Springfield Avenue
Urbana, Illinois 61801

Computing
Services
Office

Off-Line

University of Illinois at Urbana-Champaign VOL. 10, NO. 7 July 1982

EDITOR: Lynn Bilger
PHONE: (217) 333-6236
150 Digital Computer Lab
1304 W. Springfield Ave.
Urbana, Illinois 61801

CSO DIRECTORY - STAFF AND SERVICES

Administrative

Director	George Badger	150	DCL	333-4103
Business Manager	Stanley Rankin	150	DCL	333-6530
Secretary	Joyce McCabe	150	DCL	333-1637

User Services and Hardware/Software Support

User Accounting		**1208**	**W Springfield**	**333-7752**
Distribution Office		**1208**	**W Springfield**	**333-6760**
Systems Consulting		**1208**	**W Springfield**	**333-6133**
Statistical Services Consulting		**65**	**Comm West**	**333-2170**
Text Processing Consulting		**118**	**DCL**	**333-7318**
Terminal Repair Service		**150**	**DCL**	**333-0969**
CYBER Dial-up Numbers	**CYBER 175**	**110-300**	**baud**	**333-4000**
	CYBER 175	**1200**	**baud**	**333-4001**
	CYBER 174	**110-300**	**baud**	**333-4004**
Asst Dir User Services	Robert Penka	173	DCL	333-4709
Asst Dir Systems and Operations	Sandra Moy	177	DCL	333-4703
Asst Dir Development	J. M. Randal	175	DCL	333-9772
Asst Dir Engineering and Hardware Selection	Cliff Carter	195	DCL	333-3723
Manager Accounting Services	Gary Bouck	1208	W Springfield	333-7752
Manager Statistical Services	Beth Richardson	189	DCL	333-6276
Documentation	Lynn Bilger	120	DCL	333-6236
CYBER-IBM Operations	Jack Knott	194a	DCL	333-6562
UNIX Operations	Debbie Hudson	123	DCL	333-8150
Telecommunications	Tom Kerkering	18	DCL	333-0816
Laboratory Support Project	Mike Gardner	18	DCL	333-7904
RJE Operations North	Rex Duzan	162	DCL	333-6285
RJE Operations South	Don McCabe	1208	W Springfield	333-2171 333-7752

RJE Sites (see Reference Guide RF-0.3 for operating hours)

Agriculture	M103	Turner Hall	333-8170
Chemistry	153	Noyes Lab	333-1728
Commerce West	70	Comm West	333-4500
CRH Snack Bar	120	Snack Bar	333-1851
DCL Routing Room	14	DCL	333-6203
Electrical Engineering	146	EEB	333-4936
Florida Ave Res Hall		FAR	333-2695
Illinois St Res Hall		ISR	333-0307
Mechanical Engineering	65	MEB	333-1430
Psychology	453	Psych Bldg.	333-7531
Social Science	202	Lincoln Hall	333-0309

OFF-LINE is the monthly newsletter of the Computing Services Office at the University of Illinois at Urbana-Champaign. Unless otherwise indicated, permission to reprint is freely granted, provided that the author, if named, and the Computing Services Office (CSO) are credited. Information in this issue is current as of June 18, 1982.

CSO operates a CDC CYBER 175 with 262K words of central memory and a CDC CYBER 174 with 196K words of central memory. The 175 and 174 run under the NOS Operating System and share 512K words of ECS. The 175 serves over 200 simultaneously active text and graphics terminals and the 174 serves over 100 simultaneously active terminals. CSO also operates an IBM 4341 with 4 million bytes of memory running HASP-OS/MVT under VM. In addition, CSO operates a DEC VAX 11/780 with 4 million bytes of memory, running under the UNIX Operating System and driving a GSI CAT-8 phototypesetter.

Operating Hours (see HEARYE or SCHEDUL for exceptions):

	CYBERS 174/175	IBM
M-F	8 am to 6 am	8 am to 6 am
SAT	8 AM - Midnight	8 AM - 6 AM
SUN	Noon - 6 AM	Noon - 6 AM

POLICY

CSO SHORT COURSES - REMINDER

This is a reminder about the short courses being offered by CSO during the summer semester. To register for a course:

please come in person to Room 150 DCL, or
phone 333-6630.

Registration is free and limited to 30 people in some classes. If you find that all of the available classes on a topic are full, please leave your name on our waiting list. We will call you if someone drops a class.

PLEASE NOTE THAT YOU MAY NOT REGISTER FOR A CLASS UNTIL THE WEEK PRECEDING THE WEEK THE CLASS IS TO BE TAUGHT!

As announced in the April and May issues of *OFF-LINE,* some of the manuals formerly distributed free by CSO are now being sold. The following manuals are required for some of the courses listed below. These manuals may be purchased at the Illini Union Bookstore (Reference Section), 715 South Wright Street.

- Introduction to the CYBER Systems $2.00
- A Tutorial Guide to the ICE Text Editor $1.25
- RNF Documentation - Tutorial, Macros and Reference $4.00

The classes offered are as follows:

INTRODUCTION TO THE CYBER COMPUTER SYSTEM

This course is intended for the first-time CYBER computer user. The emphasis is on time-sharing usage and the ICE text editor; batch usage will also be discussed. Required manuals: *Introduction to the CYBER Systems* and *A Tutorial Guide to the ICE Text Editor.*

July 12, 13, 14, 15, 16	12 noon - 1 PM	115 DCL

INTRODUCTION TO STATISTICAL COMPUTING

The purpose of this course is to equip the novice computer user or the newcomer to this campus with the basic knowledge necessary to utilize the CSO computing resources to perform statistical analyses. Topics covered include: an overview of the CSO system and the statistical software available, file concepts, program and data entry using the ICE text editor, a general overview of the four major packages available (SPSS®, SAS, SOUPAC, and BMDP),

and how to run basic statistics using each of these packages. Required manuals: *Introduction to the CYBER Systems* and *A Tutorial Guide to the ICE Text Editor.* The course will consist of three 2-hour lectures and two 2-hour labs. One session is offered:

July 9, 12, 14 (Lectures)	1 PM - 3 PM	160 Commerce West
July 10, 17 (Labs)	9 AM - 11 AM	70 Commerce West

INTRODUCTION TO RNF

This is a beginning-level discussion of the RNF text formatter used in the preparation of letters, manuals, or other documents. Topics will include tabbing, margins, spacing, paragraphing, and justification. Familiarity with the CYBER and ICE is assumed. Required manual: *RNF Documentation - Tutorial, Macros and Reference.*

July 12, 14, 16	12 noon - 1 PM	201 Astronomy

INTRODUCTION TO SPSS® (Statistical Package for the Social Sciences)

This is an introductory course on the SPSS statistical package on the CYBER. Topics to be covered include: preparation of data for input, basic components of the SPSS® language, running programs from cards or at a terminal, using SPSS® to modify the data, obtaining basic statistics and crosstabulations. Assumes: Introduction to Statistical Computing or equivalent knowledge. One session is offered:

July 19, 21	3 PM - 5 PM	241 Commerce West
A 2-hour lab session to be arranged		70 Commerce West

INTRODUCTION TO BMDP (Biomedical Statistical Package)

BMDP, a widely-used package of statistical programs, is oriented toward the biological, medical, agricultural and veterinary sciences. This is an introductory course on the use of BMDP on the CYBER. Topics covered include: data preparation, elements of the BMDP language, running BMDP programs from cards or at a terminal, data editing, creation and use of system files, and examples of using BMDP for descriptive statistics, regression and analysis of variance. The course consists of two lectures and a lab session. Assumes: Introduction to Statistical Computing or equivalent knowledge. One session is offered:

July 26, 28	3 PM - 5 PM	241 Commerce West
A 2-hour lab session to be arranged		70 Commerce West

EASY GRAPHING

Elementary presentation of a high-level interactive plotting program for X-Y plots, bar and pie charts. Its English-like commands require no programming experience to generate plots. Assumes familiarity with CYBER. One session is offered:

July 12, 13, 14	12 noon - 1 PM	239 DCL

GRAPHICS COMPATIBILITY SYSTEM (GCS)

Discusses producing plots on various graphics devices. Familiarity with the CYBER and FORTRAN is assumed. One session is offered:

July 19, 20, 21, 22	3 PM - 4 PM	239 DCL

MATHEMATICAL LIBRARIES

An overview of the various mathematical packages and libraries available on the CYBER systems. Assumes a working knowledge of FORTRAN and familiarity with the CYBER. One session is offered:

July 6, 7, 8	12 noon - 1 PM	237 DCL

NETWORKING AT CSO

A group of people at CSO have been looking into the various aspects of networking in an attempt to identify the type of networking services that CSO should and could offer, and how our users could benefit from networking equipment as it becomes available. Certain networking applications are already in place, and others are under development or being planned. This article provides some background information on networking, describes some of the new services being offered, and briefly lists some plans for the future.

Background

Traditionally, connections among computers, between computers and terminals, and between computers and peripherals such as printers and plotters have been point-to-point. In order to establish point-to-point communications, the two devices are connected by some physical means, such as a wire or a telephone connection. If the user of a given terminal, which is connected in this way to one computer, wishes to use a different computer, either he must move to a different terminal, or the connection must be moved from the first to the second computer.

The idea behind networking is to allow communication between any two devices on the same network, without making any physical changes. This then allows for many possibilities, inlcuding:

- A terminal user may access any computer on the network from a single terminal.
- A user on any computer may print on any printer on the network.
- A file may be transferred from one computer to another on the network.

CSO has installed a broadband coaxial cable system over most of the University campus which is similar to the cable used for cable television. A broadband cable has many channels which can be used for separate, simultaneous, communications. The cable is currently used for

- physical plant monitoring of the environment in campus buildings
- television broadcast of messages about the status of the CSO computers to some of our job entry sites
- data transmission to and from CSO's job entry sites

Recent Developments

To take further advantage of the cable, we have acquired LocalNet network equipment from SYTEK, Inc. SYTEK supplies devices, which they refer to as Tboxes and Tmuxes, that allow terminals and computers to connect to the coaxial cable. A Tbox supports two connections or ports (e.g., two terminals) and a Tmux supports eight connections; otherwise the Tboxes and Tmuxes provide the same functions. In general, we will use the term Tbox to refer to either of these devices.

The use of LocalNet equipment provides us with these capabilities:

- transmission speed of 9600 baud
- error detection and correction performed by the hardware
- ability for any two Tboxes on the same channel to form the equivalent of a point-to-point connection
- flow control of data - such that if one device is sending data faster than the receiving device can handle it, the flow of data will be suspended until the receiving device is ready - without any loss of data
- files may now be transferred from one computer to another rather easily

CYBER Data Transfers

Many people who own minicomputers or microcomputers need to transfer files to and from the CYBER systems. There are problems associated with these types of transfers:

- The speed is limited to 300 or 1200 baud.

- There is no flow control on the CYBER. If you send data to the CYBER at full speed from, say, a microcomputer, some of the data can be lost.

To circumvent these problems, we have attached a PDP 11/40 to one of the CYBER channels. A microcomputer that can connect to the PDP 11/40 can do proper flow control to avoid lost data. If the access to the PDP 11/40 is via Tbox connections, there is also error checking and correction. This makes file transfers much easier at speeds of up to 9600 baud, and requires little or no programming on the non-CSO machines.

Current Services

We have been able to develop the following services, using the LocalNet equipment:

- DEC RT-11 floppy disk reading and writing. See the article "File Transfer Services" in this issue for more details.

- VAX to CYBER file transfer, whereby someone can request to have one or more files transferred from the CSO VAX (running UNIX) to the CYBER, or vice-versa. Call Debbie Hudson (333-8150) for more details.

- Self-service printer. We have had a self-service printer running at the Mechanical Engineering building during the spring semester for the use of Computer Science students on the CYBER system. Due, in part, to the successful use of this service, we intend to install at least one more self-service printer at another job entry site. We expect the primary use to be by CYBER users working at or near that site. However, the same printer can easily be used from any other computer on the network.

- Laboratory Data Transfer. Laboratories with their own computers that gather data now have the potential to transfer that data across the network to one of the CSO mainframes, such as the CYBER.

- Time-sharing access to all computers on the network. We now offer a limited number of time-sharing ports with access to the CSO computers; the VAX 11/780 running UNIX operating system and the CDC CYBER 175 running NOS operating system. We expect to also have access to the CDC CYBER 174 as soon as more equipment arrives.

Future Plans

There are a number of plans for the future. Some of these are long-range while others may be forthcoming in three to six months:

- A network control center to allow easier access to the network, which is to be provided by SYTEK.

- A high-speed computer to computer link, in the range of two megabits transfer rates.
- More self-service printers at job entry sites.
- Electronic mail transfer protocols among campus machines.
- More laboratory computer access for data transfers.
- More generalized microcomputer support for file transfers to the CYBER systems.
- Better tracking of files and programs as they pass through our systems.

Pricing

If you wish to have your own Tbox connection to the network, you can plan on spending $3000 per port (keep in mind that a Tbox has two ports and a Tmux has eight ports). This cost provides you with a connection to the network. You should generally have access to the mainframe of your choice; however, we cannot quarantee access at all times of the day. If the ports on the mainframe become busy too often, we will add more ports. Also, connect-time rates to the CYBER will be less than normal as a result of your purchase of a port. We have ordered additional SYTEK Tboxes and expect to have them available in the fall.

Information/Questions

We are very open to suggestions as to how to make the network work for you. If you have any questions about the network, how it may help you, or how you can connect to it, feel free to contact either Scott Lathrop, 193 DCL (333-6618), or Mike Gardner, 18 DCL (333-7904).

CYBER

FILE TRANSFER SERVICE

CSO is now offering a floppy disk and DEC tape file transfer service. Bring your floppy or DEC tape to Room 123 DCL between 9 AM - 4 PM, or call 333-8150 to make arrangements for transferring any portion of your floppy disk or DEC tape to your CYBER disk area. We can also move your CYBER disk files onto your floppy disk or DEC tape. Note, however, that the files must be character data rather than binary.

The mediums we support must be in DEC RT-11 format. We support:

- DEC Tape 2 (cassette tape)
- RXO2 floppy, single or double density (this is compatible with the IBM 2740 format)
- AED 6200P floppy
- RL02 hard disk (will eventually be available)

Note that RSX-11 systems can generate RT-11 format floppies.

At some time in the future, we will charge for this service. During our break-in period, it will be a free service.

If you have any questions, please contact:

Debbie Hudson
123 DCL
phone: 333-8150

APPLE FILE TRANSFER SOFTWARE

People who have an APPLE computer and wish to transfer files between the APPLE and the CSO CYBER systems may now obtain the necessary tools from CSO.

You must bring a blank floppy disk to:

Debbie Hudson
123 DCL
phone: 333-8150

She will copy our software onto your floppy disk. A detailed document is also available that describes the use of the software.

FEATURE ARTICLES

FORTRAN FUTURES

This article is part of a series on the work of the FORTRAN Standards Committee X3J3 in producing the next revision of the FORTRAN standard. The reader is reminded that the features described in this article are not a part of any FORTRAN compiler currently available, but rather are proposed requirements for FORTRAN processors produced in the late 1980's through the mid 1990's. Although every effort has been made to accurately describe the current position of X3J3 on these matters, the development of a revision to the FORTRAN standard is an evolutionary process, and these proposals may be subject to refinement, revision, or even retraction. Comments on these proposals may be given to Kurt Hirchert of the CSO Systems Consulting staff, who is a member of X3J3.

Although the FORTRAN standard describes the meaning of FORTRAN without reference to concepts like compile-time, link-time, or run-time, the proposals approved for the next revision of FORTRAN include a number of facilities which can be considered to be compile-time facilities.

The first of these was necessitated by the "core+modules" structure which has been adopted for the standard itself. Since the meaning of a program unit may depend not only on the "core" language, but also on one or more modules of additional language features or applications facilities, the language processor needs to be informed what modules a program unit is using. The USING statement has been adopted to supply this information. It will be usable to access both "official" modules described in a standard and "unofficial" modules, such as the extensions to FORTRAN provided by a particular vendor.

The second compile-time facility is a set of compile-time control structures which can be used for such tasks as controlling which of several variant versions of a section of code is to be compiled. Compile-time variables have also been provided as a means of directing these compile-time control constructs. Compile-time variables are also a more flexible alternative to the use of the PARAMETER statement as a means of providing symbolic names for constants. To make these compile-time facilities easy to recognize and understand, they have the same form as their execution-time equivalents except that they are preceded by one or more periods. For example, the production of debugging output in a program might be controlled by a compile-time variable:

```
.......SUBROUTINE SUB
.......LOGICAL DEBUG
! CHANGE DEBUG TO .TRUE. WHEN A DEBUGGING VERSION
! OF THIS ROUTINE IS DESIRED.
.......DEBUG=.FALSE.
.......IF(DEBUG)THEN
! COUNTER IS USED IN DEBUGGING MODE TO KEEP TRACK
! OF THE NUMBER OF SPECIAL CASES PROCESSED.
        INTEGER COUNTER
.......END IF
          .
          .
          .
.......IF(DEBUG)THEN
        COUNTER=0
.......END IF
```

```
          .
          .
          .
.......IF(DEBUG)THEN
          IF(X.LT.0.0)COUNTER=COUNTER+1
.......END IF
          .
          .
          .
.......IF(DEBUG)THEN
          PRINT *,"AT THE END OF STEP 3, COUNTER=",COUNTER
.......END IF
          .
          .
          .
          END
```

Note that when the compile-time variable DEBUG is set to .FALSE., none of the debugging code (in this case, the specification, computation, and printing of the variable COUNTER) is even compiled. The control structures with compile-time equivalents include the block-IF, the block-DO, and the block-CASE constructs.

The final facility in this category is far from complete. X3J3 is working on a standard FORTRAN macro facility. Current emphasis has been on statement-macros, i.e., macros which specify how what appear to be programmer-defined statement types are to be translated into standard FORTRAN. At this time, the portion of this facility which has been adopted includes only "statements" in which nothing is allowed to vary, i.e., pure text inclusion, but proposals to parameterize these statements are under development. In the slightly less immediate future, X3J3 is also expected to consider proposals for macros which specify replacements for syntactic entities which are only part of a statement, e.g., function references. It is hoped that these facilities will make it easier for FORTRAN programmers to "tailor" the language to specific application areas while retaining program portability.

MISCELLANEOUS

NATIONAL CENTER STUDYING APPLICATIONS OF MICROCOMPUTERS IN VOCATIONAL EDUCATION

The National Center for Research in Vocational Education at the Ohio State University is conducting an analysis of applications of microcomputers in Vocational Education. The purpose of this study, which is scheduled for completion in early 1983, is to gather information relative to microcomputer delivered instruction, microcomputer managed instruction, and the use of microcomputers as a tool within specific occupations.

The final project report is intended to aid vocational education leaders in dealing with the issues and concerns microcomputers pose for policy and planning. Individuals engaged in applying microcomputers at the local level are encouraged to submit information regarding their efforts for review by project staff. Please contact: Dr. Leonard Nasman, The National Center for Research in Vocational Education, 1960 Kenny Road, Columbus, Ohio 43210, phone: 800-848-4815, ext. 23.

SALES - EXCHANGES - HELP WANTED

PART-TIME FORTRAN PROGRAMMER

A position is available for the 1982-1983 school year as programmer for a research project designing new models and algorithms for clustering, logit and discriminant analysis, and related methods. An average of 10 hrs/week is expected, and the salary range is up to $11/hr. Prerequisites are (1) strong background in FORTRAN, (2) extensive familiarity with magnetic tape formats and facilities on IBM, CDC, and other systems, and (3) background in working on different computer systems. Also helpful would be (1) experience in managing large files, (2) acquaintance with combinatorial optimization, (3) familiarity with numerical methods and multivariate analysis. Send inquiries/applications (with a vita and phone number where you can be reached) to:

Phipps Arabie
Department of Psychology
University of Illinois
603 E. Daniel
Champaign, IL 61820

or use TELL,ARABIE on the CYBER 175. You will receive a reply in mid or late August.

COMPUTER PROGRAMMER NEEDED

Graduate student needed with programming experience to assist with development of inovative state-wide data base. Experience/interest in user friendly interfaces desired. Summer work available with option for 1/2-time assistantship for 1982-83. Contact Richard Zollinger or James White immediately, Institute of Government and Public Affairs; phone 333-3340.

OFF-LINE's Mailing List

If you wish to be placed on our mailing list for future issues of *OFF-LINE*, if you wish to be removed from the list, or if you wish to enter an address correction, please complete and return this page. **(Current subscribers are kept on the mailing list until a specific request for removal is received, or until a mailing is returned as undeliverable.)**

Check one:

- ☐ Place my name on mailing list
- ☐ Make the following corrections or changes
- ☐ Delete my name from mailing list

_____

First name -- Initial -- Last Name

UI Address:

_____

Department

_____

Room - Building

Off-campus Address:

_____

Organization or Company (if applicable)

_____

Street Address

_____

City -- State -- Zip Code

If address correction, give old address and zip code below.

_____

_____

_____

SEND TO:

OFF-LINE
150 Digital Computer Laboratory
University of Illinois at Urbana-Champaign
1304 West Springfield Avenue
Urbana, Illinois 61801

Computing
Services
Office

Off-Line

University of Illinois at Urbana-Champaign VOL. 10, NO. 8 August 1982

EDITOR: Lynn Bilger
PHONE: (217) 333-6236
150 Digital Computer Lab
1304 W. Springfield Ave.
Urbana, Illinois 61801

CSO DIRECTORY - STAFF AND SERVICES

Administrative

Director	George Badger	150	DCL	333-4103
Business Manager	Stanley Rankin	150	DCL	333-6530
Secretary	Joyce McCabe	150	DCL	333-1637

User Services and Hardware/Software Support

User Accounting		**1208**	**W Springfield**	**333-7752**
Distribution Office		**1208**	**W Springfield**	**333-6760**
Systems Consulting		**1208**	**W Springfield**	**333-6133**
Statistical Services Consulting		**65**	**Comm West**	**333-2170**
Text Processing Consulting		**118**	**DCL**	**333-7318**
Terminal Repair Service		**150**	**DCL**	**333-0969**
CYBER Dial-up Numbers	**CYBER 175**	**110-300**	**baud**	**333-4000**
	CYBER 175	**1200**	**baud**	**333-4001**
	CYBER 174	**110-300**	**baud**	**333-4004**
Asst Dir User Services	Robert Penka	173	DCL	333-4709
Asst Dir Systems and Operations	Sandra Moy	177	DCL	333-4703
Asst Dir Development	J. M. Randal	175	DCL	333-9772
Asst Dir Engineering and Hardware Selection	Cliff Carter	195	DCL	333-3723
Manager Accounting Services	Gary Bouck	1208	W Springfield	333-7752
Manager Statistical Services	Beth Richardson	189	DCL	333-6276
Documentation	Lynn Bilger	120	DCL	333-6236
CYBER-IBM Operations	Jack Knott	194a	DCL	333-6562
UNIX Operations	Debbie Hudson	123	DCL	333-8150
Telecommunications	Tom Kerkering	18	DCL	333-0816
Laboratory Support Project	Mike Gardner	18	DCL	333-7904
RJE Operations North	Rex Duzan	162	DCL	333-6285
RJE Operations South	Don McCabe	1208	W Springfield	333-2171 333-7752

RJE Sites (see Reference Guide RF-0.3 for operating hours)

Agriculture	M103	Turner Hall	333-8170
Chemistry	153	Noyes Lab	333-1728
Commerce West	70	Comm West	333-4500
CRH Snack Bar	120	Snack Bar	333-1851
DCL Routing Room	14	DCL	333-6203
Electrical Engineering	146	EEB	333-4936
Florida Ave Res Hall		FAR	333-2695
Illinois St Res Hall		ISR	333-0307
Mechanical Engineering	65	MEB	333-1430
Psychology	453	Psych Bldg.	333-7531
Social Science	202	Lincoln Hall	333-0309

OFF-LINE is the monthly newsletter of the Computing Services Office at the University of Illinois at Urbana-Champaign. Unless otherwise indicated, permission to reprint is freely granted, provided that the author, if named, and the Computing Services Office (CSO) are credited. Information in this issue is current as of July 18, 1982.

CSO operates a CDC CYBER 175 with 262K words of central memory and a CDC CYBER 174 with 196K words of central memory. The 175 and 174 run under the NOS Operating System and share 512K words of ECS. The 175 serves over 200 simultaneously active text and graphics terminals and the 174 serves over 100 simultaneously active terminals. CSO also operates an IBM 4341 with 4 million bytes of memory running HASP-OS/MVT under VM. In addition, CSO operates a DEC VAX 11/780 with 4 million bytes of memory, running under the UNIX Operating System and driving a GSI CAT-8 phototypesetter.

Operating Hours (see HEARYE or SCHEDUL for exceptions):

	CYBERS 174/175	IBM
M-F	8 am to 6 am	8 am to 6 am
SAT	8 AM - Midnight	8 AM - 6 AM
SUN	Noon - 6 AM	Noon - 6 AM

CYBER

TIMETABLE FOR IAF

Since late June the CYBER 174 has been running with IAF, a new version of the time-sharing monitor. When it was first made available in test mode to the user community, we announced our intention to install it as the production time-sharing system by fall. We have since decided to postpone its installation.

Instructors who had been advised that documentation distributed to students during the fall semester would have to be updated to reflect the change to IAF should note that changes will **not** be required. The system available to students during the entire fall semester will be identical to that in use last year.

The plan to install IAF this fall was motivated by a hope that it would solve the problem of lost lines of terminal output. This problem had been seen at relatively few CDC sites, but was becoming more prevalent locally as the load on the system increased.

The time-sharing monitor handles all terminal input and output. Since IAF is internally quite different than its predecessor TELEX, there seemed to be a good chance that the problem would not be duplicated there.

The present version of the operating system was distributed with both TELEX and IAF. When we installed it several years ago, we chose to stay with the familiar TELEX since IAF was quite new and unproven. The problem of lost terminal output had not yet occurred at that time.

Last spring we received a new version of the operating system which included a version of IAF that incorporated many bug fixes. It was not possible in the time remaining to install the entire new operating system by fall. As an alternative, we decided to graft the new IAF to the old operating system. After some initial problems of incompatibility between the old and new code, this proved to be workable. The system now running on the CYBER 174 is the result.

Shortly after IAF was running successfully we discovered, to our dismay, that the problem of lost lines of terminal output was still occurring with IAF. Since the problem previously had defied repeated attempts to solve it, the prospect of a solution by fall was not good.

Fortunately, at about the same time, CDC found that the problem was in an area of code common to both time-sharing monitors. They reported a fix to us which we installed. This fix, together with an additional modification developed locally, appears to have solved the problem.

Since IAF is not part of the solution, we decided to postpone its installation until the entire new version of the operating system released with it is ready for installation. This will eliminate the risk of incompatibility between old and new code. Although this risk appears small, there is little to be gained now by an early installation.

As implied above, we will eventually install IAF, probably next January. CDC has announced that they will no longer support TELEX, so the move is needed to assure continued vendor support. There are other benefits to be gained as well, such as freeing the CPU from some of the overhead of handling terminal communications.

We have continued to run IAF on the CYBER 174 so that both you and we can gain experience with it. We hope that many users will try it out and in so doing, ease the eventual transition. To encourage use, we are billing all CYBER 174 use at the weekend rate (40% discount) until August 23.

The external changes introduced by IAF aren't complex, but they are in areas which have become ingrained by habit. There is, for instance, a new way to cancel a line of terminal input or to interrupt an executing job. To obtain a four-page document describing all known changes introduced by IAF, enter the commands:

```
WRITEUP,IAF.
PRINT,IAFDOC/AS/CC/EJ/RJE=site.
```

where 'site' specifies the RJE site at which the document is to be printed. You should read this document before attempting to use IAF for the first time.

MATHEMATICAL SERVICES

ACSL RESTRICTED TO UNIVERSITY ACCOUNTS

Due to licensing restrictions on our copy of the ACSL simulation package, it will no longer be available on commercial accounts. Users of ACSL must have a University class account or a research board account. This change will be effective September 1, 1982.

NEW EDITION OF IMSL

CSO has received Edition 9 of the IMSL library and installed it on both the CYBER and IBM systems. Since there are many changes with this edition, we have installed it for now as a "future" version which will not become the default version until October 1, 1982. After this date, the current (edition 8) library will remain available as a "past" version throughout the fall semester. Detailed descriptions of the changes to the library are described below.

Until October 1, 1982, Edition 9 can be accessed as follows:

1. On the CYBER systems, by entering the command:

```
GRAB,IMSL/F.
```

or

```
FUTURE,IMSL.
```

2. On the IBM, by using:

```
LIBFILE='SYS9.IMSL'
```

for example:

```
// EXEC FORTLDGO,LIBFILE='SYS9.IMSL'
```

After October 1, 1982, the new IMSL will be the default, and the old IMSL will be accessible until the end of the semester as follows:

1. On the CYBER systems by entering:

```
PAST,IMSL.
```

2. On the IBM, by using:

```
LIBFILE='SYS8.IMSL'
```

Until October 1, writeups on the CYBER for IMSL Edition 9 can be obtained through the MATH procedure by adding FUTURE=YES to the call; for example:

```
MATH,IMSL,ZXSSQ,FUTURE=YES.
```

Owners of IMSL manuals are advised that there is an extensive set of updates for Edition 9 which can be ordered for $15.00. The manual has grown to four volumes now, so you may wish to order a volume 4 binder from IMSL for $6.00. The price of a complete manual, without binders, is now $38.00 for university customers. The IMSL manuals on display in the DCL user area, the Systems Consulting Office and the Statistical Consulting Office are being updated now to reflect Edition 9.

The following information has been provided by IMSL about the changes in Edition 9.

IMSL, Inc. announces the release of Edition 9 of the IMSL library. This set of FORTRAN routines, designed to be used in the development of scientific and engineering application programs, now consists of 517 subroutines covering the general fields of mathematics and statistics. This reflects 40 new routines and 19 deleted routines. The deleted routines are:

DBCEVU	-	replaced by DBCEVL
DBLINT	-	replaced by DBLIN
EQRH1F	-	deleted
FTCMP	-	replaced by FTCP
FTMPS	-	replaced by FTMA
FTMXL	-	replaced by FTML
FTTRN	-	replaced by FTTR
IBCEVU	-	replaced by IBCEVL
IBCICU	-	replaced by IBCCCU

NMKEN	-	replaced by NMKN
USCRDM	-	deleted
USHIST	-	replaced by USHST
USHIUT	-	replaced by USHST2
USHV1	-	replaced by USHHST
USPLT	-	replaced by USPLO
USPLTD	-	replaced by USPLOD
USRDM	-	deleted
USRDV	-	deleted
ZSYSTM	-	replaced by ZSPOW and ZSCNT

In addition, the following 19 chapter E routines have been reclassified as nuclei and are no longer documented in the reference manual: EBALAC, EBALAF, EBBCKC, EBBCKF, EHBCKF, EHBCKH, EHESSC, EHESSF, EHOBKS, EHOUSH, EHOUSS, ELRH1C, ELRH2C, HELZHC, ELZVC, EQRH3F, EQZQF, EQZTF, EQZVF.

Highlights of Edition 9 are described below.

BASIC STATISTICS

Two new routines for frequency tabulation have been added to Chapter B of the Library. One, BDTAB, tabulates counts for multivariate data and the other, BDTWT, forms two-way tables and computes the Pearson chi-squared statistic. Three other routines perform computations of statistics such as means, variances, covariances,skewness and kurtosis coefficients, and confidence limits. These new routines allow differential weighting of the observations; hence, the new routine for bivariate statistics, BECOVW, will simplify the computations for weighted regression, using the routines of Chapter R. These routines also allow specifications of missing values and provide user options for handling of observations with missing values. There are three easy-to-use routines for producing histograms and a new routine for producing probability plots for normal, lognormal, half-normal, exponential, Weibull, and extreme value distributions.

OTHER STATISTICAL PROCEDURES

A new routine in Chapter C is provided for life table analysis. The routine will produce the usual statistics for either a current life table or a cohort table. Friedman's tests and nonparametric multiple comparisons in a randomized complete block design are performed by a new routine in Chapter N. Some of the routines in Chapter F for time series analysis have been replaced by routines which are more stable in computing preliminary maximum likelihood estimates.

DIFFERENTIAL EQUATIONS, QUADRATURE, AND DIFFERENTIATION

A subroutine to solve a partial differential equation system of the form $u_t = f(x, t, u, u_x, u_{xx})$ using a collocation method combined with the method of lines (a modified version of DGEAR is used to solve the stiff ODE's) has been added. A boundary value problem solver, based on IMSL Advisor Victor Pereyra's work, which uses finite differences with deferred corrections, and a Gaussian integration routine designed for integrating multivariate functions over hyperrectangles are also featured additions. Other new subroutines integrate a bivariate

over non-rectangular regions and calculate derivatives of a user-supplied function. DGEAR has also been modified to handle banded Jacobians such as frequently arise in solving differential equations.

EIGENSYSTEM ANALYSIS

A new subroutine has been added to solve the generalized eigenvalue problem $Ax=\lambda Bx$ when A and B are symmetric and B is positive definite. This situation arises frequently in differential equation calculations.

RANDOM NUMBER GENERATION

There are new routines for generation of variates from a von Mises distribution for generation of a random correlation matrix, and for generation of a random contingency table with given row and column totals. This latter routine may be useful in estimation of probabilities of various table configurations.

INTERPOLATION, APPROXIMATION, SMOOTHING

The IMSL routine which calculates the coefficients of a bicubic spline has been replaced by a much more efficient routine written by IMSL Advisor Carl de Boor. Routine IQHSCV, for bivariate interpolation to scattered data, has been modified according to a suggestion by Richard Franks of the Naval Postgraduate School to use a better derivative estimation algorithm, resulting in improved accuracy.

LINEAR ALGEBRAIC EQUATIONS

Two new routines are designed to efficiently solve very large linear systems. Users in a virtual memory environment may choose LEQIF to drastically cut down on the number of page faults when solving a large system, while others may choose LEQOF to solve the system out of core.

SPECIAL FUNCTIONS

New routines for Weierstrass P-functions and their derivatives for both the lemniscatic and equianharmonic cases have been added to Chapter M. Seven new routines in Chapter M compute Bessel functions and modified Bessel functions of the first kind, and modified Bessel functions of the second kind. Other new routines compute exponential integrals of general order and incomplete elliptic integrals of the first, second, or third kind.

UTILITY FUNCTIONS

The Edition 9 printer plotting subroutines have a much simpler input sequence, particularly as regards the input of title and axis labeling information. New routined have been added for easy printing of complex vectors and matrices.

SORTING

Sorting a matrix using keys in the rows or columns is performed by four new routines in the

Edition 9 Library. Two of these routines are easy-to-use versions with very simple calling sequences for sorting columns only. The other two allow several options, such as sorting rows or columns by algebraic or by absolute value or by their l_1, l_2, or l_∞ norms, in ascending or descending sequence. In addition, they can keep track of the sort in a permutation vector which is returned to the user and/or provide an index of the distinct rows or columns. The option of leaving the original matrix unchanged (i.e., performing a detached key sort) is also available.

ZERO AND EXTREMA

A new routine for solving non-linear systems based on the MINPACK implementation of IMSL Advisor M.J.D. Powell's hybrid algorithm has been added to the Edition 9 Library. Another new subroutine from Advisor Powell finds a minimum of a single argument function. Both routines should work well both near and moderately far from the solution.

Another subroutine attempts to find the global minimum of a multivariate function in a hyperrectangle. This program uses ZSRCH to generate various initial values and implements a change of variables which converts all possible global minima (including those on the boundary) into local minima. ZXMIN is then invoked to find and compare the local minima of the now unconstrained problem.

FEATURE ARTICLES

FORTRAN FUTURES -- SUMMARY OF TOPICS

FORTRAN FUTURES has been appearing in *OFF-LINE* since last July. A summary of the topics covered by previous issues is listed below. If you are interested in a topic covered by an issue you missed, you may obtain back issues of *OFF-LINE* by contacting Lynn Bilger, 120 DCL (333-6236).

Vol. 9, No. 7 (July 1981) contains a brief history of FORTRAN standardization, a description of the basic organization and philosophy behind the work on the next revision of the FORTRAN standard, and a summary of the actions taken at the X3J3 meeting held May 11-15, 1981 in Toronto, Ontario, Canada.

Vol. 9, No. 8 (August 1981) contains a description of the enhancements to the next revision in the area of program form, including the elimination of column dependencies and the significance of blanks.

Vol. 9, No. 9 (September 1981) contains a description of enhancements in the area of control structures and a summary of the actions taken at the X3J3 meeting held August 10-14, 1981 in Los Alamos, New Mexico.

Vol. 9, No. 10 (October 1981) contains a description of the new facilities for data structuring.

Vol. 9, No. 11 (November 1981) contains the first article on array valued expressions and other new array processing features.

Vol. 9, No. 12 (December 1981) contains the continuation of the article on array processing and a summary of the actions taken at the X3J3 meeting held November 2-6, 1981 in Yorktown, New York.

Vol. 10, No. 1 (January 1982) contains a description of the new features supporting internal procedures and the sharing of variables, constants, etc. among multiple external procedures.

Vol. 10, No. 2 (February 1982) contains a description of the planned enhancements of the CHARACTER data type and a description of the BIT data type.

Vol. 10, No. 3 (March 1982) contains a description of the features intended to simplify the production of portable numeric software, including specification of precision requirements and numerical "environmental inquiry" intrinsic functions. It also contains a summary of the actions taken at the X3J3 meeting held February 8-12, 1982 in La Jolla, California.

Vol. 10, No. 4 (April 1982) contains a description of the features extending the procedure interface, including keyword specifications of arguments, optional arguments, and the specification of the intended usage of arguments.

Vol. 10, No.5 (May 1982) contains a summary of the planned input/output enhancements, including name-directed input/output, extensions to list-directed input/output, an "engineering" form of the E edit descriptor for "metric" output, new specifiers in the OPEN statement to control file positioning and limit allowed operations on the file, requirements for more consistent generation of endfile records, and an extension to allow the "/" edit descriptor to have a repetition factor.

Vol. 10, No. 6 (June 1982) contains the description of three "miscellaneous" features - an extension of the DATA statement, IMPLICIT NONE, and recursion. It also contains a summary of the actions taken at the X3J3 meeting held May 10-14, 1982 in Minneapolis, Minnesota.

Vol 10, No. 7 (July 1982) contains a description of the compile-time facilities to be added to FORTRAN, including a statement to specify what "modules" of the language are being used, a compile-time variable and control structure facility which allows variant versions of a program to be maintained using a single source file, and the beginnings of a standard FORTRAN macro facility.

MISCELLANEOUS

DIGITAL RAISES LABOR RATES

We have been informed of the recent rate changes by Steve Stultz, Field Service Unit Manager, Digital Equipment Corporation (Champaign Office).

Effective July 1, 1982, Digital's per call labor rates are increasing from $76.00 per hour to $83.00 per hour for time 8:00 am to 5:00 pm Monday through Friday, and from $90.00 per hour to $99.00 per hour for all other times. The mileage charge remains at $0.35 cents per mile.

Also the charge for personal computers and terminal service by Digital's terminal repairman remains the same at $63.00 per hour 8:00 am to 5:00 pm Monday through Friday,and $75.00 per hour for all other time.

SALES - EXCHANGES - HELP WANTED

PART-TIME FORTRAN PROGRAMMER

A position is available for the 1982-1983 school year as programmer for a research project designing new models and algorithms for clustering, logit and discriminant analysis, and related methods. An average of 10 hrs/week is expected, and the salary range is up to $11/hr. Prerequisites are (1) strong background in FORTRAN, (2) extensive familiarity with magnetic tape formats and facilities on IBM, CDC, and other systems, and (3) background in working on different computer systems. Also helpful would be (1) experience in managing large files, (2) acquaintance with combinatorial optimization, (3) familiarity with numerical methods and multivariate analysis. Send inquiries/applications (with a vita and phone number where you can be reached) to:

Phipps Arabie
Department of Psychology
University of Illinois
603 E. Daniel
Champaign, IL 61820

or use TELL,ARABIE on the CYBER 175. You will receive a reply in mid or late August.

PART-TIME PROGRAMMER

Part-time programmer to work on contractual basis with DEC system. **Must** know RT-11 operating system. Top dollars paid to qualified candidate. Contact:

John Quinn	or	Karl Newell
461 Freer Gym		Inst. for Child Behav.
Phone: 333-8561		333-6563

OFF-LINE's Mailing List

If you wish to be placed on our mailing list for future issues of *OFF-LINE*, if you wish to be removed from the list, or if you wish to enter an address correction, please complete and return this page. **(Current subscribers are kept on the mailing list until a specific request for removal is received, or until a mailing is returned as undeliverable.)**

Check one:

- ☐ Place my name on mailing list
- ☐ Make the following corrections or changes
- ☐ Delete my name from mailing list

_____

First name -- Initial -- Last Name

UI Address:

_____

Department

_____

Room - Building

Off-campus Address:

_____

Organization or Company (if applicable)

_____

Street Address

_____

City -- State -- Zip Code

If address correction, give old address and zip code below.

_____

_____

_____

SEND TO:

OFF-LINE
150 Digital Computer Laboratory
University of Illinois at Urbana-Champaign
1304 West Springfield Avenue
Urbana, Illinois 61801

Computing
Services
Office

Off-Line

University of Illinois at Urbana-Champaign VOL. 10, NO. 9 September 1982

EDITOR: Lynn Bilger
PHONE: (217) 333-6236
150 Digital Computer Lab
1304 W. Springfield Ave.
Urbana, Illinois 61801

CSO DIRECTORY - STAFF AND SERVICES

Administrative

Director	George Badger	150	DCL	333-4103
Business Manager	Stanley Rankin	150	DCL	333-6530
Secretary	Joyce McCabe	150	DCL	333-1637

User Services and Hardware/Software Support

User Accounting		**1208**	**W Springfield**	**333-7752**
Distribution Office		**1208**	**W Springfield**	**333-6760**
Systems Consulting		**1208**	**W Springfield**	**333-6133**
Statistical Services Consulting		**85**	**Comm West**	**333-2170**
Text Processing Consulting		**118**	**DCL**	**333-7318**
Terminal Repair Service		**150**	**DCL**	**333-0969**
CYBER Dial-up Numbers	**CYBER 175**	**110-300**	**baud**	**333-4000**
	CYBER 175	**1200**	**baud**	**333-4001**
	CYBER 174	**110-300**	**baud**	**333-4004**
Asst Dir User Services	Robert Penka	173	DCL	333-4709
Asst Dir Systems and Operations	Sandra Moy	177	DCL	333-4703
Asst Dir Development	J. M. Randal	175	DCL	333-9772
Asst Dir Engineering and Hardware Selection	Cliff Carter	195	DCL	333-3723
Manager Accounting Services	Gary Bouck	1208	W Springfield	333-7752
Manager Statistical Services	Beth Richardson	189	DCL	333-6276
Documentation	Lynn Bilger	120	DCL	333-6236
CYBER-IBM Operations	Jack Knott	194a	DCL	333-6562
UNIX Operations	Debbie Hudson	123	DCL	333-8150
Telecommunications	Tom Kerkering	18	DCL	333-0816
Laboratory Support Project	Mike Gardner	18	DCL	333-7904
RJE Operations North	Rex Duzan	162	DCL	333-6285
RJE Operations South	Don McCabe	1208	W Springfield	333-2171 333-7752

RJE Sites (see Reference Guide RF-0.3 for operating hours)

Agriculture	M103	Turner Hall	333-8170
Chemistry	153	Noyes Lab	333-1728
Commerce West	70	Comm West	333-4500
CRH Snack Bar	120	Snack Bar	333-1851
DCL Routing Room	14	DCL	333-6203
Electrical Engineering	146	EEB	333-4936
Florida Ave Res Hall		FAR	333-2695
Illinois St Res Hall		ISR	333-0307
Mechanical Engineering	65	MEB	333-1430
Psychology	453	Psych Bldg.	333-7531
Social Science	202	Lincoln Hall	333-0309

OFF-LINE is the monthly newsletter of the Computing Services Office at the University of Illinois at Urbana-Champaign. Unless otherwise indicated, permission to reprint is freely granted, provided that the author, if named, and the Computing Services Office (CSO) are credited. Information in this issue is current as of August 24, 1982.

CSO operates a CDC CYBER 175 with 262K words of central memory and a CDC CYBER 174 with 196K words of central memory. The 175 and 174 run under the NOS Operating System and share 512K words of ECS. The 175 serves over 200 simultaneously active text and graphics terminals and the 174 serves over 100 simultaneously active terminals. CSO also operates an IBM 4341 with 4 million bytes of memory running HASP-OS/MVT under VM. In addition, CSO operates a DEC VAX 11/780 with 4 million bytes of memory, running under the UNIX Operating System and driving a GSI CAT-8 phototypesetter.

Operating Hours (see HEARYE or SCHEDUL for exceptions):

	CYBERS 174/175	IBM
M-F	8 am to 6 am	8 am to 6 am
SAT	8 AM - Midnight	8 AM - 6 AM
SUN	Noon - 6 AM	Noon - 6 AM

POLICY

CSO SHORT COURSES - FALL SEMESTER

CSO is offering the following short courses during the 1982 fall semester to acquaint people with our facilities and the Control Data Corporation's (CDC) CYBER 175 and CYBER 174 computer systems.

To register for a course:

> please come in person to Room 123 DCL, or
> phone 333-8640.

Registration is free and limited to 30 people in some classes. If you find that all of the available classes on a topic are full, please leave your name on our waiting list. We will call you if someone drops a class. (At the time of this printing, rooms for some of the short courses had not yet been arranged. When you register for such a course, please ask for the room assignment.)

> ***PLEASE NOTE THAT YOU MAY NOT REGISTER FOR A CLASS UNTIL THE WEEK PRECEDING THE WEEK THE CLASS IS TO BE TAUGHT!***

As announced in the April and May issues of *OFF-LINE,* some of the manuals formerly distributed free by CSO are now being sold. The following manuals are required for some of the courses listed below. These manuals may be purchased at the Illini Union Bookstore (Reference Section), 715 South Wright Street.

- Introduction to the CYBER Systems $2.00
- A Tutorial Guide to the ICE Text Editor $1.25
- ICE Reference Manual $3.25
- RNF Documentation - Tutorial, Macros and Reference $4.00

BASIC CONCEPTS IN COMPUTER INFORMATION PROCESSING

An introductory survey of computer information processing concepts and terminology, intended especially for those thinking of acquiring a small computer or word processing equipment. The needs of UI administrative, clerical and nonacademic personnel have been central to the planning of the course. One session is offered:

Sept 27, 18, 19, 30, Oct 1 (Edwards-Iwe)	3:00 - 4:00 PM	192 Lincoln Hall

INTRODUCTION TO THE CYBER COMPUTER SYSTEM

This course is intended for the first-time CYBER computer user. The emphasis is on time-sharing usage and the ICE text editor; batch usage will also be discussed. Required manuals: *Introduction to the CYBER Systems* and *A Tutorial Guide to the ICE Text Editor.* Five sessions are offered:

Sept 13, 15, 17, 20, 22, 24 (Jones)	Noon - 1:00 PM	115 DCL
Sept 27, 28, 29, 30, Oct 1 (Primer)	8:00 - 9:00 AM	115 DCL
Oct 4, 6, 8, 11, 13, 15 (Pommert)	Noon - 1:00 PM	115 DCL
Oct 18, 20, 22, 25, 27, 29 (Moy)	8:00 - 9:00 AM	115 DCL
Nov 1, 3, 5, 8, 10, 12 (Scheid)	1:00 - 2:00 PM	115 DCL

INTERMEDIATE ICE (Illinois Central Editor)

A course on the use of the ICE text editor, covering techniques for solving a variety of editing problems: transfer of text lines, extended modify, case conversion, tab fields, windows, loops, stripe, procedures, plus others. Assumes familiarity with the CYBER and/or statistical packages. Required manual is the *ICE Reference Manual.* Time and room assignment to be arranged; check when you register. One session is offered:

Nov 8, 10, 12 (Dewan)

INTRODUCTION TO RNF

This is a beginning-level discussion of the RNF text formatter used in the preparation of letters, manuals, or other documents. Topics will include tabbing, margins, spacing, paragraphing, and justification. Familiarity with the CYBER and ICE is assumed. Required manual: *RNF Documentation - Tutorial, Macros and Reference.* Time and room assignment to be arranged; check when you register. Three sessions are offered:

Sept 27, 29, Oct 1 (Dewan)
Oct 18, 20, 22 (Dewan)
Nov 15, 17, 19 (Dewan)

ADVANCED RNF

A second course on the RNF text formatter covering the use of macros, variables and arrays in special applications such as hanging indents, formatted lists, tables of contents, page overflow control (headers and footers), superscripts and subscripts. Required manual: *RNF Documentation: Tutorial, Macros and Reference.* Time and room assignment to be arranged; check when you register. One session is offered:

Oct 25, 27, 29 (Dewan)

USE OF MANAGE

How an instructor or project director allocates funds and computer resources to individual users. Assumes familiarity with simple use of the CYBER. Time and room assignment to be arranged; check when you register. One session is offered:

Sept 14 (Primer)

SURVEY OF CSO GRAPHICS FACILITIES

An overview of CSO's graphics offerings. A brief description of the graphics hardware and software supported by CSO will be presented. Assumes familiarity with the CYBER. Time and room assignment to be arranged; check when you register. One session is offered:

Sept 16 (Tuchman)

NCAR GRAPHICS PACKAGE

An overview of the capabilities and use of the NCAR graphics software will be presented. NCAR provides device-independent graphics including contour plots, 3-D surfaces and map projections. Assumes a working knowledge of CYBER FORTRAN. Time and room assignment to be arranged; check when you register. One session is offered:

Sept 30 (Tuchman)

EASY GRAPHING

Elementary presentation of a high-level interactive plotting program for X-Y plots, bar and pie charts. Its English-like commands require no programming experience to generate plots. Assumes familiarity with CYBER. Time and room assignment to be arranged; check when you register. One session is offered:

Oct 4, 6, 8 (Tuchman)

GRAPHICS COMPATIBILITY SYSTEM (GCS)

GCS can be used to produce plots on various graphics devices. The fundamentals of using GCS will be presented. Familiarity with the CYBER and FORTRAN is assumed. Time and room assignment to be arranged; check when you register. One session is offered:

Oct 11, 13, 15, 18, 20, 22 (Lathrop)

USING MAGNETIC TAPES

Covers the use of magnetic tapes on the CYBER. Assumes familiarity with the CYBER. One session is offered:

Oct 26, 28, Nov 2, 4 (Wetzel) Noon - 1:00 PM 115 DCL

MATHEMATICAL LIBRARIES

An overview of the various mathematical packages and libraries available on the CYBER systems. Lecture 1 will cover the major subroutine libraries, such as IMSL; lecture 2 will cover software for linear equations and eigenanalysis; lecture 3 will cover differential equations and optimization. Assumes a working knowledge of FORTRAN and familiarity with the CYBER. Time and room assignment to be arranged; check when you register. One session is offered:

Oct 11, 13, 15 (Kerr)

THE SHAZAM ECONOMETRICS PACKAGE

SHAZAM is a comprehensive econometrics program with numerous special capabilities including restricted least squares, estimation of models with higher order autoregressive and moving average components, residual testing, distributed lags, nonlinear regression, and many others. One session is offered:

Nov 9, 11 (Jones) 7:00 - 8:00 PM 130 Comm West

USING SUBROUTINE LIBRARIES

How the user can create, use and modify his own library of special or frequently used subroutines. Time and room assignment to be arranged; check when you register. One session is offered:

Oct 12 (Hirchert)

FTN5 AND FTN TO FTN5 CONVERSON

Differences between CYBER FORTRAN IV (FTN) and the new FORTRAN 77 standard, as realized by CYBER FTN5. Converting FTN programs to FTN5. New features in FTN5. Assumes a good working knowledge of FTN. Time and room assignment to be arranged; check when you register. One session is offered:

Sept 21, 23 (Hirchert)

CYBER CONTROL LANGUAGE (CCL)

Using CCL to write procedures to be executed by the CYBER system. Time and room assignment to be arranged; check when you register. One session is offered:

Nov 16, 18 (Pommert)

INTRODUCTION TO STATISTICAL COMPUTING

The purpose of this course is to equip the novice computer user or the newcomer to this campus with the basic knowledge necessary to utilize the CSO computing resources to perform statistical analyses. Topics covered include: an overview of the CSO system and the statistical software available, file concepts, program and data entry using the ICE text editor, a

general overview of the four major packages available (SPSS®, SAS, SOUPAC, and BMDP), and how to run basic statistics using each of these packages. Required manuals: *Introduction to the CYBER Systems* and *A Tutorial Guide to the ICE Text Editor.* The course will consist of one required 3-hour lecture on the basics, followed by four optional 1½ hour lectures (each briefly describing one of the major statistical packages), and a 2-hour lab session covering basic file concepts and time-sharing terminal usage. Two sessions are offered:

Sept 13 (Lecture - Szoke)	6:30 - 9:30 PM	115 DCL
Sept 14 (SPSS)	6:30 - 7:50 PM	115 DCL
Sept 14 (SAS)	8:00 - 9:30 PM	115 DCL
Sept 15 (SOUPAC)	6:30 - 7:50 PM	115 DCL
Sept 15 (BMDP)	8:00 - 9:30 PM	115 DCL
Sept 18 (Lab session)	9:00 - 11:00 AM	70 Comm West
Oct 26 (Lecture - Kesner)	6:30 - 9:30 PM	115 DCL
Oct 27 (SPSS)	6:30 - 7:50 PM	115 DCL
Oct 27 (SAS)	8:00 - 9:30 PM	115 DCL
Oct 28 (SOUPAC)	6:30 - 7:50 PM	115 DCL
Oct 28 (BMDP)	8:00 - 9:30 PM	115 DCL
Oct 29 (Lab session)	9:00 - 11:00 AM	70 Comm West

INTRODUCTION TO SOUPAC

An introductory course on the SOUPAC statistical package on the CYBER. Topics covered: data preparation, structure of the SOUPAC language, data modification, running SOUPAC programs at terminals, and using SOUPAC for basic statistics, regression and analysis of variance. The course consists of two 3-hour lectures and one 3-hour lab. Prerequisite: Introduction to Statistical Computing or equivalent knowledge. One session is offered:

Sept 21, 23 (Mills)	6:30 - 9:30 PM	115 DCL
Sept 25 (Lab)	9:00 AM - Noon	70 Comm West

INTRODUCTION TO SPSS® (Statistical Package for the Social Sciences)

An introductory course on the SPSS statistical package on the CYBER. Topics to be covered include: preparation of data for input, basic components of the SPSS® language, running programs at a terminal, using SPSS® to modify data, obtaining basic statistics and crosstabulations. Course consists of two 3-hour lectures and one 2-hour lab. Prerequisite: Introduction to Statistical Computing or equivalent knowledge. Two sessions are offered:

Sept 27, 29 (Richardson)	6:30 - 9:30 PM	241 Comm West
Oct 2 (Lab)	9:00 - 11:00 AM	70 Comm West
Nov 2, 4 (Richardson)	6:30 - 9:30 PM	243 Comm West
Nov 6 (Lab)	9:00 - 11:00 AM	70 Comm West

INTRODUCTION TO SAS

An introductory course on the SAS statistical package on the IBM 4341. Topics covered: data

preparation, the SAS 'DATA' step, data modification, necessary control language for running SAS programs, and using SAS for basic statistics, regression and analysis of variance. The course consists of three 2-hour lectures and one 2-hour lab. Prerequisite: Introduction to Statistical Computing or equivalent knowledge. One session is offered:

Oct 7, 12, 14 (Woan)	7:00 - 9:00 PM	245 Comm West
Oct 9 (Lab)	9:00 - 11:00 Am	70 Comm West

INTRODUCTION TO SAS FOR SPSS® USERS

An introductory course on the SAS statistical package for current SPSS users who wish to learn the SAS language. The SAS language will be introduced and explained using SPSS terminology. Where possible, one to one correspondences between the languages will be provided. The course consists of two 2-hour lectures. Prerequisite: Advanced knowledge of SPSS language. One session is offered:

Nov 8, 10 (Mills)	7:00 - 9:00 PM	166 Comm West

INTRODUCTION TO BMDP (Biomedical Statistical Package)

BMDP, a widely-used package of statistical programs, is oriented toward the biological, medical, agricultural and veterinary sciences. This is an introductory course on the use of BMDP on the CYBER. Topics covered include: data preparation, elements of the BMDP language, running BMDP programs from cards or at a terminal, data editing, creation and use of system files, and examples of using BMDP for descriptive statistics, regression and analysis of variance. The course consists of two 2-hour lectures and a lab session. Prerequisite: Introduction to Statistical Computing or equivalent knowledge. One session is offered:

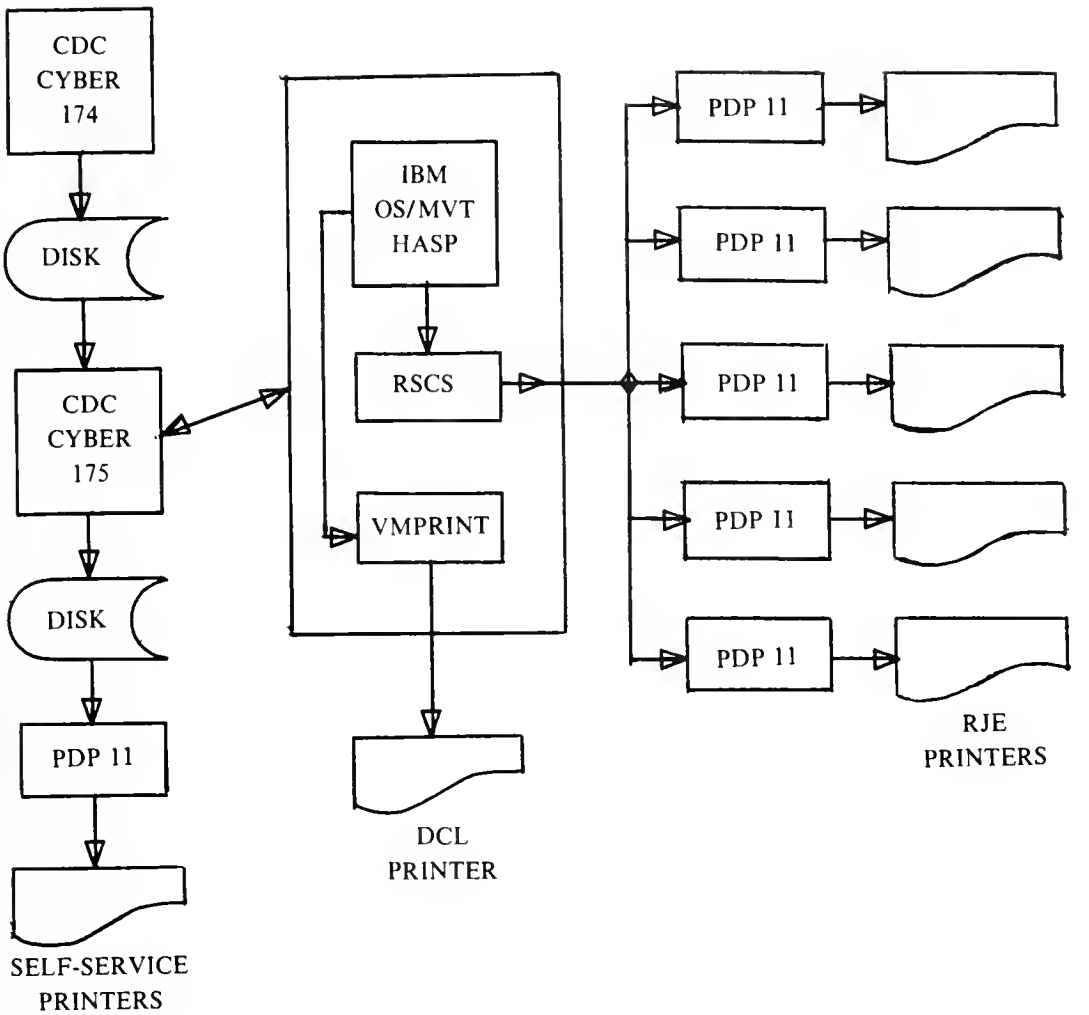
Nov 15, 17 (Szoke)	7:00 - 9:00 PM	166 Comm West
A 2-hour lab session to be arranged		

CYBER

LOCATING JOBS AND OUTPUT

The job enquiry commands QUERY and HASP (&D,I) are used to track the progress of jobs through the system. To use them effectively, you must know the appropriate use of each of these commands. As you read this article, take special note of how much each of these commands can "see", and of those times when neither command can find a job even though it is in the system. The accompanying diagram may help you picture what is being described.

Figure 1



On each of the computers, the operating system keeps track of the jobs entrusted to it by maintaining lists, called queues, of the names of jobs needing each kind of processing (e.g., execution, printing). It is important to note that each of the computers (CYBER 175, CYBER 174, IBM) maintains a separate set of queues and that the queues are accessible only to that computer. Each of the job enquiry commands works by searching through the queues that are accessible to it. The QUERY command has access to the queues maintained on the system that is processing the command, but not to queues maintained on the other systems. For example, a QUERY command issued on the CYBER 175 has access to queues on the CYBER 175, but not on the CYBER 174. The HASP command searches through queues on the IBM system. (Actually, the HASP command is a bit more complicated, but we'll get to that later.)

As the diagram above illustrates, jobs can flow from the CYBER 174 to the CYBER 175, and in either direction between the CYBER 175 and the IBM. Since a CYBER 174 job which produces output at an RJE site must traverse both communication links, we will use such a job for illustration.

While our mythical job is being processed on the CYBER 174, its name will always appear in one of the queues on that machine. Hence, a QUERY command executed on the CYBER 174 will be able to find it, but a HASP command or a QUERY command on the CYBER 175 will not.

Since the CYBER 174 does not have a printer or a plotter, any request for printing or plotting that is generated by our mythical job must be exported to a machine with access to the proper equipment. Jobs to be plotted, or jobs to be printed, must be sent to the CYBER 175, which does have access to the proper equipment. Jobs to be printed at an RJE site must be sent to the IBM 4341 which controls the RJE printers, but as the diagram shows, these jobs must first be sent to the CYBER 175 because there is no direct link between the CYBER 174 and the IBM.

The transmission process involves cooperating programs run on both CYBER systems. Periodically, a CYBER 174 program removes all jobs in its print and plot queues, and writes them out as permanent files to a disk shared with the CYBER 175. The cooperating program on the CYBER 175 periodically examines this disk and whenever it finds such a file, places it in the corresponding queue on the CYBER 175. As much as five minutes may elapse between the time a job is removed from a queue on the CYBER 174 and its appearance in a queue on the CYBER 175. During this period of transmission, a QUERY command executed on either the CYBER 174 or the CYBER 175 will be unable to locate the job. Once the transmission has been completed, a QUERY command executed on the CYBER 175 will be able to locate the job.

Since our mythical job is to be printed at an RJE site, it must be routed through the IBM 4341 which controls the printers at all RJE sites. Our job is sent to the IBM from the CYBER 175 via a special program called TIELINE. Once the job has been transferred to the IBM, a QUERY command will no longer find the job. You should now use the &DJ (HASP) command to enquire about your job.

As previously mentioned, the process now becomes a little complicated. The response to an &DJ command on the CYBER 175 will vary depending on how the job is being handled by the IBM. Any job on the IBM (or any job that has been transferred to the IBM) that is awaiting execution or is currently executing is handled by MVT.

A print job that is awaiting print at DCL is handled by VMPRINT (see diagram). If our mythical print job was to be printed at DCL rather than at an RJE, it would go through VMPRINT at this point. An &DJ (HASP) enquiry on the CYBER 175 would then produce the message

"IS AWAITING PRT"

if the job is in the DCL print queue.

However, our mythical job is to be printed at an RJE site so it is handled on the IBM by the RSCS (see diagram) which routes the job to the correct RJE printer via a PDP 11 computer. A job being processed by the RSCS is invisible to any &DJ or QUERY command. Therefore, once our mythical job is being handled by RSCS, we can no longer trace it by means of a system command. The only way such a job can be traced is through the RJE operator.

In addition to the regular printers at DCL and the RJE sites, there are two self-service printers available to users (one is located at Lincoln Hall and the other at ME). Print requests from the CYBER 174 or the CYBER 175 that are queued for these two printers are handled in yet another way. These jobs are kept on a special disk shared by the CYBER 174 and the CYBER 175 and cannot be located by a QUERY command from either machine. In addition, these jobs cannot be printed unless the print command is issued from the terminal adjacent to the self-service printer.

IBM

UPGRADE OF IBM EQUIPMENT

We have made plans to upgrade our IBM equipment. The upgrade will take place in several steps, but in aggregate will result in approximately 2.6 times the present computing power. At the time this article was written, the precise delivery dates for the equipment involved were not known. We expect that some or all of the upgrade will have been accomplished by the time you read this.

The first step taken will be to change the present IBM 4341 from a "group 1" with four megabytes of memory to a "group 2" with eight megabytes of memory. This will result in an increase of about 60% in computing power. Later, we will install an additional CPU, an IBM 4341-L10 with four megabytes of memory and one billion bytes of disk storage. This system will have computing power about equal to the present system. The tape subsystem will be upgraded to support 6250 bpi tapes and will be shared by both IBM systems.

This upgrade will accomplish several goals. It should improve our performance on our present workload, particularly in RJE support. It will permit us to allow terminal access to some new applications requiring interactive IBM service, e.g., the graphics support in SAS and SPSS®. The upgrade will allow us to extend interactive IBM support to additional classes, allowing students to work on-line, and further reduce the need for keypunches. Finally, it will increase the availability of large amounts of memory to those jobs requiring it.

IBM interactive services will be limited to those users whose needs demand it. Initially, use will be limited to users wanting to make use of the graphics facilities of SAS. Later, services will be extended to SPSS® users needing that package's graphics facility. The details of how work will be shared between the two systems have not yet been worked out.

IBM RATES TO CHANGE

Concurrent with the machine upgrade (assumed to have been completed by the time you read this), new scheduling and billing algorithms were installed on the IBM Hasp system. The new algorithms place less emphasis on the amount of memory used, recognizing that memory has become both cheaper and more plentiful. The modified billing algorithm also takes into account the increased speed of the upgraded IBM 4341.

The new OS/MVT scheduling algorithm uses job classes defined as follows:

class	*max (cpu seconds + hundreds of I/O requests)*	*max kilobytes of core*
A	60	500
B	60	1000
C	180	500
D	180	1000
E	600	500
F	600	3000
G	1800	3000
H	>1800	>3000

Jobs are placed in the smallest possible class. In general, jobs in small classes are run in preference to jobs in larger classes, much the same as always.

The old charging algorithm charged jobs according to the formula:

$$\text{Service Unit charge} = 0.00036*(cc + IO)*(0.0014*rf + 0.7)$$
$$= 0.0252*t*(1 + rf/500)$$

where:

cc = centiseconds of IBM 4341 cpu time
IO = number of I/O requests
rf = kilobytes of core used
t = cc/100 + IO/100

The new charging formula is:

$$\text{Service Unit charge} = 0.048*t'*(1-\min(t'/3600,0.5))*(1 + rf/2000)$$

Where t' has been calculated as t above, except that the centiseconds which enter into the calculation are assumed to be centiseconds of cpu time on the upgraded IBM 4341. (The IBM 4341 runs about 1.6 times faster since the upgrade).

The net effect of this change is to increase the cost of I/O by a factor of 1.65, decrease the penalty for using large amounts of core, and decrease the cost of cpu time or IO as large amounts of it are used. We applied the new charging formula to billing data gathered over

the past six months to assess its impact. The average effect was a significant reduction in price for the vast majority of users. Only 41 users would have experienced an increase in their bill of more than 5% and 10 service units. The largest price increase over the six-month period was 100 service units. Many jobs would have cost less under the new algorithm.

The charge for printing was raised from 1.00 service units to 1.15 service units per thousand lines printed. This charge results in better cost recovery for printing services and compensates for decreased revenues resulting from the modification of the billing algorithm.

FEATURE ARTICLES

TOM KERKERING AND HAROLD LOPEMAN RETIRE

Tom Kerkering and Harold Lopeman, longtime employees of CSO, retired at the end of August. Tom and Harold are both computer pioneers, having worked on computers at the University of Illinois at a time when there were at most half a dozen digital computers in the entire world. Both men helped build computing equipment which made pioneering use of new technology.

Tom and Harold both helped build ORDVAC, the first digital computer built at the University of Illinois. ORDVAC was a project of the "Electronic Digital Computer Laboratory," headed by Dr. R. E. Meagher. Tom joined the project as a student in 1949 and as a full-time employee in 1950. Harold came on board as a full-time employee in 1949, bringing with him a background as a radio engineer. Both men were part of the construction team building ORDVAC.

Tom Kerkering

ORDVAC was completed in 1951 and moved to Aberdeen Proving Grounds, Maryland, where it was used for ballistics research. The machine, built with vacuum tubes, was moved in a specially cushioned van. Tom was assigned to ride with the truck driver to ensure that he never went faster than 45 mph, a restriction imposed to minimize vibration and shock. The rest of the team followed by train. ORDVAC was operational within a week after its arrival.

Returning to Urbana, the construction team started building a successor machine to be known as ILLIAC I. It was completed and put into campus service in the fall of 1952. Tom and Harold continued to work on it, providing hardware maintenance and building advanced

graphics output devices and instrumentation designed to help keep ILLIAC I running as a production machine. ILLIAC I went into full-time (24 hour) service in 1954.

Investigations into magnetic core memory started in the mid fifties. Tom helped build a prototype memory for use at Chanute Air Force base which was the first use of small cores for information storage.

Harold Lopeman

Tom and Harold both helped build equipment designed around early transistors. Harold worked on TRANCE, the first computer which had logical elements built entirely of transistors. TRANCE was a test bed machine built to test the reliability of such elements. Harold recalls that transistors were expensive in that era and that a construction mistake could easily destroy hundreds or thousands of dollars worth of components.

In 1960 the group moved to the newly completed Digital Computer Laboratory where work on ILLIAC II began. Tom was placed in charge of the shop and Harold joined the team building ILLIAC II. Harold helped build the shift registers used in that machine and later did circuit testing and machine repair on it. In 1965 Tom joined the ILLIAC II group, working on a project to interface a PDP-7 minicomputer to ILLIAC II for teletype support. This was to be the start of Tom's specialization in communications, and the start of time-sharing on this campus.

ILLIAC II was taken out of service in 1968. The graphics work which had grown up as part of the ILLIAC II project was spun off as a project on its own and Harold joined it. The PDP-7 and its teletype support was moved to the IBM 360/75 where the PLORTS time-sharing system grew up around it. Tom continued his emphasis on communications, providing hardware support for PLORTS.

Harold joined the IBM group in 1971, the same year it was split off from Computer Science and organized as the Computing Services Office (commonly referred to as CSO). Both he and Tom worked on the project to deploy the present RJE stations across campus. Tom concentrated on the communications aspects while Harold focused on maintenance of the PDP-11 minicomputers, card readers and printers which formed the RJE stations.

As time-sharing services grew, Harold concentrated on the maintenance of the equipment and Tom on the installation and maintenance of the communications gear. Several years ago, Harold was placed in charge of the CSO shop where he oversaw the maintenance of equipment ranging from keypunches to minicomputers. Tom developed into a campus expert on data communications and a wide variety of multiplexors, interfaces, test equipment, etc.

Tom and Harold carry with them a large and valuable store of knowledge. Their intuitive skills and the knowledge gained by their long experience with computing and communications hardware will be greatly missed by the rest of us at CSO. We all thank them for their contributions to the campus and to CSO, and wish them well in their retirement.

FORTRAN FUTURES

This article is part of a series on the work of the FORTRAN Standards Committee X3J3 in producing the next revision of the FORTRAN standard. The reader is reminded that the features described in this article are not a part of any FORTRAN compiler currently available, but rather are proposed requirements for FORTRAN processors produced in the late 1980's through the mid 1990's. Although every effort has been made to accurately describe the current position of X3J3 on these matters, the development of a revision to the FORTRAN standard is an evolutionary process, and these proposals may be subject to refinement, revision, or even retraction. Comments on these proposals may be given to Kurt Hirchert of the CSO Systems Consulting staff, who is a member of X3J3.

In earlier articles in this series, we have seen that the next revision of FORTRAN will contain a number of new attributes which can apply to variables, including data structures, specification of real precision and exponent range requirements, and argument intents. In this article we will look at plans for changing the way attributes are specified.

The current FORTRAN specification statements can be termed "attribute-oriented." With some minor exceptions, like the specification of an array declarator on a type statement, a single FORTRAN specification statement applies only a single attribute, although possibly to multiple entities. As a result, although it would be possible to specify all the attributes of an entity together through the use of a series of specification statements, it is more convenient to specify lists of entities which share an attribute. Unfortunately, in the comprehension of a program, it is generally more useful to know all the attributes of an entity than to know all the entities with an attribute. X3J3 has adopted an extended "entity-oriented" specification system in which it will be possible, but not required, to specify all the attributes of a variable in a single statement. It is hoped this will make FORTRAN programs easier to read and understand.

In devising this new specification system, X3J3 is also trying to regularize the syntax of attribute specification. For example, the DATA and PARAMETER statements can be seen as performing similar functions (i.e., binding a compile-time value to an entity), but the syntax of these two statements is quite different. The DATA statement allows a list of entities followed by a list of values, while the PARAMETER statement requires entities and values to be paired. The PARAMETER statement allows the value to be specified as a constant expression, while the DATA statement allows only simple constants. The PARAMETER statement uses the character '=' to separate the entity from the value, while the DATA statement distinguishes the list of values by surrounding it by the character '/'. Where such irregularities exist, the new specification system will include new attributes with more regular syntax. For example, an INITIAL attribute is expected to be the long term replacement for the DATA statement (although both will be available in the coming revision).

Another benefit of the new specification system will be an eventual reduction in the complexity of the overall FORTRAN syntax. Under the current system, if one needs to add a new attribute to the language, one must add a new specification statement to declare that attribute.

One can hope that this new statement will have a syntax analogous to some existing specification statement, but this has not always been the case. In the new system, it will be necessary only to add a new attribute keyword to the existing general "declare" statement. As an encouragement to convert to the new system, it is expected that the new attributes in the next FORTRAN standard will be able to be specified only through the new system and that the old "attribute-oriented" statements will be removed from the language in the following standard.

Recent Actions by X3J3

The 83rd meeting of X3J3 was held August 9-13, 1982 in Boulder, Colorado. Actions taken at that meeting include the following:

- More than a quarter of the meeting was spent in consideration of a single proposal to determine the list of the new features which will appear in the next FORTRAN standard. Committee agenda time will not be scheduled for the consideration of features not appearing on this list, but the list itself may be amended by a simple majority vote, so the inclusion of a feature in this list or exclusion from it is only indicative of the feature's status with respect to the next standard, not an absolute decision. It is hoped that by limiting the features the committee will now be considering, the time to produce the next standard can be reduced.

  Most of the time spent on this proposal was spent in consideration of the specific features to appear in the list. There were votes on virtually every feature which had been previously considered by the committee, but which had been excluded from the initial version of the list to determine if it should be added. There were also votes on several of the features included in the initial version of the list to determine if they should be removed. The result of this voting was a version of the list which included all features previously adopted by the committee or currently under active consideration except the following:

  block construct names
  the bit data type
  varying length character data type
  compile-time facilities including macros

- Earlier in the meeting, a proposal was adopted to allow the EXIT statement to indicate an exit from a loop other than the innermost one by specifying the block construct name of the loop to be exited. A CYCLE statement was adopted which exits the current iteration of a loop, but which continues with the next iteration if there is one. The CYCLE statement can also apply to loops other than the innermost by use of a block construct name. (In light of the subsequent vote to limit the work of the committee, the status of these proposals with respect to the next standard is unclear.)

- A proposal was adopted which clarified the behavior of internal procedures with recursive hosts. The behavior specified is analogous to the behavior of similar constructs in existing languages such as PASCAL, ALGOL, and PL/I.

- The rules limiting the use of a dummy argument when there is a second means of access to the same entity (e.g., through a second dummy argument or through common) were liberalized and simplified.

- Rules were adopted which reconcile the differences between the "storage mapping" semantics used for array argument association in FORTRAN 77 and the "dope vector" semantics required to support the association of arbitrary rectangular array sections, as has been proposed for the next FORTRAN standard. In some cases, the implementation of these rules may require an execution-time test to determine which semantics are to be applied.

- A number of the array intrinsic functions were replaced by a smaller number of more general intrinsics using optional arguments specified by keywords.

- Proposals were adopted clarifying the meaning of a DATA statement in a recursive procedure. The second of these was adopted in spite of a general feeling that the behavior it described was unacceptable, because it was felt that the problem was with the existing nonrecursive rules and that this was merely a compatible extension of those rules. Thus the proposal was adopted to "plug a hole" until a revision of the rules for both the recursive and nonrecursive case can be proposed.

- A proposal was adopted to provide a means of distinguishing syntactically between the value to be returned by a function and a recursive reference to the function. It was noted that this feature may be dropped if a more general name conflict resolution facility is adopted.

- The variant data structure proposal adopted at the previous meeting was, in effect, readopted because of a procedural problem with the original proposal. The procedural problem was corrected without further consideration of the technical content of this proposal.

- A set of proposals were presented for the incorporation of event/exception/error handling into FORTRAN. After an informal vote endorsing the basic concept of these proposals, they were withdrawn from formal consideration because of misgivings expressed concerning some of the specific details of the proposals. It is expected that these proposals will be presented to the committee again after undergoing further refinement.

- A proposal to allow subscripting of any array valued expression, substringing of any string valued expression, and component extraction from any structure valued expression received a lukewarm reception, at least in part because some of the restrictions required to make this particular proposal "work" syntactically contradicted previous committee decisions.

- There was extensive discussion on what would be necessary to extend the concept of the SHARED DATA subprogram in such a way that it could serve as a means of implementing applications module standards. Among the topics covered in these discussions were the following:

  Facilities to allow user-supplied definitions for operators as well as functions
  Means of controlling which entities in a SHARED DATA subprogram are available outside that subprogram
  Controls on the entities a program unit accesses from a SHARED DATA subprogram and, especially, the defaults for these controls
  Strategies for resolving the conflict it it is necessary to access entities with the same name from different SHARED DATA subprograms
  Storage class rules for variables in a SHARED DATA subprogram
  The specification of aliases (i.e., alternative names) for entities known to a program unit and the relationship of an aliasing facility to the name conflict resolution strategies

- There was a lively debate on the extent to which it should be possible to use features new to FORTRAN in conjunction with features whose use is being deprecated and which may be removed from future standards.

- There was a presentation on possible approaches to providing array constants.

- There were presentations concerning the possible impact on FORTRAN of current standards work in the areas of open systems interconnection and data description/data manipulation languages (database management systems).

MISCELLANEOUS

STAN RANKIN HONORED

Stan Rankin of CSO was recently awarded a 25-year pin for his service to the University. We appreciate the continuing effort of Stan and recognize the value of his knowledge and experience.

COMPUTER-RELATED DISCOUNTS

Following are the current prices for FY 1982-83 for Data Processing Equipment from various vendors under the Illinois Educational Consortium contract.

CRT TERMINALS

IBM

Model 3101-10	$1,116.52
Model 3101-11	1,170.08
Model 3101-20	1,281.32
Model 3101-21	1,343.12
Model 3101-22	1,301.92
EIA Cable	65.00

The ordering address is:

Illinois Educational Consortium
1306 S. Sixth Street
Springfield, IL 62703
Telephone: 217-782-0268 Richard Newman

HAZELTINE

Model H-1420		$550.00
Model H-1500		905.00
Model H-1510		1,000.00
Model H-1520		1,240.00
EX-80 Mod. 20		1,090.00
EX-80 Mod. 30		1,185.00
Exec-10		975.00
Esprit I		545.00
Esprit II	with detachable keyboard	580.00

The ordering address is:

Hazeltine Corporation - Information
515 Kiowa Drive
Naperville, IL 60540
Telephone: 312-986-1414 Joe Gurgone

Hazeltine Corporation - Purchase Orders
95-25 Queens Bldg.
Rego Park, NY 11374

LEAR SIEGLER

ADM 5		$510.00
ADM 31		817.00
ADM 36		995.00
ADM 42		1,620.00
RG-1000	Retro-graphics Cards Includes LS 80 adaptor	$739.00

The ordering address is:

Hall Mark Electronics
13789 Rider Trail
Earth City, MO 63045
Telephone: 800-325-1021 Kim White - Jerry West
Purchase orders attention Diane Waterman

INFOTON

Model GT 101	$855.00
Model GT 100A	765.00
Model Visual 210	835.00
Model Visual 200	770.00
Model Visual 50	560.00

Visual Options

12" P31 Green Display (n/a for Vis. 50)	$30.00
12" P31 Green Display - Visual 50	20.00
14" P31 Green Display (n/a for Vis. 50)	75.00
4 additional pages of memory V400	350.00
7 additional pages of memory V300	300.00
Buffered printer port V100, V110, V400	125.00
Modem Cable - 10'	25.00
Maintenance Manuals	65.00

New Visual Products

Visual 500 Graphics terminal	1,795.00
Visual 550 Graphics terminal	1,975.00
Visual 100 - DEC VT-100 Emulator (includes Advanced Video option)	995.00
Visual 400 - Advanced 132 column smart terminal	1,050.00

Visual 300 - Smart terminal	795.00
Visual 50 - Data entry terminal	535.00
Visual 110 - Data General Dasher 200 & 6053 Emulator	995.00

The ordering address is:

Bronson & Bratton, Inc.
220 Shore Drive
Hinsdale, IL 60521
Telephone: 312-655-3600 Robert A. Raatz

TERMINALS - VT-100AA

DEC

VT100AA	w/detachable keyboard & EIA interface	$1,325.00
Advanced Video option		160.00

The ordering address is:

Data Collection Systems, Inc.
1740 West Ogden Avenue
Downers Grove, IL 60515
Telephone: 312-960-4050 Greg Schaller

PRINTERS

DIABLO - Xerox, high-qulaity

Model 1730 RO	$1,995.00
Model 1730 KSR	2,418.00

Above models substitute for 1740 and 1750.

The ordering address is:

Xerox Printing Systems Division
2301 West 22nd Street
Suite 205
Oak Brook, IL 60521
Telephone: 312-789-8100 Bruce Ganger

TEXAS INSTRUMENT - Non-portable terminal

Model 810 RO		$1,120.00
Model 820 RO		1,424.00
Model 820 KSR	with numeric keypads	1,545.00
Model 840 RO		711.00
Model 840 KSR		889.00

The ordering address is:

Hall Mark Electronics
13789 Rider Trail
Earth City, MO 63045
Telephone: 800-325-1021 Kim White - Jerry West
Purchase orders attention Diane Waterman

TELEPRINTER - Teletype Corporation

Model 4320 AAA(TTL)	$925.00
Model 4320 AAK(EIA)	995.00
Model 4320 AAB	1,365.00

The ordering address is:

Bronson & Bratton, Inc.
220 Shore Drive
Hinsdale, IL 60521
Telephone: 312-655-3600 Robert A. Raatz

TEXAS INSTRUMENT - Portable

Model 743 U	$775.00
Model 743 UL	855.00
Model 745 U	1,175.00
Model 745 UL	1,255.00
Manuals	20.00

Options for T.I. Models 743 and 745:

Upper/lower case	$100.00
APL keyboard	175.00
Answer back	75.00
Texprint 136	375.00
EIA Cable kit	60.00

The ordering address is:

Data Access Systems
8129 N. Austin Avenue
Morton Grove, IL 60053
Telephone: 312-967-0440 Joe Kilgore

DECWRITERS - DEC, non-portable

Mod LA-34AA (Decwriter IV)	$896.00
Mod LA-120DA (Decwriter III)	1,966.00
Letter Printer 100	1,563.00
Mod LA-12ABCD	1,507.00

The ordering address is:

Inland Associates Inc.
13100 Manchester Road
St. Louis, MO 63131
Telephone: 314-391-6901 Bob Omer

SPINWRITERS - NEC, non-portable

Model 7710-1	$2,032.00
Model 7720-1	2,378.00
Model 7715-1	2,087.00
Model 7725-1	2,444.00
Model 7730-1	2,042.00

The ordering address is:

Hall Mark Electronics
13789 Rider Trail
Earth City, MO 63045
Telephone: 800-325-1021 Kim White- Jerry West
Purchase orders attention Diane Waterman

FLEXIBLE DISK SYSTEMS

DEC Compatible

Model DSD 880 D/8-L11-A	$5,246.00
Model DSD 880 D/30-L11-A	6,971.00
Model DSD 480-L11-2-A	3,446.00

Model DSD 470	3,296.00
Model DSD 440-L11-2-A	2,996.00
Model DSD 430-A	2,621.00
Model DSD 734-10	5,246.00
Model DSD 736-40	6,971.00
Model DSD 724-10	6,001.00
Model DSD 726-40	7,748.00

The ordering address is:

Data Systems Design, Inc.
c/o First Representative Co.
500 Park Boulevard, Suite 390
Itasca, IL 60143
Telephone: 312-773-9540 George Bliss

MODEMS

Acoustic Couplers

Model 302A2-13		$178.90
Model 302A2-33		198.90
Model T212A		796.90
Includes a permissive cable RJ-11-C or a programmable cable RJ-45, but must specify which cable you want.		
Model 370E2-12		278.90
Model 370E2-42		338.90
Model 305E2-12/22	Modem	148.90
Model 305F2-12/22	Modem	138.90
Model 350F3-72	Line Driver	138.90
Model 354F3-72L	Quad Line Driver	342.90
Model 334E2-42L	Modem	638.90
Model 334F2-22L	Modem (dial-up/leased line)	588.90
Model 208A/B		1,751.90

The ordering address is:

ComData Corporation
7900 North Nagle Avenue
Morton Grove, IL 60053
Telephone: 312-470-9600 Philip M. Towle

Modems

Model MT212D-R		$425.00
Model MT212D-S		520.00
Model MT212A-S	300/1200 baud w/CA211 cable	600.00

The ordering address is:

Multi-Tech Systems, Inc.
82 Second Avenue
New Brighton, MN 55112
Telephone: 612-631-3550 Thomas E. Heimerman

COMPUTERS

DEC - LSI 11/2 and 11/23

HARDWARE

KD11-HA	LSI-11/2 Central Processor Unit (CPU). One module	$538.00
KEV11	Fixed and floating point instruction set (EIS/FIS) for LSI-11/2 CPU. 40-pin DIP.	128.00
KD11-GC	LSI-11/2 CPU (KD11-HA) plus 32k byte Multifunction Option Module (MXV11-AC). Two modules.	890.00
KDF11-AA	LSI-11/23 CPU with Memory Management Unit (MMU). One module.	1350.00
KEF11-AA	Floating point instruction set option for LSI-11/23 CPU. 40-pin DIP. Requires a CPU with MMU.	269.00
KXT11-AA	The Falcon SBC-11/21 16-bit single board computer which contains the following: * 40-pin PDP-11 microprocessor * 4Kb of RAM (expandable to 8Kb) with battery backup hooks * four 28-pin memory sockets for on-board RAM or ROM(32Kb) * two asynchronous serial I/O ports * 24 parallel lines * real time clock * LSI-11 bus interface	506.00
KXT11-A2	Set of 2 PROMs to provide monitor operation for FALCON.	64.00
MSV11-DD	64K bytes of RAM including on-board refresh. One module.	320.00

MSV11-LF	128K bytes of RAM including on-board refresh. One module.	650.00
MSV11-LK	256K bytes of RAM including on-board refresh. One module.	950.00
MSV11-PL	512K bytes of RAM with full function parity and on-board refresh. One quad-height module.	1650.00
MXV11-AC	Multifunction option. Includes 32K bytes of RAM, 60 Hz crystal clock, two serial line units (150 to 38400 baud). Accepts two customer-supplied 5V 24-pin UVPROK or fusible link PROM chips. One module.	480.00
DLV11-J	Four independently configurable serial lines. Baud rates from 150 to 38400. One module.	371.00
DZV11-B	Four-line asynchronous multiplexor. Includes modem control programmable speed, character length parity and stop bits. 50 to 9600 baud. One quad-height module.	704.00
RL01-AK	Add-on 5.2 byte top-loading removable cartridge disk drive.	1920.00
RL02-AK	Add-on 10.4M byte top-loading removable cartridge disk drive.	1920.00
RLV11-AK	A RL01-AK with controller for LSI-11. Two quad-height modules. Requires H9273-A backplane or BA11-N series box.	3392.00
RLV21-AK	A RL02-AK with controller for LSI-11. Two quad height modules. Requires H9273-A backplane or BA11-n series box.	3657.00
RLV22-AK	A RL02-AK with 22-bit controller. One quad height module. requires H9275 or equivalent backplane and memory and CPU capable of supporting 22-bit addressing.	3657.00
RXV21-BA	Dual drive double density floppy disk subsystem, 1M byte capacity plus one LSI-11 DMQ interface. One module.	2241.00
BA11-ME	4 x 4-slot expansion box which includes the H9270 backplane and the H780 slave power supply. Accepts up to 8 double-height module. Not compatible with RLV11 or RLV21.	1024.00
BA11-NE	4 x 9-slot expansion box which includes the H9273-A backplane and a 240 watt power supply. Accepts up to 9 double or 9 quad-height modules. Compatible with RLV11 and RLV21.	1280.00
H9275-A	4 x 9-slot nonexpandable backplane with card guides and built-in terminators. Contains the 22-bit LSI-11 Qbus. Will accept 18 double-height or 9 quad-height modules. Not compatible with RLV11,RLV21,BCV1A,BCV1B,TEV11,REV11-A.	272.00

Software

QJ013-DZ	RT11 Operating System. No media. Copy only license.	960.00
QJ813-AY	FORTRAN/RT11 language on floppy disk. Digital support.	1050.00
QJ813-DZ	FORTRAN/RT11. No media. Copy only license.	410.00
QJ913-AY	BASIC-11/RT-11 language on floppy disk. Digital support.	1218.00
QJ913-DZ	BASIC-11/RT-11. No media. Copy only license.	378.00
QJ628-DZ	RSX11-M Operating System. No media. Copy only license.	1920.00
QP230-AQ	FORTRAN IV/RSX11-M language on RL01. Digital support.	1050.00
QP230-DZ	FORTRAN IV/RSX11-M. No media. Copy only license.	448.00
QP668-AQ	FORTRAN IV-PLUS(F77)/RSX11-M language on RL01. Digital support.	5603.00
QP668-DZ	FORTRAN IV-PLUS/RSX11-M. No media. Copy only license.	2797.00

The ordering address is:

Digital Equipment Corporation
2400 North Main Street
East Peoria, IL 61611
Telephone: 309-694-4235 William C. Ziccardi

APPLE II+ and APPLE III

Apple II+ 48K Computer		$1,126.00
Basic System	2,708.00	
Scientist's System	2,924.00	
Educator's System	2,983.00	
Business Manager's System	4,260.00	
Apple III - Packaged Systems		
128K bytes of RAM and System Software Apple III monitor, Silentype Printer with Interface		$3,090.00
128K bytes of RAM and System Software Apple III monitor Disk II for Apple III, Prism 80 without option		4,162.00

128K bytes of RAM with System Software Electrohome 1302-1 13" RGB Color Monitor, Disk II for Apple III, Qume Sprint 9/45 LTD, Qume Forms Tractor	5,717.00

The ordering address is:

Random Access Machine Corporation
d/b/a Byte Shop
1602 S. Neil Street
Champaign, IL 61820
Telephone: 217-352-2323 Mark Brown

Hardware and Software accessories compatible with the Apple II and Apple III Computer Systems listed above:

Qume Sprint 9/45 LTD	$2,129.00
C.Itoh F10-40PU	1,405.00
IDS Prism 80 with color option	1,460.00
IDS Prism 80 without color	972.00
IDS Prism 132 with color option	1,593.00
IDS Prism 132 without color	1,285.00
Zenith ZVM-121 12" Green Monitor	112.00
Apple Language Card	137.00
Apple II Pascal	169.00
Apple Super Serial Interface	137.00
Visicalc (Apple II)	183.00
Videx Videoterm	253.00
Apple III 128K Computer	2,616.00
Apple III 256K Computer	3,184.00
Apple III Green Monitor	174.00
Apple III Pascal	171.00

Apple III Business Basic	84.00
Visicalc III	169.00
Profile Hard Disk Drive	2,617.00
Apple III Universal Parallel Interface Card	158.00
Epson MX80-FT Printer	603.00
Epson MX-100 Printer	724.00
Epson Parallel Interface & Cable	90.00
Tractor Feed for C.Itoh F-10	250.00
Electrohome 1302-1 13" RGB Color Monitor	385.00

The ordering address is:

Byte Shop
1602 S. Neil Street
Champaign, IL 61820
Telephone: 217-352-2323 Mark Brown

SPECIAL

CSO VIDEOTAPES AVAILABLE

Reprinted from *OFF-LINE*, September 1981.

CSO has produced a series of eight videotapes which introduce the novice to the CYBER Systems. A viewing guide containing the major pictorials used in the series is available and can be used to facilitate note taking.

The title and a brief synopsis of each of the videotapes is given below. Running time is 10-15 minutes for each videotape.

Introduction to Computing at CSO

A brief look at the steps required to solve a problem using a computer and some of the hardware used.

Using a Terminal

A description of the physical operation of a terminal and some of the keys that have a special meaning to the CYBER.

Introduction to CYBER Time-Sharing

A tutorial for logging on and off the CYBER.

File Usage - Local File and Indirect Access to Permanent Files

An introduction to CYBER files and the commands used to manipulate them.

Introduction to ICE Text Editing

A tutorial on entering and modifying files with ICE.

Running a FORTRAN Program - Concepts

The concepts of compilation, loading and execution.

Running a FORTRAN Program - The PROGRAM Statement

The PROGRAM statement and its relationship to files accessed by the program.

Running a FORTRAN Program - Control Statement

The control statements used to compile, load and execute a FORTRAN program.

Anyone can view these videotapes by going to the Undergraduate Library in person to make a reservation for use of the videotape equipment. Ask for a copy of the viewing guide when you check out the videotape for viewing.

ZETAVU: A UTILITY FOR PREVIEWING ZETA PLOT FILES

Reprinted from *OFF-LINE*, November 1981.

All ZETA plot files produced by ZETA software, GCS, NCAR, Easy Graphing or other software may be displayed on Tektronix Graphics terminals. ZETAVU is a new program which reads these ZETA plot files (often named TAPE99), interprets the instructions contained therein, and displays the graphics on a Tektronix 4006, 4010, 4014, or 4027 (color) graphics terminal. ZETAVU is run on the CYBER with the commands:

```
GRAB,ZETAVU.
ZETAVU.
```

A 16-page user manual for ZETAVU may be obtained and printed by entering:

```
WRITEUP,ZETAVU.
PRINT,ZVUDOC/CC/ASCII/EJ.
```

TREATMENT OF OVERHEAD CHARGES

Reprinted from *OFF-LINE*, January 1982.

As of January 1, 1982, all sponsored research accounts operating under the new overhead structure of "modified total direct cost," will be handled and flagged automatically through the UFAS accounting system. Charges made on a ledger 5 account (sponsored projects) will have an additional charge of 8.3% added by the UFAS accounting system against an object code of 5569. Accordingly, 5569 is no longer a legitimate object code which can be used by departments; it is restricted to use by the UFAS accounting system. This additional charge is for overhead charges associated with computing services. Computing service is not included in the direct cost to which the 42.4% overhead is applied.

Questions concerning this matter should be directed to the contracts office.

NEW USER-CONTRIBUTED SOFTWARE LIBRARY

Reprinted from *OFF-LINE*, January 1982.

CSO is now sponsoring a library of user-contributed software. The purpose of the library is

to foster sharing of software between CYBER users. CSO is providing the disk space, the tools used to access the offerings in the library, and is making all library updates, but is not otherwise involved. This is a user-to-user exchange.

Updates to the library will be made on a periodic basis. CSO will decide which programs are entered into the library using the following guidelines:

- The program or package must be generally useful and not of interest primarily to a single class or research group. Games will not generally be accepted for inclusion in the library.

- The contributor of the software must provide a machine-readable description of it sufficient to allow a potential user to judge its usefulness and, hopefully, begin using it. Ideally, complete documentation should be in machine-readable form. If complete documentation is available only in printed form, a copy must be supplied for storage at the Systems Consulting Office (1208 W. Springfield) where it will be made available for inspection.

- The user must state the level of support given to the package. At a minimum, the contributor must agree to make any modifications required by installation of a new version of the operating system, must give his/her name, phone number and address, and must be willing to talk to users encountering difficulties using the program. The contributor's name, phone number and address will be made available to users of the program.

- The program must be stable and not undergoing continuing development. Repeated updates will not be made to an existing program in the library.

- Copies of all files used must be provided to CSO for inclusion in the library.

CSO will make broad judgements about the usefulness of the software, but will not make any tests for reliability or usability. As a consequence, CSO will not provide software maintenance or consulting help for the contents of the library. These functions remain the contributor's responsibility.

To list a directory of the library, enter the command:

```
SHARED.
```

The output generated will contain one line for each product in the library. Each line will contain a product name followed by a one-line description.

To obtain a description and on-line documentation for a specific product in the library, enter the command:

```
SHARED,product name/WRITEUP.
```

This command will produce a local file or files containing on-line documentation for the product named. The file and its attributes (e.g., ascii, carriage control) will be identified in the output produced by the command. Finally, to use one of the programs or packages in the

library, enter the command:

SHARED,*product name.*

Then follow the instructions given in the on-line documentation.

Programs will be removed from the library if they are not being used enough to warrant their inclusion or if the contributor has stopped providing support. Advance warning will be given before any deletions are made. If you plan to use a program in the library on an ongoing basis, we recommend that you obtain a source copy from its contributor as a precaution against its eventual removal from the library. CSO will not maintain source copies of programs available in the library for archival or distribution purposes.

If you have software which you would like to contribute to the SHARED public library, please contact Kurt Hirchert, 173 DCL (333-6264).

SAVING COPIES OF OUTPUT

Reprinted from *OFF-LINE*, March 1982.

Users are advised to save copies in disk files of print, punch and plot output produced by expensive or long-running jobs. These file copies would serve as insurance against possible loss or destruction of printed, punched, or plotted output, and can be easily purged once the output is safely in hand. This is especially important to time-sharing users since we ordinarily do not give refunds for lost time-sharing work. Batch users also can benefit since it is usually much more convenient to requeue a file for output than to regenerate the data in the file.

THE CYBER FILE MIGRATION SYSTEM

Reprinted from *OFF-LINE*, March 1982.

Disks, like airline seats, are commodities which are much too expensive to leave unused. Whereas unused airline seats are defined as empty, unused disks can be either empty or full of unused files. There are two possible solutions to the problem. One is to absorb the cost of the unused space (and of course, pass the additional cost on to the users); the other is to deliberately overbook the space. Both the airlines and CSO find it more cost effective to do the latter.

We allow about 3,600,000 PRUs of permanent files to reside on our permanent file system. This has been determined, mostly by trial and error, to be the maximum number which still allows the system to provide reasonable response times during the afternoon peak. (The system needs free disk space to swap executing jobs from core, allocate local files, and store new or extended files.) At any given time on our system, about four times the number of PRUs are migrated as are resident on our disks.

Nightly, during the third shift, the operator runs a procedure to enforce the 3,600,000 PRU limit to insure that we can make it through the following day. The programs which run as part of this procedure read the entire permanent file catalog. They ignore any files which already have been migrated or have been accessed during the previous working day. Since this process occurs after midnight, any files accessed during that day will be on disk the next day (files accessed on Friday will be on disk on Monday since Saturday and Sunday are not working days).

For all files which have not been accessed that day or already migrated, the procedure assigns a priority according to the formula:

priority = size * # working days since access * (unclaimed)

Where *(unclaimed)* is normally 1. If, however, the file has been marked as "must be claimed" (i.e., the DIRECT program prefaces its name with a minus sign), then *(unclaimed)* takes as a value the number of days the file has been in the "must be claimed" state. This unclaimed factor causes disk files which have no one paying for them to be taken off disk faster than normal. Files are then chosen for migration in decreasing priority order until the disk space left is within acceptable limits. Normally, files with a priority of less than 2000 are chosen infrequently for migration, and files with a priority of less than 1000 are rarely chosen.

To help illustrate this (assuming that files of priority less than 1000 are never chosen), the following table shows the number of working days a file of a particular size should remain on disk if it is not accessed or not claimed.

Size (PRUs)	*Days Unaccessed*	*Days Unclaimed*
5	280	22
10	140	15
50	28	7
100	14	5
250	6	3
600	2	2
1000	1	1

If a file is chosen to be migrated, the actual freeing of its disk space can occur in two ways. If the file was previously migrated and was reloaded but not modified, then the version of the file on the migration tape is still current and the file can just be marked as migrated and its disk space freed. If the file has never been migrated, or has been reloaded and modified, then the current version of the file on disk must be written to a migration tape. After a number of files are written to a migration tape, and either the tape is full or enough space has been freed on the disks, the tape is rewound and read to insure its integrity. Only after this has occurred, will a file be marked as migrated and its disk space freed. To further insure reliability, the tape is again rewound and copied to another tape. This second tape is then stored in another campus building to insure that the files are still "saved" in the event the original tape is damaged.

After migration runs are completed for the night, the tapes are again mounted and copied to a Mass Storage System (MSS) which we have acquired from CDC. The MSS is sort of a "juke box" for cartridges of magnetic tape. Each cartridge contains about one quarter of a

1600 bpi reel of tape and the MSS holds 2000 cartridges. The device, although fairly cantankerous, can mount, locate and retrieve a file and return the cartridge in about a minute without operator intervention.

This is, no doubt, enough information about the removal of files. The important thing to most people is how to get a file back onto disk once it has been removed. When a RELOAD or a USE command is executed for a particular file, the request is added to a list of files to be reloaded. If there already is a request in the list for that particular file, the new request is ignored. When the MSS is ready to reload another file, it scans the list and assigns a priority to each file in the list. The priority it assigns is the number 1 divided by the number of outstanding RELOAD requests for files belonging to the same user number. The MSS then attempts to reload the file with the highest priority. If the reload is successful, the file is back on disk and its entry in the list is deleted. If the reload from the MSS fails or cannot be done (this happens for about 5% of all reload requests), the file is marked as a MSS failure in the list. These failed reloads then are reloaded from tape.

A tape reload processor periodically scans the list and begins the reloading of files which have been marked as MSS failures. The tape processor issues a request to the operator to mount a migration tape and then reloads the files from it. To speed up the entire reloading process, after MSS failures have been reloaded from a tape, the tape processor will reload any other files in the list which are also on the mounted tape. Finally, if the tape processor completes reloading all failed files and finds requests that are more than 1/2 hour old, it will reload these files to help the MSS.

In conclusion:

- Migrated files are safe. The files reside on two tapes and, most likely, the MSS.
- The migration of a file, however, does not affect its deletion by the enforcement of CSO's disk policy (the purging of files unused for one year or unclaimed for 120 days).
- The response to a reload request for one file should be fairly quick; however, the response will be slower for requests of more than one file at the same time. If you issue a request to reload the same file repeatedly, it does not affect the speed of the reloading process, since these subsequent requests are ignored. The average time to reload a file from the MSS is less than five minutes, and the average time from tape is about 30 minutes.
- The type of operation (access or modification) performed on a file does not affect the likelihood of the file being migrated.
- The larger a file is, the more prone it is to be migrated. Also, taking into consideration the priority formula and the fact that files with a calculated priority of less than 1000 are never migrated, one can deduce that a validly claimed file of less than four PRUs should never be migrated. This is because 4 PRUs * 250 (approximate number of work days in a calendar year) equals 1000. Remember, however, that a file that has been unaccessed for 365 days (even if less than 100 PRUs) will be purged for nonuse.

COST RECOVERY FOR VARIOUS SERVICES

Reprinted from *OFF-LINE*, April 1982.

There is a continuing debate, both within CSO and with user groups, about the cost recovery for various services. CSO services are heavily subsidized by campus funds, which are used to support the Research Board and CICU allocation systems.

Present practice at CSO is to try to price each service in accordance with the costs of providing the service. Although we do limit some services to paying customers only, we provide most services through allocations. The issue of where to draw the line between "free" and "charged" services is a matter of policy with significant operational implications.

Alternate modes of operation would be to charge users for all services, distributing the campus funds to user departments as earmarked or discretionary funds, or to apply the campus funds to subsidizing prices for all users. There are both philosophical and practical advantages to the various approaches. If you look around the country at other universities, you will find almost every model in use, with none clearly superior at making service available.

Our present model makes a large fraction of the service available as a campus resource backed by campus funds. The major advantages of this funding model are:

- providing service on the basis of need or objective rather than on the basis of available external funding,
- stability of service,
- large volume of service at low cost, and
- relatively full utilization of available resources.

There are, however, some problems which this model does not handle easily and we must find ways to deal with them. Some of these problems are finding funds to provide new or specialized services, finding some way to control waste of "free" facilities, having a rational basis for deciding between departmental (or project) facilities and a central one, and funding growth or services in areas of high marginal cost, all while preserving the fiscal integrity of CSO. Almost every case which has come up for discussion involves more than one of these problems, and many cases reflect a conflict with those things our model does well.

To give a few examples, consider the following:

> ***Photocomposition:*** A highly specialized service where much of the cost is for labor and supplies rather than capital equipment. Free use would create a huge demand with no means of covering the costs, and most of the use would be unnecessary. The current charge rates result in fairly low use of equipment, but recovery of most of the costs.
>
> ***Diablo:*** A service of general utility where the alternatives are departmental

equipment, word processors, or traditional secretarial facilities. Intensive on labor. Attractive output would cause heavy usage without a charge. With a charge, large users can decide to buy private facilities, to our mutual advantage.

Manuals: A necessity to those who want to get work done. Much more heavily demanded if free and accessible. If free, it's easier to take another one than to carry a copy. Charging should help limit waste, and will provide funds for manual production and acquisition.

Dial-up Terminal Access and Dedicated Ports: With a relatively small investment, people can join the clamor for the pool of available dial-up ports. A sufficiently large number of people have taken this option and as a result, the available ports are overloaded for only a few hours each day while these same ports go idle a large fraction of the time. A major effect is to concentrate demand into greater peaks than public facilities present. A real charge for this service would generate funds to expand the number of lines (perhaps with a resultant overload on the computer itself). The high charge for dedicated ports makes them attractive only to those who have service demands outside of peak times as well as during them. (A more extensive discussion of dial-ups appears in the following article.)

The examples shown are typical of the services where limitations to real money (vs allocated) have been discussed. They are sufficiently different to point out the variety of considerations involved. As funds from the campus, and earnings from grants and contracts, come under increasing pressure, the influence of fiscal stability, financing growth and special services, and out-of-pocket costs (printing, phone bills, labor, etc.) cannot be ignored. We would appreciate your comments on these issues. You can certainly expect to see continuing activity in this area as we try to get users to express their needs and we try to evolve the center in response to these needs.

A sampler of other related issues:

- Should there be a charge for storing and/or submitting tapes?
- Should there be some fee, flat or based on use, for the "free student account"?
- Should hard money users have preferred access to services (e.g., some extra lines reserved for their use)?
- Under what circumstances are "games" a waste of resources?
- Should we continue to subsidize keypunching?
- Should all use of consumable supplies (e.g. paper, plotter pens, etc.) require hard money?

DIAL-UP SERVICE

Reprinted from *OFF-LINE*, April 1982.

Dial-up service is beginning to be a problem again this semester, and a new dimension has been added with the increased number of 1200-baud modems that users are purchasing. In this article we attempt to characterize the situation, discuss where dial-ups fit into present and future services, and outline a few alternatives.

At the present time the CYBER 175 system includes 66 dial-up lines for 300-baud terminals and 16 lines for 1200-baud terminals. The CYBER 174 has 16 dial-up lines for 300-baud terminals and none for 1200 baud. Dial-up service is not presently a significant part of the IBM service. These ports support privately (or departmentally) owned terminals, since all CSO public terminals have now been connected by direct lines.

The number of terminals owned outside of CSO is not known, but it is certainly several hundred. Increasingly, these terminals are owned directly by faculty and students, and most new terminals can operate at speeds of at least 1200 baud. Currently, dialable phone facilities are limited to 1200 baud, and CYBER service is also. Neither of these factors is a permanent condition. Although the number of terminals is unknown, data is available on the composite patterns of use (e.g. numbers of calls tried and completed, time of day usage, length of call).

Chart 1 shows a representative time-of-day pattern for a weekday during the semester. Chart 2 shows the comparable pattern for the weekend. Table 1 shows a few relevant statistics related to this data for a recent week.

Table 1 - DATA

	174 300 baud	175 300 baud	175 1200 baud
Number of dial-up ports	16	66	16
Number of sessions	3337	8781	2357
Sessions per port	209	133	147
Connect hours	1133	3302	939
Hours per port	71	50	59
Average session (min.)	20	23	24
Estimated % of hours saturated*	20	10	25
% of available use**	67	48	54

* Based on 106 hours per week (8AM - Midnight weekdays plus scheduled hours on weekend).

** Data based on a week in late February 1982.

A number of points are clear from the charts. For example, on normal workdays there are several hours during which the system cannot answer every call. During those hours some calls are answered, but only when someone has released a line. On the other hand, the last few lines are getting only 10-15 hours of use per week, and this only during the busiest weeks. Thus, the revenue associated with these lines is small, and that associated with the next few lines that might be added would be even smaller.

Another obvious factor is the common timing of all the peaks, and the height of the peaks relative to average loads. These peaks would be even larger if more ports were available, and would result in either degraded performance or the requirement for stronger limitations on what could be done (i.e., memory limits, etc.). We believe that, in general, work which cannot be done during peaks is moved to other times, at some inconvenience to the user, but with a better distribution of the workload. Balancing inconvenience vs distribution is a continuing practical issue which eventually gets back to a financial question.

At the present time we do not have funds available to provide more dial-up facilities, and with the exception of the 1200-baud service, do not feel that the overload and system capacity would justify such an addition. Thus, users deciding to acquire modems should recognize that service will be available, but with contention from other users.

As a general practice, dial-up service is not intended for those who make very extensive use of the systems. That is, dial-ups are not intended as a private and preemptive pathway to the computer for use many hours per day. That function is better provided, on campus at least, by the acquisition of a private port. The latter, in turn, produces real revenue to CSO, and is priced to cover its costs. This service also could become overloaded unless revenues from some source, campus budget or grants, keep the capacity of the systems growing to meet the demand. We do expect that private ports will be an increasingly important part of the services offered.

Another alternative which has been discussed is to reserve a portion of the dial-up pool for paying customers only, and increase the pool if this would generate more revenue. The principal advantages are that this encourages people to get funds and makes certain that they have the opportunity to spend them.

This issue will receive further discussion in CSO and its committees. At the present time, the only decision is to hold off on expansion of the number of dial-ups, and to promote acquisition of private ports.

The dial-up situation has other policy issues related to fairness in managing a limited resource. Last year we tried a variety of measures which did not prove very satisfactory. We have no new measures to announce at this time, but may have to institute some if the situation becomes a problem, and is being aggravated by unreasonable use. We ask your cooperation in minimizing the problem by moving your use of dial-ups to less busy times whenever possible, minimizing call duration, and releasing lines when not needed.

GOOD QUALITY PRINTED OUTPUT

Reprinted from *OFF-LINE*, April 1982.

The Consulting Office consistently gets questions about obtaining good quality printouts. Hopefully this article will provide answers for our users. There are three types of printers: the line printers, the PPS (Page Printer), and the Diablo.

- ***Line Printer:*** If you want better quality printouts from the regular line printer, you can obtain adequate results by using the following command on the CYBER:

  ```
  PRINT,filename/FORMS=6/RJE=LOCAL/EJ.
  ```

  Be sure to include the switches /AS if your file is in upper/lower case and /CC if your file has carriage control.

  On the IBM, include the following ID card:

  ```
  /*ID FORMS=6
  ```

  The FORMS=6 option will provide you with 6 lines per inch (rather than the normal 8 lines per inch), 11" X 14" paper (rather than the normal 8 1/2 X 14" paper), the reverse side of the paper (no lines), and a fresh ribbon for better print quality.

- ***Page Printer:*** Output from the page printer is on 8 1/2" X 11" paper. The printing is turned 90 degrees from normal so that there are 132 print columns along the 11" dimension and 64 lines along the 8 1/2" dimension. There are three holes punched in the top (along the 11" dimension).

  This type of printout is especially useful for keeping listings of very large programs in three-ring binders. The quality, while good, is not appropriate for reproduction. To send a print job to the PPS from the CYBER, enter:

  ```
  PRINT,filename/FORMS=PPS.
  ```

  Be sure to include the switches /AS if your file is upper/lower case and /CC if your file has carriage control.

  To send a print job to the PPS from the IBM, include the ID card:

  ```
  /*ID FORMS=PPS
  ```

  The output from the PPS should normally be available in Room 14 DCL within 24 hours.

- ***Diablo:*** For those users who desire a high quality printout (for theses, reports, etc.), CSO offers a carbon ribbon printing service via a Diablo printer. Two type styles are available: PICA (10 characters per inch) and ELITE (12 characters per inch). Vertical spacing is 6 lines per inch. The paper used is 20 pound, unlined bond, fanfold, 11" X

15", suitable for most purposes.

CSO charges $0.05 per PRU for printing files on the Diablo. (Printing of an RNF upper/lower case file averages out to about 20 cents per page.) There is a minimum charge of $1.00 *for each Diablo run.* CSO accepts personal checks payable to the University of Illinois, and can bill charges to any valid, real-money University Account Number.

Printed results can be picked up within 24 hours in Room 123 DCL. You will be notified via MESSAGE on the CYBER when your file has been printed. A billing form will be attached to the front of the output, which the user fills out and returns to the payment bin, with check attached if payment is by check. All questions should be directed to the Diablo operator, Room 123 DCL, phone 333-3994.

To have a file printed on the Diablo, you must use the DIABLO proc. In order to use the DIABLO proc, you must first do

```
GRAB,DIABLO.
```

Next, you must make the file to be printed local, with the GET command, or create it via RNF. The file to be printed must contain carriage control, and must begin with a page eject. If you are having an RNF output file printed, include the RNF command .LPT in the RNF source file, and the carriage control will be generated automatically.

Finally, execute the DIABLO proc by typing

```
DIABLO.
```

You will be prompted for your surname, the desired mode of payment (ACCOUNT or CHECK), the font desired (PICA, 10 characters per inch, or ELITE, 12 characters per inch), and the name of the file to be printed. If you specify ACCOUNT for the mode of payment, you will be prompted for the ACCOUNT NUMBER and the ACCOUNT TITLE. This must be a ***real money*** U of I account, and ***not*** a PS number or CYBER account. If the file you specify is not local, DIABLO will abort.

Before each file is printed, the Diablo print head is set so that column 1 of your document starts about 3 inches from the left of the paper, which is 15 inches wide. This gives you ample room to trim off enough paper to give correct margins, in most cases. If you have a very wide document (say over 100 columns wide) then you might need to reset this parameter; for information on this, call the Text Processing Consulting Office, 333-7318.

Also, the mechanical top of form on the Diablo is set to the second line of the form, so as to allow an extra 1/6 inch at the top for RNF users, giving them (if defaults are used) a total of 4/6 = 2/3 inch above the page number. This extra 1/6 inch comes off the bottom of the page, which then becomes 7/6 = 1 1/6 inches.

The appropriate horizontal and vertical spacings are set for each file in the queue. Thus, the user can include Diablo escape sequences in the file for changing Diablo parameters, without affecting the settings for other users.

A sequence number is printed in columns 1-5 of each page, for use by the Diablo operator in case a restart is necessary. These sequence numbers do not affect the user's document in most cases, because of the 3 inch offset mentioned above; i.e., they will normally be trimmed off. However, users who wish to suppress this sequence number may do so by including any character on the first line of each page of text, following the page eject "1" in column 1. For RNF users, this requires modification of the FRCPAGE macro (see the RNF Reference Manual or the Text Processing Consultant).

MANUALS TO GO ON SALE

Reprinted from *OFF-LINE*, May 1982.

As indicated in the April issue of *OFF-LINE*, CSO will begin selling some of the manuals formerly distributed free. Six manuals were removed from the list of free documentation on the basis of cost to CSO. Most of the manuals which will now be sold are elementary in nature. Some have been distributed at the rate of more than 3000 copies per year.

We have made arrangements with the Illini Union Bookstore, 715 South Wright Street, to sell these manuals individually. The manuals will be available in the Reference Section of the bookstore beginning June 10, 1982 (the start of summer session registration).

Because of space and manpower limitations, the CSO Distribution Center will only sell the *RNF Documentation - Tutorial, Macros and Reference* manual individually. The other five manuals will be shrink wrapped and sold only as a unit. Users are reminded that the Distribution Center handles no cash; manuals must be paid for by check.

The manuals affected and their prices are as follows:

	Illini Union Bookstore	***Distribution Center***
RNF Documentation - Tutorial, Macros and Reference	$4.00	$4.00
Introduction to the CYBER Systems	$2.00	*
A Tutorial Guide to the ICE Text Editor	$1.25	package
ICE Reference Manual	$3.25	price =
An Index to Software on the CYBER	$3.25	$11.00
CYBER FORTRAN Debugging	$1.25	*

Instructors who expect to recommend any of these manuals for use by their students are asked to inform the Illini Union Bookstore of that fact and the number of students who can be expected to buy the manual. This will ensure an adequate supply of manuals on the shelf.

DO YOU USE HEARYE?

Reprinted from *OFF-LINE*, May 1982.

News of current or planned changes and/or problems is broadcast to the user community through HEARYE bulletins. The bulletins can be read by any user signed onto either CYBER system and are posted in printed form at the RJE sites.

Since HEARYE bulletins are available on-line, they represent the fastest method available to us for contacting our user community. There are no time delays introduced by reproducing or distributing the message. The system does, however, assume that users will take an active role, regularly checking for news bulletins.

In special circumstances we issue a banner at signon advising users to "TYPE HEARYE FOR NEWS." However, since we sometimes issue bulletins daily, we do this only in emergency situations so as not to dilute the impact of the banner.

Experience has taught us that many users never read HEARYE bulletins that directly affect them. The purpose of this article is to tell you about some of the useful forms of the HEARYE command. Secondly, we want to tell you how to package its use so that invoking it is convenient and automatic.

In the simplest case, you can read news bulletins by entering the command:

```
HEARYE.
```

All news bulletins will begin listing at your terminal, starting with the most recent and continuing until you press the S key or enter STOP to terminate the program. This form of the command requires only one step, but it gives you no control over what is listed. If you want to be more selective, enter the command:

```
HEARYE/F.
```

This command will list only headlines which accompany the news bulletins. Each headline will be preceded by the date on which the bulletin was issued and a keyword. The keyword can be specified on the HEARYE command to selectively retrieve the text of the news bulletin. For example, if the keyword RATES has been assigned to a headline which looks interesting, the command

```
HEARYE,RATES
```

can be used to retrieve the full text of the bulletin.

Finally, when ***/S=−nDAYS*** is appended to the command, all news bulletins except those issued in the last ***n*** days are ignored. For example:

```
HEARYE/S=−3DAYS
```

will list only news bulletins issued during the last three days. You can also use ***/S=−nDAYS*** together with either of the two alternate forms of the command noted above. For example:

```
HEARYE/F/S=—3DAYS
HEARYE/S=—3DAYS,RATES
```

To make checking for new HEARYE bulletins automatic, we suggest that you create a procedure file which is invoked at every signon. For those of you not familiar with the concept, a procedure file is a list of system commands stored in a file which can be executed as a group with a single command.

For example, suppose that you have created a permanent indirect access file named PROCFIL which contains the following record:

```
.PROC,START.
$BILL,chrg,proj.
$HEARYE/F/S=—3DAYS.
$REVERT.
```

Note that these lines must be placed in a separate section of file PROCFIL. (In the simplest case this means that the file contains only these lines.) Note also that the name PROCFIL was chosen purposefully. It must be used as the file name if the remainder of this article is to apply to your procedure file.

You can then invoke this procedure file at signon instead of issuing the usual BILL or CHARGE command. Invoking the procedure will result in the execution of the required BILL command followed by a HEARYE command which will list the headlines of all news bulletins issued during the last three days. If you regularly enter other commands at signon, you can add these to the procedure file as well.

To invoke the procedure file, enter the command:

```
BEGIN,START.
```

If you're at a time-sharing terminal, you can abbreviate this to

```
—START.
```

For the ultimate in abbreviation, if the above procedure file is the first or only record in the file PROCFIL, you can invoke it at signon on a time-sharing terminal by entering a dash, i.e.,

```
—
```

when responding to the prompt RECOVER/CHARGE.

Procedure files are a feature of the CYBER Control Language (CCL). CCL provides more elaborate facilities that you might want to investigate. You can, for instance, have more than one procedure in a file PROCFIL and can pass parameters to procedures at the time they are invoked. There are CCL statements which can make use of passed parameters to govern the sequence of statements executed. CCL is documented in the *NOS Reference Manual, Volume I,* Chapter 4.

RUN A BATCH JOB - SAVE A SERVICE UNIT

Reprinted from *OFF-LINE*, June 1982.

Are you aware that it costs less to run a job in batch than in time-sharing? The cost reduction varies, depending on the amount of central memory used. Compared to its cost when run as a time-sharing job, a job requiring a small amount of core will cost about 10% less when run as a batch job. A job which uses the maximum possible core will cost about 40% less.

For many users, "batch" and "time-sharing" are elusive terms. Understanding is clouded by the fact that there is a "batch subsystem" available to time-sharing jobs and that a batch job can be submitted from a time-sharing job. We'll try to clear away some of the confusion.

The term "batch job" refers to a complete jobstream - job card, signon sequence, system commands, and necessary input - which is presented to the system to be processed at its convenience. The commands in the jobstream are executed in sequence. Input which would normally come from a terminal is read from the jobstream. Output which would normally go to a terminal goes, instead, to a file which is printed or routed to the FETCH queue (user's choice). The system uses batch jobs to fill idle moments, shunting them aside when a user at a time-sharing terminal enters a command. This accounts for the reduced cost.

A batch job can be submitted either from a card reader or from a time-sharing terminal. Since most batch jobs originate from a time-sharing terminal, we'll treat only this mode of submission.

To submit a batch job from a time-sharing terminal you must first create a file (we'll assume you've named it TINJOB) which contains the jobstream. Then you present the contents of the file to the system to be run as a batch job with the command:

```
SENDJOB,TINJOB.
```

In the simplest case, file TINJOB would contain, in sequence, two formatting directives interpreted by SENDJOB, a job statement, and system commands. For example:

```
/JOB
/NOSEQ
WIZARD.                          Job statement
SIGNON(123456789)                Signon
MYCODE.                          Password
BILL,CSO,PS3445.                 Accounting statement
PRINT/RJE=location.              Print output at 'location'
GET,TINMAN.                      Make program file local
FTN,I=TINMAN.                    Compile program (with listing)
LGO.                             Load and execute
```

This job would compile and execute the program found in file TINMAN and print the output at the location specified on the PRINT command.

It is also possible to include program source and execution time input in the file along with the system commands. Each group of input must reside in a separate section of the file. The sections must be arranged in the same order in which they will be read by the job. An example will clarify this. Suppose you build a file containing the following lines (note that a line containing the characters /EOR has been entered in the file to separate sections):

```
/JOB
/NOSEQ
WIZARD.
SIGNON(123456789)
MYCODE.
BILL,CSO,PS3445.
PRINT/FETCH.                          Route output to FETCH queue
FTN.                                  Compile
LGO.                                  Load and Execute
/EOR
      PROGRAM WIZARD(INPUT,OUTPUT,TAPE5=INPUT,TAPE6=OUTPUT)
      DO 10 K=1,3
      READ 100,I
100   FORMAT(I3)
      I = I * 4
10    PRINT 200,I
200   FORMAT(I4)
      END
/EOR
100
200
300
```

This file contains all the required system commands in the first section, the program source in the second, and the input required by the program in the third. When the job is processed, the system will read the first section and place it in a buffer from which the commands will be retrieved and executed. The jobstream is left positioned at the start of the second section. Since the FTN command doesn't name the file containing the program source, the system assumes a file name of INPUT. For batch jobs the file INPUT is equivalenced to the jobstream. The compiler reads the jobstream until it encounters an end-of-section, leaving the jobstream positioned at the start of the third section. Similarly, when the executing program reads from file input, the request is satisfied from the third section of the jobstream.

There are three pitfalls you should look out for. First, every command must begin in column 1. Second, every system command and the password in a batch job must end with a period or right parenthesis. Finally, the signon command must be preceded by a job statement. In its simplest form (used in the above examples), the job statement consists of 1 to 7 alphanumeric characters starting in column 1, followed by a period.

For more information about batch jobs, see the *NOS Reference Manual, Volume 1* and the description of the SENDJOB command contained in the *CSO Utilities Manual.*

INDEX

POLICY

SYSTEM NOTES

CYBER

IBM

STATISTICAL SERVICES

MATHEMATICAL SERVICES

FEATURE ARTICLES

DOCUMENTATION

MISCELLANEOUS

OFF-LINE's Mailing List

If you wish to be placed on our mailing list for future issues of *OFF-LINE*, if you wish to be removed from the list, or if you wish to enter an address correction, please complete and return this page. **(Current subscribers are kept on the mailing list until a specific request for removal is received, or until a mailing is returned as undeliverable.)**

Check one:

- ☐ Place my name on mailing list
- ☐ Make the following corrections or changes
- ☐ Delete my name from mailing list

_____

First name -- Initial -- Last Name

UI Address:

_____

Department

_____

Room - Building

Off-campus Address:

_____

Organization or Company (if applicable)

_____

Street Address

_____

City -- State -- Zip Code

If address correction, give old address and zip code below.

_____

_____

_____

SEND TO:

OFF-LINE
150 Digital Computer Laboratory
University of Illinois at Urbana-Champaign
1304 West Springfield Avenue
Urbana, Illinois 61801

Computing
Services
Office

Off-Line

University of Illinois at Urbana-Champaign VOL. 10, NO. 10 October 1982

EDITOR: Lynn Bilger
PHONE: (217) 333-6236
150 Digital Computer Lab
1304 W. Springfield Ave.
Urbana, Illinois 61801

CSO DIRECTORY - STAFF AND SERVICES

Administrative

Director	George Badger	150	DCL	333-4103
Business Manager	Stanley Rankin	150	DCL	333-6530
Secretary	Joyce McCabe	150	DCL	333-1637

User Services and Hardware/Software Support

User Accounting		**1208**	**W Springfield**	**333-7752**
Distribution Office		**1208**	**W Springfield**	**333-6760**
Systems Consulting		**1208**	**W Springfield**	**333-6133**
Statistical Services Consulting		**85**	**Comm West**	**333-2170**
Text Processing Consulting		**118**	**DCL**	**333-7318**
Terminal Repair Service		**150**	**DCL**	**333-0969**
CYBER Dial-up Numbers	**CYBER 175**	**110-300**	**baud**	**333-4000**
	CYBER 175	**1200**	**baud**	**333-4001**
	CYBER 174	**110-300**	**baud**	**333-4004**
Asst Dir User Services	Robert Penka	173	DCL	333-4709
Asst Dir Systems and Operations	Sandra Moy	177	DCL	333-4703
Asst Dir Development	J. M. Randal	175	DCL	333-9772
Asst Dir Engineering and Hardware Selection	Cliff Carter	195	DCL	333-3723
Manager Accounting Services	Gary Bouck	1208	W Springfield	333-7752
Manager Statistical Services	Beth Richardson	189	DCL	333-6276
Documentation	Lynn Bilger	120	DCL	333-6236
CYBER-IBM Operations	Jack Knott	194a	DCL	333-6562
UNIX Operations	Debbie Hudson	123	DCL	333-8150
Telecommunications	Tom Kerkering	18	DCL	333-0816
Laboratory Support Project	Mike Gardner	18	DCL	333-7904
RJE Operations North	Rex Duzan	162	DCL	333-6285
RJE Operations South	Don McCabe	1208	W Springfield	333-2171 333-7752

RJE Sites (see Reference Guide RF-0.3 for operating hours)

Agriculture	M103	Turner Hall	333-8170
Chemistry	153	Noyes Lab	333-1728
Commerce West	70	Comm West	333-4500
CRH Snack Bar	120	Snack Bar	333-1851
DCL Routing Room	14	DCL	333-6203
Electrical Engineering	146	EEB	333-4936
Florida Ave Res Hall		FAR	333-2695
Illinois St Res Hall		ISR	333-0307
Mechanical Engineering	65	MEB	333-1430
Psychology	453	Psych Bldg.	333-7531
Social Science	202	Lincoln Hall	333-0309

OFF-LINE is the monthly newsletter of the Computing Services Office at the University of Illinois at Urbana-Champaign. Unless otherwise indicated, permission to reprint is freely granted, provided that the author, if named, and the Computing Services Office (CSO) are credited.

CSO operates a CDC CYBER 175 with 262K words of central memory and a CDC CYBER 174 with 196K words of central memory. The 175 and 174 run under the NOS Operating System and share 512K words of ECS. The 175 serves over 200 simultaneously active text and graphics terminals and the 174 serves over 100 simultaneously active terminals. CSO also operates an IBM 4341 with 4 million bytes of memory running HASP-OS/MVT under VM. In addition, CSO operates a DEC VAX 11/780 with 4 million bytes of memory, running under the UNIX Operating System and driving a GSI CAT-8 phototypesetter.

Operating Hours (see HEARYE ,, SCHEDUL for exceptions):

	CYBERS 174/175	IBM
M-F	8 am to 6 am	8 am to 6 am
SAT	8 AM - Midnight	8 AM - 6 AM
SUN	Noon - 6 AM	Noon - 6 AM

POLICY

SERVICE UNIT PRICE REDUCED

In keeping with our pattern of periodic reductions of the price of the service unit, the price will be reduced, effective 1 November 1982, to $0.59 from the present value of $0.62. In addition, the charging rate for CYBER bulk computing will be reduced from 25% of normal charge to 18%. The rates for bulk utilization on the IBM machine have been replaced by the new billing algorithm announced in last month's *OFF-LINE*.

PLANNING OF FUTURE SERVICES

The heavy financial commitments to the current CYBER equipment will be concluding in about two years. With this in mind, CSO is entering into the planning of services beyond that point, including not only the computers, but the service strategies regarding networks, specialized software and various other alternatives for service expansion that are available.

A request has been circulated to Deans, Directors and Department Heads requesting that they inform us of the needs for both instructional and research computing as viewed by their faculty. Initial responses have been requested for October 18, but we welcome both further information from departments and written information from individual faculty or students.

The various CSO committees, in particular the Liaison Committee, the Committee on Instructional Computer Use and the Computer Policy Committee, will receive this information for use in their discussion of priorities for various types of service.

Information regarding the planning process will be carried in *OFF-LINE*.

In addition to the various services, we solicit suggestions regarding the operating policies which most affect the productivity of the service. Discussions of some of these policies will be a part of the planning process and some discussion on them will appear in future issues of *OFF-LINE*.

CYBER

FILE TRANSFER SERVICE

CSO is now offering a floppy disk and DEC tape file transfer service. Bring your floppy or DEC tape to Room 123 DCL between 9 AM - 4 PM, or call 333-8150 to make arrangements for transferring any portion of your floppy disk or DEC tape to your CYBER disk area. We can also move your CYBER disk files onto your floppy disk or DEC tape. Note, however, that the files must be character data rather than binary.

The mediums we support must be in DEC RT-11 format. We support:

DEC Tape 2 (cassette tape)

RXO2 floppy, single or double density (this is compatible with the IBM 2740 format)

AED 6200P floppy

RL02 hard disk (will eventually be available)

Note that RSX-11 systems can generate RT-11 format floppies.

At some time in the future, we will charge for this service. During our break-in period, it will be a free service.

If you have any questions, please contact:

Debbie Hudson
123 DCL
phone: 333-8150

DI-3000 GRAPHICS SOFTWARE FOR THE CYBERS

Within a few months we will be installing DI-3000, a commercial implementation of the ACM/SIGGRAPH CORE system. DI-3000, a product of Precision Visuals, Inc., has been installed at over 200 sites. DI-3000 is a device-independent library of graphics support subroutines, providing full support for graphics primitives such as lines, polygons, and text as well as a wide variety of attributes for each. The package also supports 3-D viewing, retained segments, graphics input, inquiry functions, good error reporting, and powerful debugging capabilities.

DI-3000 promises reliable, easy-to-use software, with a high level of vendor support. The documentation is well written and contains numerous examples.

Articles providing more details on DI-3000 will appear in *OFF-LINE* when the software is installed and its availability is announced.

STATISTICAL SERVICES

SAS RELEASE 79.6 INSTALLED

Release 79.6 of SAS has been installed on the IBM 4341. Forty-one problems that existed in Release 79.5 have been fixed (a list of these fixed problems is available in the CSO South Consulting Office, Room 85 Commerce West). The most important fix is to the probability levels of the F-tests in PROC REG which were wrong in the 79.5 release.

There are also some minor enhancements. The most useful enhancement is the six new multiple comparison methods incorporated in the MEANS statement in PROC ANOVA and PROC GLM. These six new methods are:

BON (Bonferroni T-test)
LSD (Fisher's least-significant difference test)
SCHEFFE (Scheffe's multiple comparison procedure)
SNK (Student-Newman-Keul's multiple range test)
SMM (Hochberg's studentized maximum modulus test)
TUKEY (Tukey's studentized range test (HSD))

The general format for employing any of the above options is as follows:

MEANS factor1 factor2 ... factorn/option ALPHA=level;

where ***option*** is any one of the six methods mentioned above, and ***level*** is a number between 0 and 1. For example:

PROC GLM; CLASS A B; MODEL Y=A|B; MEANS A B/TUKEY ALPHA=.05;

NOTE: In the above example, the expression Y=A|B shows A and B separated by a vertical bar. Since some terminal keyboards do not have a vertical bar key, you may use the exclamation mark in its place (i.e., Y=A!B).

SAS USERS, TELL US WHO YOU ARE!

We are continuing to develop our list of local users of the SAS Statistical Package as announced in early August. We will be using this list to inform local SAS users of SAS enhancements, changes, and/or training seminars. If you would like to be put on this list and have not yet contacted us, please specify the following:

Your Name
ID Number
Telephone Number
Campus Address
Departmental Staff Title

in a message to Greg Kesner via the CYBER command: TELL,KESNER.

SPSS-CDC RELEASE 8.3 AVAILABLE

SPSS-CDC Release 8.3 and SPSS/ONLINE Release 8.3 are now available on the CDC CYBER 174 and CDC CYBER 175 computers. The SPSS statistical package which we run on our CDC computers is an adaptation of the SPSS Batch System which SPSS Inc. in Chicago, Illinois distributes to IBM installations. The SPSS Batch System has been converted to run on CDC computers by Northwestern University in Evanston, Illinois. Northwestern distributes the converted software to CDC installations. In addition, Northwestern distributes the SPSS/ONLINE system which provides editing and program execution facilities for SPSS users in one package. Several years ago we received and installed Release 8.0 which represented an extensive change in SPSS and required a new set of documentation. Releases 8.1 and 8.2 which followed were corrective code releases which did not alter user specifications of the software, and required no changes in documentation. The 8.3 release, however, represents a significant upgrade from the previous release. There are over 70 new bug fixes. Program changes and the addition of new features have brought about an extensive change in the documentation. New features in Release 8.3 include a new MANOVA, a new version of SURVIVAL, subfile generation via SORT CASES, a separate optional matrix input file, as well as extensions to REPORT, SCATTERGRAM, and other procedures. SPSS users are encouraged to immediately begin using the 8.3 release and to obtain the new documentation. Direct questions and comments to Beth Richardson, 189 DCL (3-6276), or TELL,UN=STAT.

How to Access the SPSS Software

Currently, the various SPSS-CDC systems may be accessed as follows:

GRAB,SPSS/F	accesses SPSS-CDC Release 8.3
GRAB,SPSS	accesses SPSS-CDC Release 8.2
GRAB,SPSSONL/F	accesses SPSS/ONLINE Release 8.3
GRAB,SPSSONL	accesses SPSS/ONLINE Release 8.2

On October 18, 1982, the following changes will go into effect:

GRAB,SPSS	will access SPSS-CDC Release 8.3
GRAB,SPSS/P	will access SPSS-CDC Release 8.2
GRAB,SPSSONL	will access SPSS/ONLINE Release 8.3
GRAB,SPSSONL/P	will access SPSS/ONLINE Release 8.2

Program Changes and New Features of Release 8.3

The following describes selected program changes and new features:

END INPUT DATA

The END INPUT DATA statement can be used to delimit the data when the data are included in the same file with the SPSS program statements. The use of this statement can be illustrated with an example. The following is a complete set of program statements for obtaining descriptive statistics on the three variables SEX, AGE and INCOME when the data lines are stored in the same file as the SPSS program statements.

```
VARIABLE LIST  SEX,AGE,INCOME
INPUT FORMAT   (F1,F2,F5)
CONDESCRIPTIVE ALL
READ INPUT DATA
   the data lines go here
END INPUT DATA
```

FINISH, INPUT MEDIUM, N OF CASES, READ INPUT DATA, READ MATRIX

These statements are no longer required for running SPSS programs when the data lines are in separate file from the SPSS program statements. For example, the following is a complete set of program statements for obtaining descriptive statistics on the three variables SEX, AGE and INCOME when the data lines are in a separate file from the SPSS program statements.

```
VARIABLE LIST  SEX,AGE,INCOME
INPUT FORMAT   (F1,F2,F5)
CONDESCRIPTIVE ALL
```

When the data lines are in the same file as the SPSS program statements you need to use READ INPUT DATA and END INPUT DATA as shown above, but no longer are required to use FINISH, INPUT MEDIUM, or N OF CASES.

MANOVA

This procedure has been rewritten for the 8.3 release. Programs set up to run on the old version need to be changed. In particular, the STATISTICS and OPTIONS have been eliminated in favor of keyword phrases.

MATRIX INPUT

A new SPSS control statement parameter **M=lfn** has been implemented which allows matrix input to be on a file other than the SPSS program statement file. Specify the name of the local file containing the matrix input after the M= parameter on the SPSS control statement.

N OF CASES and SUBFILE LIST

These statements may appear at any point in the program and any number of these statements may appear in a program. Their usage allows processing of small sets of data from large files

for testing purposes.

Scratch File Names

The SPSS internal scratch file names have changed from XXSPSn to ZZSPSn.

SORT CASES

This facility has been rewritten for this release. It can group cases and define new subfiles according to the values of variables on the data file.

For example, the statement

```
SORT CASES  RACE,SEX/SUBFILES
```

will cause SPSS to sort the data on RACE and SEX and define all the cases having any particular combination of values of these two variables as a subfile. There will be subfiles created for white males, white females, black males, black females, etc.

SURVIVAL

This release contains a completely new version of the SURVIVAL procedure. Programs set up to run with the old version may need to be modified to work with the new version. In particular, the OPTION numbers have changed.

Documentation

The following defines and describes the documentation appropriate for using SPSS-CDC Release 8.3, explains how the various documents are related, and explains how to obtain the documentation.

The McGraw-Hill SPSS 2nd Edition Book

This book documents the SPSS package as released by SPSS Inc. through Release 6. It is needed by SPSS-CDC users but must be used in conjunction with the other documentation described in this article. It lacks information about changes in the package since Release 6, and lacks information specific to the CDC conversion of SPSS. Appendix F of this book which concerns the CDC implementation should not be used because it is out of date and inaccurate. The SPSS-CDC Release 8.3 Update Manual should be used instead of Appendix F.

The McGraw-Hill SPSS Update 7-9 Book

This book documents the enhancements to the SPSS package as released by SPSS Inc. which were incorporated in Releases 7, 8 and 9. This book is needed by SPSS-CDC users but must be used in conjunction with the other documentation described in this article. The following chapters of this book should be ignored by SPSS-CDC users because they describe features which have not been implemented in Release 8.3:

Topic	Ignore - Not Available
BOX-JENKINS	Chapter 2
NEW REGRESSION	Chapter 3
SPSS Graphics Option	Chapter 4
Extended Input Formats	Chapter 13

The following sections of this book should be ignored by SPSS-CDC users because the features described are documented more accurately for the CDC implementation in the SPSS-CDC Release 8.3 Update Manual:

Topic	Ignore	See in SPSS-CDC Update Manual
GET SCSS/SAVE SCSS	Chapter 10	pages 27-32
LAG	Chapter 11 - pgs 280-283	page 9 and page 41
Other Changes	Chapter 19	see table of contents for topic
CDC Version	Appendix D	see table of contents for topic

The SPSS-CDC Release 8.3 Update Manual

This manual documents the differences between the features of SPSS which are available in SPSS-CDC Release 8.3 and the features of SPSS which are described in the pair of books *SPSS Second Edition* and *SPSS Update 7-9.* However, it does not include detailed documentation about the many statistical procedures which are unique to the CDC conversion of SPSS and does not document the extensions to the REGRESSION procedure which are available.

Other SPSS-CDC Documents

Separate writeups are available which document the following statistical procedures which are unique to the CDC conversion of SPSS:

G3SLS (Generalized and Three Stage Least Squares)
JFACTOR (Joreskog's Factor Analysis)
NONLINEAR (Nonlinear Regression)
PLOT (ZETA Plotting)
SPECTRAL (Spectral Analysis)
SUMMARY TABLES
TETRACHORIC (Tetrachoric Correlations)

In addition, a separate writeup is available which documents the many extensions to the REGRESSION procedure in SPSS-CDC Release 8.3. (These extensions comprise a subset of improvements that were incorporated in the IBM SPSS procedure named NEW REGRESSION.)

The SPSS/ONLINE Manual

This manual is needed only by SPSSONL users. SPSSONL, written and distributed by Northwestern, provides editing and program execution facilities in one package.

The SPSS-CDC Release 8.0 to 8.3 Changes Bulletin

This bulletin summarizes the major program differences between Release 8.0 and Release 8.3 of the SPSS-CDC package. The new features, changes and enhancements are briefly described and the related document in which the feature is completely described is noted.

The SPSS-CDC Release 8.3 Documentation Cross-Reference

This manual defines and describes the documentation which corresponds to the 8.3 release of SPSS-CDC, and explains how the various documents are related.

Changes in Documentation with the 8.3 Release of SPSS-CDC

The following information is of interest to those users who have previously purchased the Release 8.0 SPSS-CDC documentation and would like to know how it changes with the Release 8.3 implementation.

The following writeups have not changed:

REGRESSION
SPECTRAL
SUMMARY TABLES
TETRACHORIC

The following writeups have been modified:

G3SLS
JFACTOR
NONLINEAR
PLOT

The following writeups have been dropped:

Procedure	See in SPSS Update 7-9	See in SPSS-CDC Update
ANOVA	pages 305,306	
MANOVA	Chapter 1 pgs 1-79	pages 43-52
MULT RESPONSE	Chapter 8 pgs 240-247	pages 54-55
NPAR TESTS	Chapter 7 pgs 220-239	pages 56-57
RELIABILITY	Chapter 9 pgs 248-267	pages 64-65
REPORT	Chapter 5 pgs 166-204	pages 68-73
SURVIVAL	Chapter 6 pgs 205-219	pages 78-79

In addition, there is a one-page change to the SPSS/ONLINE manual.

How to Obtain Documentation

The following SPSS-CDC documents are available free at the CSO South Consulting Office. This office is located in Room 85 Commerce West and is open Monday through Friday from 9AM to 5PM.

SPSS-CDC Release 8.3 Update Manual
SPSS-CDC Release 8.0 to 8.3 Changes Bulletin
SPSS-CDC Release 8.3 Documentation Cross-Reference
SPSS/ONLINE Release 8.3 Update Page
List of errors fixed in SPSS-CDC Release 8.3
List of known errors in SPSS-CDC Release 8.3

In addition, a package is available free in the CSO South Consulting Office for those SPSS-CDC users who have previously obtained the Release 8.0 SPSS-CDC documentation and wish to update their documentation for the 8.3 release. The package contains just those documents which were modified for the 8.3 release. This package contains:

Documentation for the statistical procedures G3SLS, JFACTOR, NONLINEAR, PLOT
SPSS-CDC Release 8.3 Update Manual
SPSS-CDC Release 8.0 to 8.3 Changes Bulletin
SPSS-CDC Release 8.3 Documentation Cross-Reference
SPSS/ONLINE Release 8.3 Update Page

The following SPSS-CDC documents are available for sale at the CS Distribution Office. This office is located at 1208 W. Springfield, Urbana and is open Monday through Friday 8-12AM and 1-5PM:

SPSS Second Edition book sold together with the SPSS Update 7-9 book	$18.95
SPSS Update 7-9 book	$10.00
SPSS-CDC Release 8.3 Update Package which includes:	$3.00
Documentation for statistical procedures G3SLS, JFACTOR, NONLINEAR, PLOT, REGRESSION, SPECTRAL, SUMMARY TABLES, TETRACHORIC. In addition, the following documents are included free in the Update Package for your convenience: SPSS-CDC Release 8.3 Update Manual, SPSS-CDC Release 8.0 to 8.3 Changes Bulletin, and SPSS-CDC Release 8.3 Documentation Cross-Reference.	
SPSS/ONLINE Manual	$2.00

The following SPSS-CDC Release 8.3 documents are available on-line:

File name	Description
ANNOUNC	CSO announcement about Release 8.3
CHANGES	SPSS-CDC Release 8.0 to 8.3 Changes Bulletin
CROSREF	SPSS-CDC Release 8.3 Documentation Cross-Reference
ERRFIX	List of errors fixed in SPSS-CDC Release 8.3
ERRLIST	List of known errors in SPSS-CDC Release 8.3
INDEX	Index of files in UN=SPSSDOC
JFACDOC	Procedure JFACTOR Documentation
SUMRTAB	Procedure SUMMARY TABLES Documentation
UPDATE	SPSS-CDC Release 8.3 Update Manual

These files are in UN=SPSSDOC and are ASCII files with carriage control. Each file, except the UPDATE file, can be viewed at a terminal by issuing the command:

TYPE,file/G/UN=SPSSDOC/AS/CC/EJ

where *file* is the name of one of the documentation files. The UPDATE file is direct access and can be viewed at a terminal by issuing the command:

TYPE,UPDATE/ATT/UN=SPSSDOC/AS/CC/EJ

FEATURE ARTICLES

FORTRAN FUTURES

This article is part of a series on the work of the FORTRAN Standards Committee X3J3 in producing the next revision of the FORTRAN standard. The reader is reminded that the features described in this article are not a part of any FORTRAN compiler currently available, but rather are proposed requirements for FORTRAN processors produced in the late 1980's through the mid 1990's. Although every effort has been made to accurately describe the current position of X3J3 on these matters, the development of a revision to the FORTRAN standard is an evolutionary process, and these proposals may be subject to refinement, revision, or even retraction. Comments on these proposals may be given to Kurt Hirchert of the CSO Systems Consulting staff, who is a member of X3J3.

Over the past year, we have examined the new features which have been adopted for inclusion in the next revision of the FORTRAN standard. Although there is likely to be another year of refinement of these features before X3J3 undertakes the arduous task of writing the formal text of the next revision, the basic outline of these features is already fairly well known. There are, however, a few areas still under investigation in which no proposals have yet been adopted, but in which proposals may be adopted in future sessions.

Perhaps the most ambitious undertaking is the development of a facility to handle errors/exceptions/events in FORTRAN. Language standards have long described how programs behave when everything is "right," but few language standards have addressed what happens when the basic assumptions of the language are violated. The need for more robust programs, especially in the area of real-time programming, has resulted in increased interest in means of handling such errors, especially in newer languages such as Ada. The principal work in developing a standard error handling facility for FORTRAN is being done by a subcommittee of EWICS, a committee on real-time programming based in Europe.

A second area which has been of great interest lately is the provision of a facility for user-defined operators as well as functions. Although functions and operators offer the same theoretical capabilities, operators are more convenient notationally. For example, if one wants to implement a package to support interval arithmetic using a data structure to represent intervals, it will be much more convenient to use if one can write "A+ B" to get the sum of intervals than to write "INTERVAL_SUM(A,B)." Thus, it will be easier to use FORTRAN as a FORmula TRANslator in areas of mathematics and science not explicitly supported by the basic language definitions. This facility might also include the ability to define new operator symbols (especially those of the form ".WORD.") and the ability to redefine the meanings of operator symbols already defined in the language.

Much of the reason for work in the third area reflects work already done on the coming revision. Traditionally, FORTRAN compilation units have been almost completely independent, so that if the same entity is being accessed by more than one such unit (e.g., as is typically the case for variables in common) the attributes of the entity have to be respecified in each compilation unit. The next revision includes features which require dependent (although still separate) compilation. These features help eliminate the need to respecify the attributes of entities in different compilation units, but in the process, introduce the possibility of conflict between names chosen in different compilation units. X3J3 has been actively considering two different schemes for resolving such conflicts, but has not reached a consensus on which approach should

be followed.

Finally, X3J3 has been looking at ways in which the language can be used to implement "modules" of language facilities. Effectively speaking, proposals in this area have extensions or modifications of the SHARED DATA facility, in some cases changing the name of the facility.

MISCELLANEOUS

STATISTICAL CONSULTING

The Mathematical and Statistical Consulting Committee (MSCC) provides assistance to members of the University Community on statistical and mathematical problems in their research work. The 1982-83 director of the MSCC is Robert A. Wijsman who is assisted by graduate statistics students Kenton Juhlin, Bruce Richardson, and Robin Shealy.

The service is offered free for problems that can be settled within several hours. To use the service, drop in at 215 Altgeld Hall, or telephone 333-9317 during the following hours: Monday, Wednesday and Friday, 3-5 PM; Tuesday and Thursday, 9-12 AM and 1-5 PM. Outside of these regularly scheduled hours, try Wijsman's number (333-1909) or leave a message with the Math Office (333-3350).

Problems that require a considerable amount of consulting time and effort are also welcome. In such cases, the consultee may be asked to contribute financially into an account that has recently been set up to help defray the cost of running the MSCC. Faculty members anticipating the need for substantial statistical help (for themselves or their students) are urged to try to obtain funds for this purpose (e.g., include funds as a budget item for statistical consulting in a research proposal).

1984 INTERNATIONAL COMMUNICATIONS CONFERENCE

Amsterdam will host the 1984 IEEE International Conference on Communications, ICC'84, which will be held from May 14-17, 1984 in the RAI Conference Center, Amsterdam, the Netherlands.

The conference will carry the theme: Science, Systems and Service. Innovation research, applications and communication policy issues will be emphasized. A state-of-the-art technical exhibition will be included. Special sessions will be organized, including the following topics: scientific basis for future communications; field trials and their outcomes; and reliability and availability.

For further information, contact:

Dr. T.A.C.M. Claasen
Secretary of the Executive Committee
Philips' Research Laboratories
P.O. Box 218
5800 MD Eindhoven
The Netherlands
Telephone: (31) 40-742131/742236

HELP WANTED

INTERESTED IN PROGRAMMING MICROCOMPUTERS?

A student who can program a microcomputer using a Cromemco Z80 Assembler is needed for part-time work at the Lincoln Avenue Dairy Automation Center. The programs needed are for realtime data collection and analysis. The programming desired will require a thorough knowledge of ASSEMBLY. An operating program, in use, needs some changes and additions to it.

The student needs time periods of 2 hours or more for a total of 12 to 20 hrs per week. The microcomputer system is a Cromemco Z2D with 64kb memory, 2 DS/DD 5-inch floppy drives, 5 serial ports, 4 parallel ports, a printer and a realtime clock.

Pay for the selected student will be determined by his or her qualifications (range $3.50 - $6.00 per hour).

If interested, contact:

H. B. Puckett
Room 226 Agr. Engr.
Telephone: 333-0808

OFF-LINE's Mailing List

If you wish to be placed on our mailing list for future issues of *OFF-LINE*, if you wish to be removed from the list, or if you wish to enter an address correction, please complete and return this page. **(Current subscribers are kept on the mailing list until a specific request for removal is received, or until a mailing is returned as undeliverable.)**

Check one:

- ☐ Place my name on mailing list
- ☐ Make the following corrections or changes
- ☐ Delete my name from mailing list

_____

First name -- Initial -- Last Name

UI Address:

_____

Department

_____

Room - Building

Off-campus Address:

_____

Organization or Company (if applicable)

_____

Street Address

_____

City -- State -- Zip Code

If address correction, give old address and zip code below.

_____

_____

_____

SEND TO:

OFF-LINE
150 Digital Computer Laboratory
University of Illinois at Urbana-Champaign
1304 West Springfield Avenue
Urbana, Illinois 61801

Computing Services Office

Off-Line

University of Illinois at Urbana-Champaign VOL. 10, NO. 11 November 1982

EDITOR: Lynn Bilger
PHONE: (217) 333-6236
120 Digital Computer Lab
1304 W. Springfield Ave.
Urbana, Illinois 61801

Contents

CSO DIRECTORY - STAFF AND SERVICES

Administrative

Director	George Badger	150	DCL	333-4103
Business Manager	Stanley Rankin	150	DCL	333-6530
Secretary	Joyce McCabe	150	DCL	333-1637

User Services and Hardware/Software Support

User Accounting		**1208**	**W Springfield**	**333-7752**
Distribution Office		**1208**	**W Springfield**	**333-6760**
Systems Consulting		**1208**	**W Springfield**	**333-6133**
Statistical Services Consulting		**85**	**Comm West**	**333-2170**
Text Processing Consulting		**118**	**DCL**	**333-7318**
Terminal Repair Service		**150**	**DCL**	**333-0969**
CYBER Dial-up Numbers	**CYBER 175**	**110-300**	**baud**	**333-4000**
	CYBER 175	**1200**	**baud**	**333-4001**
	CYBER 174	**110-300**	**baud**	**333-4004**
Asst Dir User Services	Robert Penka	173	DCL	333-4709
Asst Dir Systems and Operations	Sandra Moy	177	DCL	333-4703
Asst Dir Development	J. M. Randal	175	DCL	333-9772
Asst Dir Engineering and Hardware Selection	Cliff Carter	195	DCL	333-3723
Manager Accounting Services	Gary Bouck	1208	W Springfield	333-7752
Manager Statistical Services	Beth Richardson	189	DCL	333-6276
Documentation	Lynn Bilger	120	DCL	333-6236
CYBER-IBM Operations	Jack Knott	194a	DCL	333-6562
UNIX Operations	Debbie Hudson	123	DCL	333-8150
Laboratory Support Project	Mike Gardner	18	DCL	333-7904
RJE Operations North	Rex Duzan	162	DCL	333-6285
RJE Operations South	Don McCabe	1208	W Springfield	333-2171 333-7752

RJE Sites (see Reference Guide RF-0.3 for operating hours)

Agriculture	M103	Turner Hall	333-8170
Chemistry	153	Noyes Lab	333-1728
Commerce West	70	Comm West	333-4500
CRH Snack Bar	120	Snack Bar	333-1851
DCL Routing Room	14	DCL	333-6203
Electrical Engineering	146	EEB	333-4936
Florida Ave Res Hall		FAR	333-2695
Illinois St Res Hall		ISR	333-0307
Mechanical Engineering	65	MEB	333-1430
Psychology	453	Psych Bldg.	333-7531
Social Science	202	Lincoln Hall	333-0309

OFF-LINE is the monthly newsletter of the Computing Services Office at the University of Illinois at Urbana-Champaign. Unless otherwise indicated, permission to reprint is freely granted, provided that the author, if named, and the Computing Services Office (CSO) are credited. Information in this issue is current as of October 25, 1982.

CSO operates a CDC CYBER 175 with 262K words of central memory and a CDC CYBER 174 with 196K words of central memory. The 175 and 174 run under the NOS Operating System and share 512K words of ECS. The 175 serves over 200 simultaneously active text and graphics terminals and the 174 serves over 100 simultaneously active terminals. CSO also operates an IBM 4341 with 4 million bytes of memory running HASP-OS/MVT under VM. In addition, CSO operates a DEC VAX 11/780 with 4 million bytes of memory, running under the UNIX Operating System and driving a GSI CAT-8 phototypesetter.

Operating Hours (see HEARYE,SCHEDUL for exceptions):

	CYBERS 174/175	IBM
M-F	8 am to 6 am	8 am to 6 am
SAT	8 AM - Midnight	8 AM - 6 AM
SUN	Noon - 6 AM	Noon - 6 AM

POLICY

PROTECT YOUR COMPUTER FUNDS

Thefts of computer funds, if they are to occur at all, usually happen late in the semester. Now is a good time to consider what you can do to protect your computer funds.

You should give your signon the same consideration that you would a credit card. Some pointers:

- Change your password frequently. Use some creativity when choosing a password. Avoid obvious possibilities such as your initials, spouse's name, etc.

- Don't keep signon information in publicly accessible files.

- When signing onto a CYBER terminal, do not enter your ID and password on the same line. Enter your ID by itself and wait for the computer to prompt you for your password. If your terminal is operating in full duplex mode (true of all CSO terminals and assumed by the system to be true of all private terminals as well), your password won't print when you type it.

- Be careful with any card decks which contain your signon information.

- Observe the CYBER signon message, printed at your terminal or placed in your dayfile, which gives the time and date of the last recorded use of your signon. This message can quickly alert you to unauthorized use of your signon.

- If you are a Project Manager, you should be alert to misuse of your project. You should periodically change the codeword which is requested every time you use the MANAGE program to add user numbers to your project or to change limits or allocations for user numbers already assigned to your project. We know of several instances in which a thief gained access to a project manager's signon and proceeded to use MANAGE to add new bogus user numbers to the project for his own use.

Your signon is your responsibility and only you can take care of it. CSO will refund stolen computer funds only if CSO is at fault. We will, however, make every effort to help you identify the offender, using the logs maintained by the system.

Cases involving theft of computer funds have, in the past, been turned over to campus disciplinary authorities who have always required that restitution be made. One case which involved large sums of money was, additionally, turned over to the State Attorney's office for criminal prosecution.

ABUSE OF MESSAGE FACILITY

Over the past few months we have had several incidents involving use of the CYBER MESSAGE facility which require comment. pg The MESSAGE facility should not be used for advertising or sending other forms of junk mail to large lists of people. It should not be used for promoting political meetings or other group activities. The intention of the facility is to allow communication between users of the system who have a common interest or who are collaborating on projects, or communication between faculty and students when related to educational activities.

Although it is possible to use system software to enforce policies that would limit such abuse, we would prefer to leave the system as open as possible for constructive use and would appreciate everyone's cooperation towards this objective. If you receive mail which you feel is inappropriate use of the facility, we would appreciate receiving information about it. For example, you could send a copy of the message to us by using:

```
FORWARD,CSO. I consider this to be junk mail.
```

STATISTICAL SERVICES

BMDP 1981 RELEASE AVAILABLE

The June 1981 version of the Biomedical (BMDP) statistical programs is now available on the CDC Cyber 174 and 175 computers. This package is an adaptation of the BMDP-81 package distributed to IBM installations by UCLA's Department of Biomathematics. The conversion and distribution to CDC sites was carried out by the Vogelback Computing Center of Northwestern University, Evanston, Illinois.

Currently, the 1977 version is the default on the Cyber systems and the 1981 version may be accessed only by entering:

```
GRAB,BMDP/FUTURE.
```

On November 15, 1982, the 1981 version that now resides in FUTURE will become the default, and will be the version obtained with:

```
GRAB,BMDP.
```

A brief explanation of Cyber usage may then be obtained with:

```
BMDP,HELP=YES.
```

An on-line document containing further information on the new version may be obtained and printed by entering:

```
WRITEUP,BMDP.
PRINT,BM81DOC/AS/CC/EJ
```

This document contains basic information on the new programs, and new features of old programs, in the 1981 version.

Printed documentation on BMDP is available at the CSO Distribution Office, 1208 West Springfield Ave., in four formats:

1.	*BMDP Statistical Software*, 1981 (The "big" complete manual.)	$17.00
2.	*BMDP User's Digest: A Condensed Guide*	$5.00
3.	*BMDP Control Language Guide*	$2.50
4.	*BMDP Reference Card*	$0.50

Note that numbers 2, 3 and 4 correspond to the 1979 version of BMDP, which is highly, but not completely, compatible with the 1981 version. (The 1979 version remains the default on the IBM system. The 1982 release for IBM has been received and will be installed soon.)

The 1981 release of BMDP contains several new programs as well as enhancements to all others. We recommend that BMDP-79 users read the preface, chapters 1-5, and the relevant program-specific chapters of the BMDP-81 manual before using the 1981 version. None of the new features are mentioned here, only the changes which users of the previous version may encounter.

In general, BMDP-79 control language should work with BMDP-81 with the following known exceptions:

- BMDP1F, 2F and 3F have been discontinued and superseded by BMDP4F which has its own unique directive syntax.

- In pre-1981 versions of BMDP, blanks in data records were considered to be ZERO by default. In BMDP-81, blanks are considered to be MISSING value codes by default. Thus, if your data contains blanks and you run the same analysis using the default 1979 and 1981 settings, you may obtain radically different results. The sentence BLANK=[ZERO | MISSING] can be specified on the INPUT paragraph of the 1981 version to explicitly control the processing of blanks. (See page 42 of the 1981 manual.)

- In the pre-1981 versions of BMDP, extraneous control language was ignored and not reported. In BMDP-81, it is also ignored, but is reported.

- In pre-1981 versions of BMDP, a fatal BMDP execution error resulted in normal termination of the program. In BMDP-81, a fatal execution error results in a CP abort and subsequent system control statements are skipped.

- In the pre-1981 version of BMDP2D, the ESTIMATE sentence was specified in the COUNT paragraph. In the BMDP-81 version of BMDP2D, the ESTIMATE sentence is specified in the PRINT paragraph. (See pages 83 and 85 of the BMDP-81 manual.)

- In the pre-1981 version of BMDPLR, the CELL, SORT, PLOT and HIST sentences were specified in the REGRESS paragraph. In the BMDP-81 version of BMDPLR, these sentences are specified in the PRINT paragraph. (See page 340 of the BMDP-81 manual.)

New programs in the 1979 version were the following:

BMDPLR: Stepwise Logistic Regression.
BMDP8V: General Mixed Model ANOVA -- Equal Cells.
BMDPKM: K-Means Clustering.

New programs in the 1981 version are as follows:

BMDP4F: Two-way and Multiway Frequency Tables -- Measures of Association and the Log-linear model (Complete and Incomplete Tables).
BMDP2L: Survival Analysis with Covariates -- Cox Models.
BMDP8M: Boolean Factor Analysis.
BMDP9M: Linear Scores for Preference Pairs.
BMDP2T: Box-Jenkins Time Series Analysis.
BMDP4V: General Univariate and Multivariate Analysis of Variance and Covariance, Including Repeated Measures.

Note that a "Univariate and Bivariate Spectral Analysis" program, BMDP1T, is described in the 1981 manual, but has not yet been distributed by BMDP and is not currently available on the Cyber system.

Former users of BMDP should examine especially pages viii-x of the Preface to the 1981 manual and the on-line document (above) for more information on additions and modifications to the new version. Prospective users are reminded of the forthcoming CSO short course on BMDP scheduled for November 15 and 17, 7:00-9:00 PM, in room 166 Commerce West. You may register for the course by calling 333-8640 during office hours.

Questions and comments concerning BMDP should be addressed to Ron Szoke, 131-D DCL, (333-8630) or TELL,SZOKE from a Cyber terminal.

MATHEMATICAL SERVICES

THE LINDO LINEAR PROGRAMMING PACKAGE

We have installed on the CYBER an interactive linear programming package called LINDO. Originally developed at the University of Chicago under the principal direction of Linus Schrage of the Graduate School of Business, LINDO allows you to interactively specify a linear program in a simple algebraic form, such as

```
MIN 2X+3Y
ST
X + Y > 9
END
```

Once a problem is entered into LINDO, various commands can be used to examine and modify it before you request LINDO to solve the problem. LINDO also includes a branch-and-bound facility for solving problems with 0/1 integer variables, a simple parametrics ability which allows you to vary a right hand side through a range and observe the points where the problem basis changes, and a facility for solving simple quadratic programs.

LINDO is accessed by entering the command:

```
GRAB,LINDO.
```

This places certain files in your local file space. Then, to run LINDO, you simply type LINDO followed by a carriage return; the program is loaded by the system and prompts you for commands. It includes an interactive help command which allows you to obtain information about LINDO while running it.

LINDO is documented in Reference Guide RF-4.30 LINDO, and in a short on-line writeup which can be obtained and printed as follows:

```
WRITEUP,LINDO.
PRINT,LINDDOC/AS/CC/EJ
```

Due to copyright restrictions, the LINDDOC writeup does not contain a full description of the LINDO commands; rather, it is confined largely to describing the use of LINDO on our CYBER systems. LINDO is described in detail in the following two books:

- *User's Manual for LINDO.* Linus Schrage 1981; Scientific Press, Palo Alto, CA; 67 pages; $11.25.

- *Linear Programming Models with LINDO.* Linus Schrage 1981; Scientific Press, Palo Alto, CA; 242 pages; $20.00

These books are not currently available locally, but can be ordered from the Scientific Press, 670 Gilman Street, Palo Alto, CA 94301.

LINDO questions or problems should be taken to Stan Kerr, 179 DCL (333-4715).

AN IMSL WRITEUP

With IMSL's permission, we have compiled an on-line writeup for the IMSL library which combines all the general usage information from the IMSL manual, plus the chapter summary information from each chapter of the manual. The chapter summaries list all the routines in a given chapter and set forth general background on the use of the routines in that chapter. The writeup can be obtained and printed by entering the following CYBER commands:

```
WRITEUP,IMSL.
PRINT,IMSLDOC/AS/CC/EJ/FORMS=PPS
```

(Since the writeup is about 90 pages long, FORMS=PPS is highly recommended!)

We are interested in knowing whether this abbreviated form of the IMSL manual is useful to people. The only other on-line documentation we have for IMSL consists of the subroutine writeups available through the MATH procedure. Often, it is still necessary to copy information from the full manual, whose price ($38.00) and bulk (4 volumes) make it inconvenient for many users to obtain. If you have comments on this new writeup, please send them to Stan Kerr, 179 DCL (333-4715) or TELL,UN=MATHLIB on the CYBERs.

THE SIMSCRIPT SIMULATION LANGUAGE

CSO has installed the SIMSCRIPT II.5 simulation package from CACI, Inc. on the CYBER systems. This language includes the facilities of a general-purpose programming language like FORTRAN, with special facilities for the simulation of discrete event systems. It is accessed by the command:

```
GRAB,SIMII5.
```

This places a compiler and a run-time library in your local file space. The compiler is used in much the same fashion as the FTN compiler, e.g., a program in local file PROG can be compiled by entering the command:

```
SIMII5,I=PROG,L=LIST.
```

or

```
SIMII5,I=PROG,L=0.          (where 0 is zero in L=0)
```

Each of these forms compiles the SIMSCRIPT program in local file PROG and places a "compiled binary" in local file LGO; in the first form, a compiler listing goes to local file LIST, while in the second the listing is omitted, except for error messages, which are displayed on local file OUTPUT (the terminal).

Once the binary file LGO is produced, it can be run by entering the control statement:

```
LGO,xxx
```

where xxx represents various options which can be used at "run time." These options allow for specifying SIMSCRIPT data files and output files, dynamic storage, and other options. For instance, if the standard SIMSCRIPT READ and PRINT statements are used in the program to read data and do output, then

```
LGO,SIMU5=data,SIMU6=out.
```

is used to indicate the data file and the output file. If the file LGO is saved, it can be run at a later time, provided you issue the GRAB,SIMII5 again so that the "run time" library is present.

A short writeup explaining SIMSCRIPT control statement options can be obtained and printed by entering the CYBER commands:

```
WRITEUP,SIMII5.
PRINT,SIMSDOC/AS/CC/EJ
```

Complete manuals for SIMSCRIPT are unfortunately too expensive to keep in stock at the Distribution Office, but can be ordered from the vendor at the following address:

CACI
12011 San Vicente Boulevard
Los Angeles, CA 90049

The *SIMSCRIPT II.5 Reference Handbook,* which describes the language (but not its CDC implementation) is $30.00; the *User Information Manual* for the CDC version of SIMSCRIPT II.5 is $25.00 and describes the particular details of operating SIMSCRIPT on a CDC system. CSO will keep copies of these manuals for public inspection at the Systems Consulting Office, 1208 W. Springfield. We have made arrangements to also keep a copy of each manual on reserve in the Computer Science Library.

Questions and comments should be directed to Stan Kerr, 179 DCL (333-4715) or TELL,UN=MATHLIB.

TRANSACTIONS ON MATHEMATICAL SOFTWARE

We now have on tape all algorithms published in *Transactions on Mathematical Software* up to June 1982 (from algorithm 493, published in the first issue of March 1975, up to algorithm 584). Algorithms from the September 1982 issue have been ordered, but have not arrived.

All TOMS algorithms are kept on a public tape; file TOMS/UN=MATHLIB contains a directory of this tape, plus directions for reading particular files on it. To print a copy of this directory, enter the following CYBER commands:

```
GET,TOMS/UN=MATHLIB.
PRINT,TOMS.
```

The "Math Note" series described in the March 1981 issue of *OFF-LINE* includes a note summarizing all algorithms published in TOMS to date. To print this note (number 6), enter:

```
GET,MNOTES/UN=MATHLIB.
MNOTES,6.
PRINT,OUT/AS/CC/EJ.
```

(To print instructions on the use of MNOTES, substitute a 0 (zero) for 6 in the example above.)

The algorithms published in the March, June, and September 1982 issues are as follows (the September algorithms have not arrived yet):

In the March 1982 issue:

581: An Improved Algorithm for Computing the Singular Value Decomposition, by Tony F. Chan

In the June 1982 issue:

582: The Gibbs-Poole-Stockmeyer and Gibbs-King Algorithms for Reordering Sparse Matrices, by John G. Lewis
583: LLSOR: Sparse Linear Equations and Least Squares Problems, by Christopher C. Paige and Michael A. Saunders
584: CUBTRI: Automatic Cubature over a Triangle, by D. P. Laurie

Replacements for algorithms 506, 508, and 509 were also published in this issue.

In the September 1982 issue:

585: A Subroutine for the General Interpolation and Extrapolation Problems, by C. Brezinski
586: ITPACK 2C: A FORTRAN Package for Solving Large Sparse Linear Systems by Adaptive Accelerated Iterative Methods, by David R. Kincaid, John R. Respess, David M. Young, and Roger G. Grimes (We currently have ITPACK 2A on the CYBERs.)
587: Two Algorithms for the Linearly Constrained Least Squares Problem, by Richard J. Hanson and Karen H. Haskell

The following articles which appeared in these issues may be of interest to some of our users:

In the March 1982 issue:

An Evaluation of Mathematical Software that Solves Systems of Nonlinear Equations, by K. L. Hiebert

LSOR: An Algorithm for Sparse Linear Equations and Sparse Least Squares, by Christopher C. Paige and Michael A. Saunders

An Improved Algorithm for Computing the Singular Value Decomposition, by Tony F. Chan. (This describes Algorithm 580.)

In the June 1982 issue:

Exploratory Data Analysis in a Study of the Performance of Nonlinear Optimization Routines, by David C. Hoaglin, Virginia C. Klema, and Stephen C. Peters

Implementation of the Gibbs-Poole-Stockmeyer and Gibbs-King Algorithms, by John G. Lewis

In the September 1982 issue:

UIMP: User Interface for Mathematical Programming, by E. F. D. Ellison and Gautam Mitra

A New Implementation of Sparse Gaussian Elimination, by Robert Schreiber

If you are looking for software, *Transactions on Mathematical Software* is one of the journals to check, along with *Numerische Mathematik, BIT, Computer Journal,* and various publications of ACM (the Association for Computing Machinery) and SIAM (the Society for Industrial and Applied Mathematics).

In the Systems Consulting Office at 1208 W. Springfield, we have bound volumes containing all the algorithms published in Communications of the ACM from 1960 to 1975 (when the algorithms department of CACM became Transactions on Mathematical Software). The loose-leaf service of ACM which sends us updates of the algorithms includes an index going back to 1960. This index includes many algorithms from journals other than CACM.

FEATURE ARTICLES

ARTICLES REQUESTED FROM USERS

Several months ago we added a section entitled FEATURE ARTICLES to *OFF-LINE.* We have been running a series of articles about the workings of X3J3, the FORTRAN Standards Committee. At this time we would like to expand this section to include articles submitted by our users. We feel that many people on campus are doing work with computers that other users would enjoy reading about.

Articles should be of interest to computer users and should be related to the computer systems used at the University. However, articles on new developments in computing, or articles from other newsletters or sources, are welcome. Articles reprinted from other sources are published with an acknowledgement of the original source or permission of the author if the article is copyrighted.

All articles submitted should be typewritten and include copy-ready graphs, plots, or diagrams if these are needed to supplement the article. CSO reserves the right to decide which articles will be published.

If you have an article you would like to have considered for publication in *OFF-LINE,* please submit it to:

Editor, OFF-LINE
120 DCL
Computing Services Office

FORTRAN FUTURES

This article is part of a series on the work of the FORTRAN Standards Committee X3J3 in producing the next revision of the FORTRAN standard. The reader is reminded that the features described in this article are not a part of any FORTRAN compiler currently available, but rather are proposed requirements for FORTRAN processors produced in the late 1980's through the mid 1990's. Although every effort has been made to accurately describe the current position of X3J3 on these matters, the development of a revision to the FORTRAN standard is an evolutionary process, and these proposals may be subject to refinement, revision, or even retraction. Comments on these proposals may be given to Kurt Hirchert of the CSO Systems Consulting staff, who is the author of this article and a member of X3J3.

The effects of work on a language standard are normally felt only in the long term. For example, the features described in this series of articles are targeted for inclusion in a revision of the FORTRAN standard to be adopted some time in the late 1980's, with supporting processors not likely to be generally available until the 1990's. The subject of this month's article is features whose effects are especially long term, for they will not be felt until the implementation of the standard revision following the next one, well past the turn of the century. The reason for the delay in these features taking effect is that they are deletions from the FORTRAN language. Under current rules, features cannot be deleted from a language standard unless there is a warning of this possibility in the previous revision of the standard.

Since these rules were not in effect when the current standard was adopted, it contains no warnings, and no existing features can be omitted from the next revision of the FORTRAN standard. Thus, all that will be included in the coming revision is the warnings for deletions to take place in the following revision.

One area where there will be significant changes is control structures. Traditionally, FORTRAN has relied heavily on statement labels and the GO TO statement. In more recent years, the advantages of "structured" facilities has become more apparent and they have been introduced into the language. Although the basic GO TO statement will be retained for those few cases where "structured" facilities do not work well, the more esoteric forms of GO TO will be dropped. Thus, the computed GO TO will be effectively replaced by a CASE construct and the assigned GO TO and the ASSIGN statement will be effectively replaced by true internal procedures. Similarly, the arithmetic IF is easily replaced by logical or block IFs. On a slightly different note, the new block DO construct will replace the traditional DO loop, since it is more general and is syntactically more like the other "structured" control constructs.

Another major area of change relates to the concept of storage association. Although this concept has allowed FORTRAN programmers to express many useful things, it has also permitted many nonportable or subtly dangerous expressions. This concept is being removed completely from the language. Affected features include COMMON, EQUIVALENCE, BLOCK DATA, and the ability to pass portions of an array by passing specific array elements. The replacement for storage association is not a single concept, but a series of concepts which are more portable and safer to use. These include the SHARED DATA subprogram, data structuring, the specification of aliases for all or part of an entity, the explicit passing of subsections of an array, and the use of the IDENTIFY statement to "remap" arrays in different dimensionalities.

There are a number of features in the area of procedures which are destined for removal. The data and code sharing offered by the ENTRY statement are more limited and more dangerous than those offered by new features, so the ENTRY statement will eventually be deleted. The alternate RETURN facility is little more than an extra integer argument and a computed GO TO, so it too has been marked for elimination. Internal functions have made statement functions obsolete. Generic procedures like ABS have made specific procedures for specific types like IABS, DABS, and CABS unnecessary.

There are a number of miscellaneous deletions. The new real precision control makes a separate DOUBLE PRECISION type redundant. The baroque syntax of the DATA statement will be replaced by a more regular initialization facility. Since array dimensioning can be done on type specificaton statements and all of the type names are shorter than the word DIMENSION, the DIMENSION statement is also obsolete. Although there is no replacement for the PAUSE statement, it no longer serves a useful function in most FORTRAN and is thus also destined for the scrapheap.

Finally, with the reduced emphasis on statement labels and GO TOs, the traditional fixed column orientation of the FORTRAN source form is less appropriate than ever and a new, more flexible source form will replace it in the long run.

DOCUMENTATION

NEW AND REVISED CSO DOCUMENTATION

Revised Manual

A Tutorial Guide to the ICE Text Editor - May 1982

This manual has been updated; descriptions of the M and MOD commands, and an index have been added. In anticipation of the expected change to IAF this fall, the first chapter of this manual was updated to reflect the differences between IAF and TELEX. However, the change to IAF was delayed, so users purchasing this new version should temporarily ignore the IAF sections.

On-line Documentation

On-line documentation has been added for the following:

LINDO - an interactive linear programming package. This writeup may be obtained and printed by entering:

```
WRITEUP,LINDO.
PRINT,LINDDOC/AS/CC/EJ.
```

BMDP - 1981 release of biomedical statistical programs package. This writeup may be obtained and printed by entering:

```
WRITEUP,BMDP.
PRINT,BM81DOC/AS/CC/EJ.
```

Reference Guides

RF-0.12	OPERATING SCHEDULES	New 9/30/82
RF-2.5	LISP	Revised 10/20/82
RF-4.30	LINDO	New 10/15/82
RF-4.31	SIMSCRIPT	New 10/25/82
RF-7.2	ARCHIVE	Revised 10/8/82
RF-7.31	PLOT or PLOTZ	Revised 8/30/82
RF-11.1	ID Card	Revised 10/15/82
RF-11.2	4341 Rate/Charge Estimator	Revised 10/8/82
RF-15.2	IBM ZETA PLOT Parameters	Revised 8/30/82

SALES - EXCHANGES - HELP WANTED

USED TI PORTABLE FOR SALE

One used TI portable terminal, Model 725, for sale. Asking $350.00. If interested, please contact: Andrew Ortony, Center for the Study of Reading, 51 Gerty Drive (333-2853).

COMPUTER PROGRAMMER

Illinois Natural History Survey and U of I. Duties will include data storage, retrieval, and analysis of information related to biological systems, and obtaining data from the National Weather Service via telecommunications. Applicants should have a B.S. or M.S. degree, experience with computer systems available at the U of I, and examples of their programming. Salary based on qualification and experience. In order to ensure full consideration, applications must be received by November 12, 1982.

To apply: Submit resume or education and experience, transcripts and 4 letters of reference to: Dr. W. G. Ruesink, Illinois Natural History Survey, 607 E. Peabody Drive, Champaign, IL 61820. The Illinois Natural History Survey and the University of Illinois are Affirmative Action/Equal Opportunity Employers.

OFF-LINE's Mailing List

If you wish to be placed on our mailing list for future issues of *OFF-LINE*, if you wish to be removed from the list, or if you wish to enter an address correction, please complete and return this page. **(Current subscribers are kept on the mailing list until a specific request for removal is received, or until a mailing is returned as undeliverable.)**

Check one:

- ☐ Place my name on mailing list
- ☐ Make the following corrections or changes
- ☐ Delete my name from mailing list

_____

First name -- Initial -- Last Name

Campus Address:

_____

Room - Building

_____

Department

Off-campus Address:

_____

Organization or Company (if applicable)

_____

Street Address

_____

City -- State -- Zip Code

If address correction, give old address and zip code below.

_____

_____

_____

SEND TO:

OFF-LINE
150 Digital Computer Laboratory
University of Illinois at Urbana-Champaign
1304 West Springfield Avenue
Urbana, Illinois 61801

Computing
Services
Office

Off-Line

University of Illinois at Urbana-Champaign VOL. 10, NO. 12 December 1982

EDITOR: Lynn Bilger
PHONE: (217) 333-6236
120 Digital Computer Lab
1304 W. Springfield Ave.
Urbana, Illinois 61801

CSO DIRECTORY - STAFF AND SERVICES

Administrative

Director	George Badger	150	DCL	333-4103
Business Manager	Stanley Rankin	150	DCL	333-6530
Secretary	Joyce McCabe	150	DCL	333-1637

User Services and Hardware/Software Support

User Accounting		**1208**	**W Springfield**	**333-7752**
Distribution Office		**1208**	**W Springfield**	**333-6760**
Systems Consulting		**1208**	**W Springfield**	**333-6133**
Statistical Services Consulting		**85**	**Comm West**	**333-2170**
Text Processing Consulting		**118**	**DCL**	**333-7318**
Terminal Repair Service		**150**	**DCL**	**333-0969**
CYBER Dial-up Numbers	**CYBER 175**	**110-300**	**baud**	**333-4000**
	CYBER 175	**1200**	**baud**	**333-4001**
	CYBER 174	**110-300**	**baud**	**333-4004**
Asst Dir User Services	Robert Penka	173	DCL	333-4709
Asst Dir Systems and Operations	Sandra Moy	177	DCL	333-4703
Asst Dir Development	J. M. Randal	175	DCL	333-9772
Asst Dir Engineering and Hardware Selection	Cliff Carter	195	DCL	333-3723
Manager Accounting Services	Gary Bouck	1208	W Springfield	333-7752
Manager Statistical Services	Beth Richardson	189	DCL	333-6276
Documentation	Lynn Bilger	120	DCL	333-6236
CYBER-IBM Operations	Jack Knott	194a	DCL	333-6562
UNIX Operations	Debbie Hudson	123	DCL	333-8150
Laboratory Support Project	Mike Gardner	18	DCL	333-7904
RJE Operations North	Rex Duzan	162	DCL	333-6285
RJE Operations South	Don McCabe	1208	W Springfield	333-2171 333-7752

RJE Sites (see Reference Guide RF-0.3 for operating hours)

Agriculture	M103	Turner Hall	333-8170
Chemistry	153	Noyes Lab	333-1728
Commerce West	70	Comm West	333-4500
CRH Snack Bar	120	Snack Bar	333-1851
DCL Routing Room	14	DCL	333-6203
Electrical Engineering	146	EEB	333-4936
Florida Ave Res Hall		FAR	333-2695
Illinois St Res Hall		ISR	333-0307
Mechanical Engineering	65	MEB	333-1430
Psychology	453	Psych Bldg.	333-7531
Social Science	202	Lincoln Hall	333-0309

OFF-LINE is the monthly newsletter of the Computing Services Office at the University of Illinois at Urbana-Champaign. Unless otherwise indicated, permission to reprint is freely granted, provided that the author, if named, and the Computing Services Office (CSO) are credited. Information in this issue is current as of November 25, 1982.

CSO operates a CDC CYBER 175 with 262K words of central memory and a CDC CYBER 174 with 196K words of central memory. The 175 and 174 run under the NOS Operating System and share 512K words of ECS. The 175 serves over 200 simultaneously active text and graphics terminals and the 174 serves over 100 simultaneously active terminals. CSO also operates an IBM 4341 with 4 million bytes of memory running HASP-OS/MVT under VM. In addition, CSO operates a DEC VAX 11/780 with 4 million bytes of memory, running under the UNIX Operating System and driving a GSI CAT-8 phototypesetter.

Operating Hours (see HEARYE,SCHEDUL for exceptions):

	CYBERS 174/175	IBM
M-F	8 am to 6 am	8 am to 6 am
SAT	8 AM - Midnight	8 AM - 6 AM
SUN	Noon - 6 AM	Noon - 6 AM

CYBER

TEMPORARY TAPE STORAGE

CSO provides racks for the temporary storage of tapes which will be used for only a short time. These tapes are stored on a special rack, called the TEMP rack, where they may be kept up to a limit of three weeks. We ask that you come to 123 DCL to retrieve any tape you have placed in the TEMP rack before the three-week limit is reached.

Once a tape has exceeded the three-week storage limit, it is removed from the computer room, and the person(s) or department(s) responsible for the tape are asked to retrieve it. Once a tape has been removed from the TEMP rack, it is no longer available for use.

We appreciate your cooperation in this matter.

PRESERVING FILES AT END OF SEMESTER

Invariably, at the end of each semester, a few users encounter difficulties related to file sharing and file preservation. The problems are aggravated by the facts that many accounts expire at this time of the year and that many computer users are absent from campus over the holidays. This article lists a few facts you should bear in mind and suggests actions you can take to prevent inconvenience to yourself or your co-workers.

When you create a file, it is assigned to the **charge,project** you are using at the time. Assignment to a project means that the project is made financially responsible for all disk space charges incurred by the file. When, for any reason, you can no longer use a project, all of your files which are assigned to that particular project are made inaccessible. An attempt to access one of them will be met with the error message "FILE MUST BE CLAIMED". You must reassign (i.e., CLAIM) such a file to another valid project before it can be used. Any such file, which has not been reassigned to another project after 120 days, will be purged from the system.

If one of your projects will expire soon, you or your co-workers could be affected by the system behavior described above. The most obvious effect could be the loss of your files, assuming you do not reassign them to another project within 120 days. Less obvious, access to any of your files which are being shared with co-workers will be blocked between the time your project expires and the reassignment of the files to another valid project. This can be especially annoying if you leave town for several weeks since only you will be able to reassign the file to another project, leaving your co-workers stranded.

You can prevent both problems by reassigning files to a project to which you will have continuing access before the owning project lapses. The DIRECT command provides an easy way to accomplish this. First, enter a BILL command naming on it a **charge,project** to which you will have continuing access.

Second, enter the command:

```
DIRECT/PROJECT
```

The generated output will consist of a list of file names, each followed by the **project index hash**, four characters which uniquely identify the owning project. The pairing between project index hashes and projects will be given at the end of the list of file names. Identify the project index hash which identifies the project which will soon lapse. Then enter the command:

```
DIRECT/CLAIM/PROJECT=hash
```

where *hash* is the project index hash which you have identified above. This command will reassign all files currently assigned to the project identified by *hash* to the project named on the last BILL command.

We expect to see another problem during semester break. Invariably, someone will ask us to provide access to files which belong to a colleague or student employee who is absent during the break. In these cases we can offer no real help. The files do not belong to us and we cannot provide anyone access to them.

If you are working with someone who will leave campus during the break, anticipate any file sharing needs which will be generated by his absence. Take into account that you may well change your own work patterns and may need access to additional files owned by your absent colleague. Before your colleague leaves town, make sure that you have been given access to all necessary files. You would be also well advised to ask your colleague to verify that the files are assigned to a project which will continue to be usable during his absence. If this is not the case, access to his files will be blocked, as noted above.

MATHEMATICAL SERVICES

SLAM II VERSION 1.0 RELEASE 5.0

We have received, and installed in FUTURE, the latest release of the SLAM II Simulation package from Pritsker and Associates. This version can be accessed by entering the following CYBER command:

```
GRAB,SLAM/F.
```

It will remain in FUTURE until the end of the current semester, at which time it will become the default version of SLAM II. Some of the changes and upgrades in this release are as follows.

Binary Search Algorithm

The binary search algorithm, which increases the efficiency of inserting entries into the event files, has now been automated. It is no longer necessary to request the algorithm on the LIMIT and PRIORITY statements. The algorithm is not available for files other than the event file, since generalizing the procedure would significantly reduce its efficiency. The "standard" SLAM filing mechanism begins at the end of the file and searches forward for the location at which a new entry should be inserted. This is efficient for models in which new events tend to be scheduled near the end of the current calendar and, for such models, the new binary search algorithm will not result in much improvement in running time. For large event files, requiring a significant amount of searching during insertion, the algorithm can result in dramatically increased efficiency. A decrease of 50% in execution time is not uncommon.

Distribution Functions

Modifications have been made to function GAMA and function NPSSN (see *Introduction to Simulation and SLAM,* Appendix A), which result in greater accuracy in approximating beta, gamma, and Poisson distributions.

Preemption

When using the PREEMPT node (see *Introduction to Simulation and SLAM,* page 122), the modeler is cautioned that an entity that is currently in a QUEUE or AWAIT mode or in a service activity cannot be preempted. Any attempt to preempt in such a situation would cause an execution error and an exit from the simulation. In Release 5.0, the execution error has been eliminated. If a preemption cannot occur, the entity attempting to preempt will wait in the file associated with the PREEMPT node until the required resource is released and its associated files are polled.

DETECT Node

When detecting a continuous variable crossing a discrete variable (*SLAM II Manual,* page 24, Release 4.0), the input processor could overwrite state event information if the DETECT statement included the M field. This error has been corrected.

SELECT Node

The RANDOM rule for selecting among the non-identical service activities (*Introduction to Simulation and SLAM,* page 116), was not properly processed. The processing has now been corrected.

Error Messages

Additional input error messages have been added to prevent the definition of zero-capacity queues or activities emanating from TERMINATE nodes. These have always been error conditions in SLAM II, but were not flagged in previous releases. Additional execution error messages have been added to several random sampling functions. These will generate a SLAM error message in the event that improper parameters would result in an operating system message due to division by zero or a zero argument to the logarithm function.

FEATURE ARTICLES

PROJECTION OF A TERMINAL DISPLAY

Instructors, faculty members, how many times have you wished, while discussing the use of computer applications or methods with a large group or class, that you could simply "show" them what you mean by using a terminal and the associated terminal screen display? Impossible you say? The terminal screen is too small and they cannot possibly see it! Well, the Instructional Media Division/OIR has solved your problem. A terminal display can now be projected for a large group through the use of a computer-video projector. The projector, an EDP-57, is portable, uses standard projection screens, and can display output from either a terminal or a microcomputer.

If, in addition to the projector, you need the terminal, a Lear Siegler terminal with modem is also available. This terminal has been modified for composite video and is equipped with an RG-512 card for graphics capability. This equipment is currently being used with the CYBER and several micro-computers.

The projector and terminal are available at no charge to faculty using them for classroom instruction, but may be rented for other purposes. For more information, contact:

Warren Johnson
Instructional Media Division/OIR
Telephone: 333-3690

NOTE: If you plan to use this equipment via a dial-up line to the computer, you should take into account the possibility that all lines will be busy and should make alternate plans. Busy signals are most likely to occur between 9 AM and 4 PM.

FORTRAN FUTURES

This article is part of a series on the work of the FORTRAN Standards Committee X3J3 in producing the next revision of the FORTRAN standard. The reader is reminded that the features described in this article are not a part of any FORTRAN compiler currently available, but rather are proposed requirements for FORTRAN processors produced in the late 1980's through the mid 1990's. Although every effort has been made to accurately describe the current position of X3J3 on these matters, the development of a revision to the FORTRAN standard is an evolutionary process, and these proposals may be subject to refinement, revision, or even retraction. Comments on these proposals may be given to Kurt Hirchert of the CSO Systems Consulting staff, who is a member of X3J3.

The 84th meeting of X3J3 was held November 8-12, 1982 in Tulsa, Oklahoma. Activity at that meeting included the following:

- Representatives of IFIP working group 2.5 on numerical software met with both the full committee and its subcommittees to discuss areas of concern. Primary emphasis was placed on the array handling facilities being added to the language and the facilities needed to conveniently support data abstraction in Fortran, especially user-defined operators on structured data.

- The task group on multi-tasking facilities worked with several of the other subcommittees on refining its proposal for error/event/exception handling in Fortran. The current state of this work was presented to the full committee.

- The subcommittee on array processing presented its current thinking on the construction of array constants.

- The subcommittees involved with data sharing and dependent compilation models forged a compromise proposal unifying facilities in these areas. An initial presentation of this proposal was made to the full committee.

- A presentation was made on the results of research in generally applicable techniques which can be used to significantly improve the accuracy of several operations performed by computer arithmetic. Exploration was begun on the possibility of more closely specifying requirements for arithmetic accuracy in the Fortran standard.

- The full committee's guidance was solicited in the refinement of a proposal for specifying compile time aliases for Fortran variables, array elements, substrings, etc.

- A proposal was presented for providing access to the number of items transferred in an input/output statement.

- Several interpretations of the 1978 Fortran standard were tentatively approved.

- Extensive liaison reports were presented on the work of several bodies doing related work.

- A proposed standard vocabulary for describing computer languages was reviewed and found to be inadequate for the purposes of describing the next Fortran revision.

- Due to a limited premeeting distribution and committee operating procedures, no proposals were formally adopted at this meeting, but the informal votes taken suggest that most, if not all, of the proposals presented informally at this meeting may be formally adopted at the next meeting in February.

MISCELLANEOUS

NO JANUARY ISSUE OF OFF-LINE

Due to the long Christmas-semester break, there will be no issue of *OFF-LINE* published in January. The January and February issues will be combined and should be available around the first of February.

1983 NATIONAL EDUCATIONAL COMPUTING CONFERENCE

The 1983 National Educational Computing Conference (NECC 83), hosted by Towson State University, will be held in Baltimore, Maryland, June 6-8, 1983.

NECC 83 is a cooperative effort of 19 educational and scientific organizations. Original papers are solicited from all academic disciplines and research areas in educational computing. Actual experiences with computer use in the classroom at all levels of education are encouraged. Of particular interest are papers prepared by students at the secondary, undergraduate and graduate levels.

Authors should submit an original manuscript (double-spaced, maximum of 15 pages) by December 15 to A. J. Turner, Department of Computer Science, College of Nursing Building, Clemson University, Clemson, South Carolina, 29631.

Special sessions are also being organized with invited speakers on the current status of educational use of computers. Ideas for sessions and participants should be submitted by December 1, 1982 to Jean Rogers, Department of Computer and Information Science, University of Oregon, Eugene, Oregon, 97403.

In addition, a series of 15-20 minute informal project presentations will be organized on such topics as curriculum, course, and software development. Persons interested in presenting a project report should submit a one-page summary by December 15, 1982 to William Ryan, Computing Center, Swarthmore College, Swarthmore, Pennsylvania, 19081.

For conference information and registration forms, write Doris K. Lidtke, General Chairman, NECC 83, Department of Mathematics and Computer Science, Towson State College, Baltimore, Maryland 21204.

SALES - EXCHANGES - HELP WANTED

PASCAL PROGRAMS WANTED

I am conducting a research study on the behavior of PASCAL programs, and would greatly appreciate receiving programs of any length and application (for example, systems, non-numeric, numeric, applications, scientific and engineering programs).

These programs will be used for the purpose of computing research only, and the results of this study will be available for the asking. Please contact:

Professor Nazim Madhavji
199 Digital Computer Lab
Department of Computer Science
University of Illinois
Telephone: 333-0215

TWO PART-TIME COMPUTER PROGRAMMERS NEEDED

The Office of Statewide Faculty Research Resources on the Urbana Campus has two (2) part-time programmer positions available to assist with the operation and continued development of a statewide, on-line data base. The two positions are graduate assistantships with salary commensurate with experience.

SYSTEM ANALYST - ¼ time assistantship for Spring 1983 and expanding to ½ time (20 hours/week) in Summer 1983 and 1983-84 academic year. The graduate student will make refinements to recently developed data base application software. A major portion of the effort will be devoted to writing documentation. Prerequisites shall include *one or more* of the following:

- programming experience
- familiarity with data base design
- familiarity with the use of disk files
- technical writing experience

As well as (1) the ability to learn rapidly and (2) a strong command of the English language. **Starting date - January 1983. Apply immediately.**

USER FRIENDLY INTERFACE - ½ time assistantship for Summer 1983 and 1983-84 academic year. Graduate student will make additional refinements to recently developed user friendly interface for a data base available to statewide audiences such as industry, government and higher education. Applicant must have some experience or strong interest in user friendly interfaces while possessing both creativity and initiative to enhance existing user

friendly interface. Recommended prerequisites include: (1) programming experience and (2) a strong command of the English language. **Starting date - May/June 1983. Apply immediately.**

Applicants Note: These assistantships offer students a challenging opportunity for hands-on development and major responsibility for an innovative statewide faculty resource data base. Preference will be given to applicants who can work with other people and communicate effectively.

For further information, contact Richard Zollinger or James White at 333-8770, Office of Statewide Faculty Research Resources, 1205 w. Oregon, Urbana. Applications available by contacting 333-8770.

OFF-LINE's Mailing List

If you wish to be placed on our mailing list for future issues of *OFF-LINE*, if you wish to be removed from the list, or if you wish to enter an address correction, please complete and return this page. **(Current subscribers are kept on the mailing list until a specific request for removal is received, or until a mailing is returned as undeliverable.)**

Check one:

- ☐ Place my name on mailing list
- ☐ Make the following corrections or changes
- ☐ Delete my name from mailing list

_____

First name -- Initial -- Last Name

Campus Address:

_____

Room - Building

_____

Department

Off-campus Address:

_____

Organization or Company (if applicable)

_____

Street Address

_____

City -- State -- Zip Code

If address correction, give old address and zip code below.

_____

_____

_____

SEND TO:

OFF-LINE
150 Digital Computer Laboratory
University of Illinois at Urbana-Champaign
1304 West Springfield Avenue
Urbana, Illinois 61801

Computing
Services
Office

Off-Line

University of Illinois at Urbana-Champaign VOL. 11, NO. 3 March 1983

EDITOR: Lynn Bilger
PHONE: (217) 333-6236
120 Digital Computer Lab
1304 W. Springfield Ave.
Urbana, Illinois 61801

Contents

CSO DIRECTORY - STAFF AND SERVICES

Administrative

Director	George Badger	150	DCL	333-4103
Business Manager	Stanley Rankin	150	DCL	333-6530
Secretary	Joyce McCabe	150	DCL	333-1637

User Services and Hardware/Software Support

User Accounting		**1208**	**W Springfield**	**333-7752**
Distribution Office		**1208**	**W Springfield**	**333-6760**
Systems Consulting		**1208**	**W Springfield**	**333-6133**
Statistical Services Consulting		**85**	**Comm West**	**333-2170**
Text Processing Consulting		**118**	**DCL**	**333-7318**
Terminal Repair Service		**150**	**DCL**	**333-0969**
CYBER Dial-up Numbers	**CYBER 175**	**110-300**	**baud**	**333-4000**
	CYBER 175	**1200**	**baud**	**333-4001**
	CYBER 174	**110-300**	**baud**	**333-4004**
Asst Dir User Services	Robert Penka	173	DCL	333-4709
Asst Dir Systems and Operations	Sandra Moy	177	DCL	333-4703
Asst Dir Development	J. M. Randal	175	DCL	333-9772
Asst Dir Engineering and Hardware Selection	Cliff Carter	195	DCL	333-3723
Manager Accounting Services	Gary Bouck	1208	W Springfield	333-7752
Manager Statistical Services	Beth Richardson	189	DCL	333-6276
Documentation	Lynn Bilger	120	DCL	333-6236
CYBER-IBM Operations	Jack Knott	194a	DCL	333-6562
UNIX Operations	Debbie Hudson	123	DCL	333-8150
Laboratory Support Project	Mike Gardner	18	DCL	333-7904
RJE Operations North	Rex Duzan	162	DCL	333-6285
RJE Operations South	Don McCabe	1208	W Springfield	333-2171 333-7752

RJE Sites (see Reference Guide RF-0.3 for operating hours)

Agriculture	M103	Turner Hall	333-8170
Chemistry	153	Noyes Lab	333-1728
Commerce West	70	Comm West	333-4500
CRH Snack Bar	120	Snack Bar	333-1851
DCL Routing Room	14	DCL	333-6203
Electrical Engineering	146	EEB	333-4936
Florida Ave Res Hall		FAR	333-2695
Illinois St Res Hall		ISR	333-0307
Mechanical Engineering	65	MEB	333-1430
Psychology	453	Psych Bldg.	333-7815
Social Science	202	Lincoln Hall	333-0309

OFF-LINE is the monthly newsletter of the Computing Services Office at the University of Illinois at Urbana-Champaign. Unless otherwise indicated, permission to reprint is freely granted, provided that the author, if named, and the Computing Services Office (CSO) are credited. Information in this issue is current as of February 22, 1983.

CSO operates a CDC CYBER 175 with 262K words of central memory and a CDC CYBER 174 with 196K words of central memory. The 175 and 174 run under the NOS Operating System and share 512K words of ECS. The 175 serves over 200 simultaneously active text and graphics terminals and the 174 serves over 100 simultaneously active terminals. CSO also operates an IBM 4341 with 4 million bytes of memory running HASP-OS/MVT under VM. In addition, CSO operates a DEC VAX 11/780 with 4 million bytes of memory, running under the UNIX Operating System and driving a GSI CAT-8 phototypesetter.

Operating Hours (see HEARYE,SCHEDUL for exceptions):

	CYBERS 174/175	IBM
M-F	8 am to 6 am	8 am to 6 am
SAT	8 AM - Midnight	8 AM - 6 AM
SUN	Noon - 6 AM	Noon - 6 AM

CYBER

SHARING THE WEALTH

CSO is trying to encourage users of our facilities to share software whenever appropriate. CSO offers support for such sharing in two ways.

First, CSO is sponsoring a library of user-contributed software accessible through the SHARED command. Stable products of general interest with adequate documentation can be given to CSO for sharing in this way. For more details on eligibility requirements, see the January-February issue of *OFF-LINE* or contact Kurt Hirchert, 129 DCL (or TELL,HIRCHERT).

Second, CSO is encouraging more informal sharing by the maintenance of a software registry. If you have software that is currently running or can run on one of CSO's machines, and that might be of use or interest to others (especially if it has been written up in the literature for your field or if you have invested time and/or money in acquiring it from elsewhere), please let us know about it by dropping a note to Kurt Hirchert, 129 DCL (or TELL,HIRCHERT). We'd appreciate receiving the name of the package (including version numbers and alternate names if appropriate), a brief description of what it does (sufficient for someone not acquainted with the package to judge its usefulness), information on the availability of the package (e.g., the CSO machines on which the package is running (or can be run) or limitations in the license agreement for the package), and the name of someone who can be contacted for more information. Note that the registry is not limited to packages already running on one of CSO's machines. If you've acquired a package from elsewhere, but don't have it running yet, there may still be someone interested in knowing about your work (and maybe even willing to help you with it).

PLOTTING SERVICES

CONTOURING SYSTEM

Precision Visuals, Incorporated's Contouring System has been installed on the CYBERs. The Contouring System is built entirely on top of the DI-3000 graphics package, level A. As such, all the capabilities of DI-3000 are available to the programmer as well as the contouring utility subroutines.

The Contouring System includes three routines for producing contour plots, one which grids randomly-spaced data, and eight routines to set and inquire about attributes and options.

To use the subroutines, you first call JSSINI to initialize the library. Then, after setting any desired options, call one of the three contouring routines:

JSSC1	to draw a quick-look contour map from gridded data
JSSC2	to draw a high-quality contour map from gridded data
JSSC3	to draw a high-quality contour map from randomly located input data

The options and attributes used for drawing the plot may include:

- Drawing of map boundaries
- Annotation of local minimums and maximums
- Annotation of values at grid points
- Annotation of values at random input points
- Drawing of the triangulation used as part of gridding
- Line attribute selection for color, linestyle, etc.
- Character sizes used for labeling contour lines
- Character sizes used for other annotation

To use the Contouring System, one must access both the contour library and the DI-3000 library, selecting a graphics output device, by entering:

```
GRAB,CONTOUR.
GRAB,DI3000(TEK)
```

Then, compile and execute your program as usual. Remember, all DI-3000 products are compiled using the FTN5 compiler; thus, you should use FTN5 for compiling your programs.

Several sample programs are available to demonstrate the use of the Contouring System. For a list of the available sample decks, enter:

```
GRAB,SAMPLES.
SAMPLES,CONTOUR.
```

The catalog of sample decks will be written to the local file DOC. To get a copy of a specific sample program, enter:

```
GRAB,SAMPLES.
SAMPLES,CONTOUR,deckname.
```

The selected sample, *deckname,* will be written to the local file SAMPLE. You may then compile and execute this program.

A copy of the Contouring System manual is available for inspection in the Systems Consulting Office, 1208 W. Springfield. We plan to have copies for sale in the Distribution Office in the near future.

MATHEMATICAL SERVICES

MATLAB -- A MATRIX DESK CALCULATOR

CSO has installed MATLAB, an interactive matrix/vector desk calculator, on the CYBER systems. This package, designed by Cleve Moler of Sandia Laboratories, allows one to interactively create matrices and vectors, and perform common operations on them, such as solving linear systems, eigenvalue analysis, linear least squares, decomposition, and matrix arithmetic. The numerical abilities of the package are based on the LINPACK and EISPACK packages, two state-of-the-art collections of software for solution of linear systems and eigenvalue analysis.

MATLAB is accessed on the CYBER systems by the command

```
GRAB,MATLAB.
```

This places several files in your local file space, one of which is named MATLAB. To run the package, you must enter

```
MATLAB
```

MATLAB is loaded and prompts you for commands. It includes a simple interactive help facility, accessed through the MATLAB command HELP.

There is an on-line manual for MATLAB, which can be printed by the following commands:

```
WRITEUP,MATLAB.
PRINT,MATLDOC/AS/CC/EJ/RJE=xxx.
```

where *xxx* designates some print site on campus (see Reference Guide RF-0.3 Job Entry Sites for a list of site codes).

The version of MATLAB which is up is the single precision version. The package came with a double precision version, which will probably be installed soon.

MATLAB allows matrices and vectors to be entered in several ways:

- **By explicit list of elements**; e.g.,

```
A = < 1 2 3; 4 5 6; 7 8 9 >
```

  defines A to be a 3-by-3 matrix whose first row is 1 2 3, second row is 4 5 6 and third row is 7 8 9. This could also be entered as

```
A = < 1 2 3
      4 5 6
      7 8 9 >
```

  i.e., on 3 separate lines.

- **Use of FOR and WHILE statements.** These provide simple looping facilities, e.g.,

```
for i=1:3, for j=1:3, a(i,j) = 3*(i-1)+1 ;
```

- **Read from an external file.** MATLAB includes commands SAVE (save matrices/vectors on a file) and LOAD (load vectors/matrices from a file) which allow selective or unselective storing of information on CYBER files. The format of these files is quite simple, so they could easily be created from a FORTRAN program.

- **Execute an external FORTRAN program.** The USER command in MATLAB causes it to call a routine named USER which you can write. This routine can easily access matrices and vectors defined in MATLAB. The manual explains the proper setup for USER.

Following is a list of the MATLAB functions which can be applied to matrices and vectors.

INV(A)	- Inverse.
DET(A)	- Determinant.
COND(A)	- Condition number.
RCOND(A)	- A measure of nearness to singularity.
EIG(A)	- Eigenvalues and eigenvectors.
SCHUR(A)	- Schur triangular form.
HESS(A)	- Hessenberg or tridiagonal form.
POLY(A)	- Characteristic polynomial.
SVD(A)	- Singular value decomposition.
PINV(A,eps)	- Pseudoinverse with optional tolerance.
RANK(A,eps)	- Matrix rank with optional tolerance.
LU(A)	- Factors from Gaussian elimination.
CHOL(A)	- Factor from Cholesky factorization.
QR(A)	- Factors from Householder orthogonalization.
RREF(A)	- Reduced row echelon form.
ORTH(A)	- Orthogonal vectors spanning range of A.
EXP(A)	- e to the A.
LOG(A)	- Natural logarithm.

SQRT(A)	- Square root.
SIN(A)	- Trigonometric sine.
COS(A)	- Cosine.
ATAN(A)	- Arctangent.
ROUND(A)	- Round the elements to nearest integers.
ABS(A)	- Absolute value of the elements.
REAL(A)	- Real parts of the elements.
IMAG(A)	- Imaginary parts of the elements.
CONJG(A)	- Complex conjugate.
SUM(A)	- Sum of the elements.
PROD(A)	- Product of the elements.
DIAG(A)	- Extract or create diagonal matrices.
TRIL(A)	- Lower triangular part of A.
TRIU(A)	- Upper triangular part of A.
NORM(A,p)	- Norm with p = 1, 2 or "Infinity".
EYE(m,n)	- Portion of identity matrix.
RAND(m,n)	- Matrix with random elements.
ONES(m,n)	- Matrix of all ones.
MAGIC(n)	- Interesting test matrices.
HILBERT(n)	- Inverse Hilbert matrices.
ROOTS(C)	- Roots of polynomial with coefficients C.
DISPLAY(A,p)	- Print base p representation of A.
KRON(A,B)	- Kronecker tensor product of A and B.
PLOT(X,Y)	- Plot Y as a function of X.
RAT(A)	- Find "simple" rational approximation to A.
USER(A)	- Function defined by external program.

Although MATLAB is directed somewhat more towards the computer scientist than the ordinary user, we hope it will be a useful tool for learning and for handling small problems.

Questions about or problems with MATLAB should be taken to Stan Kerr (179 DCL, 333-4715, or TELL,UN=MATHLIB on the CYBER systems).

RELEASE 4.6 of SIMSCRIPT II.5

Release 4.6 of the SIMSCRIPT II.5 simulation language has been installed in FUTURE on the CYBER systems. It will become the default version (barring problems) in April, and can be accessed until then by entering the command

```
GRAB,SIMII5/F.
```

Features which have been added at release 4.6 include the following.

- A feature which automatically resolves at compilation time, the entry point name conflicts due to the name truncations for compatibility with the 7-character loader naming conventions.

- A program performance monitoring feature which provides the CPU time spent in the execution of each event and/or process.

The SIMSCRIPT language was installed on the CYBER systems last November. Manuals for it can be ordered from the vendor,

CACI
12011 San Vicente Boulevard
Los Angeles CA 90049

Available manuals are the following:

- *The SIMSCRIPT II.5 Reference Handbook* describes the language, but not its usage on CDC systems.

- *SIMSCRIPT II.5 User Information Manual* is published through CDC and describes specifically how to use SIMSCRIPT on CDC systems. The CDC publication number is 84000460.

  NOTE There is an update (Revision E) for this manual for release 4.6. You should order the update soon.

- *Building Simulation Models with SIMSCRIPT II.5* is a workbook designed to teach SIMSCRIPT through examination of a series of case studies. It covers basic simulation concepts and techniques and shows how these are applied in SIMSCRIPT.

THE CAYLEY GROUP THEORY LANGUAGE

In cooperation with the Department of Mathematics, CSO has acquired a language called CAYLEY, which can perform a variety of calculations for finite groups and related algebraic structures. It was developed by John Cannon at the University of Sydney, Australia. The system can, for example, compute with matrices over finite fields, permutation groups and groups given by generators and relations. The algorithms available include for example "find the character table", "find all subgroups of index n", or "test whether a group is simple".

CAYLEY is accessed on the CYBER systems by the command

```
GRAB,CAYLEY
```

which places necessary files in your local file space. Following this, you can run CAYLEY interactively by entering the command

```
CAYLEY
```

The CAYLEY program is then loaded, and prompts you for commands.

For large problems, it may be necessary to allocate more memory for CAYLEY before running it. This can be done by entering

```
RFL,nnnnnn.
```

where *nnnnnn* is the amount of memory to be allocated, e.g., 200000 represents 200000 octal words (i.e., 65536 in decimal). The amount of memory allocated must, of course, be within your account limits and the system limits for the time of day. Your current maximum (taking into account the system limit) will be displayed if you enter the command

```
DISPLAY,MFL.
```

and your account limit can be displayed by entering LIMITS (the memory limit is listed under CENTRAL MEMORY).

There is a copy of the CAYLEY manual in the System Consulting Office at 1208 W. Springfield. Questions about it or problems with it should be taken to Stan Kerr (179 DCL, 333-4715, or TELL,UN=MATHLIB).

THE PROTRAN PROBLEM-SOLVING LANGUAGE

The January-February issue of *OFF-LINE* introduced a new problem-solving language called PROTRAN, from IMSL, Inc. In this article we will show some more examples of PROTRAN programs to demonstrate its usefulness.

There is now an on-line writeup on PROTRAN, which can be obtained and printed by entering the following commands

```
WRITEUP,PROTRAN.
PRINT,PRODOC/AS/CC/EJ/RJE=xxx.
```

where *xxx* designates some print site on campus (see Reference Guide RF-0.3 Job Entry Sites for a list of site codes). Due to copyright restrictions, the writeup does not include a full description of all PROTRAN features; this appears in the *MATH/PROTRAN User's Manual*, published by IMSL, which can be purchased at the CSO Distribution Office for $15 or ordered from IMSL.

In this article, we will show an example of curve fitting using PROTRAN. Suppose we are given the following 5 data points

x	*y*
0.0	3.72011
0.2	3.44630
0.4	3.31467
0.6	3.31412
0.8	3.44796

We want to fit a curve to these points. A simple printer plot can be produced by the

PROTRAN program

```
$   DECLARATIONS
$   VECTOR  X(5),Y(5)
$   ASSIGN  X = (0.0, 0.2, 0.4, 0.6, 0.8)
            Y = (3.72011, 3.44630, 3.31467, 3.31412, 3.44796)
$   PLOT Y; VS X
$   END
```

This gives the graph which appears on the following page.

Suppose we try fitting the data with a quadratic polynomial, then see how close the fit is at the original points. We can request that the polynomial be saved as a function routine so we can easily evaluate it. The PROTRAN program to do this might look like this.

```
$   DECLARATIONS
    VECTOR X(5),Y(5),E(5)
$   ASSIGN  X = (0.0, 0.2, 0.4, 0.6, 0.8)
            Y = (3.72011, 3.44630, 3.31467, 3.31412, 3.44796)
$   APPROXIMATE Y; VS X; USING POLYNOMIALS; NBASIS=3; BY POL
$   ASSIGN  E(I) = Y(I) * POL(X(I))
$   PRINT X,E
$   END
```

The "BY" in the APPROXIMATE statement tells PROTRAN that we want it to save the result of the approximation as a FUNCTION routine named POL. The NBASIS=3 tells it we want the polynomial to have 3 terms, i.e., we want a degree of 2. The ASSIGN statement following it calculates the errors at the data points. The PRINT results in the following output

```
X

          0.00000        .20000        .40000
          .60000         .80000
E

          .00098        -.00238        .00124
          .00073        -.00057
```

Remember that to run the above program, we enter the following control cards

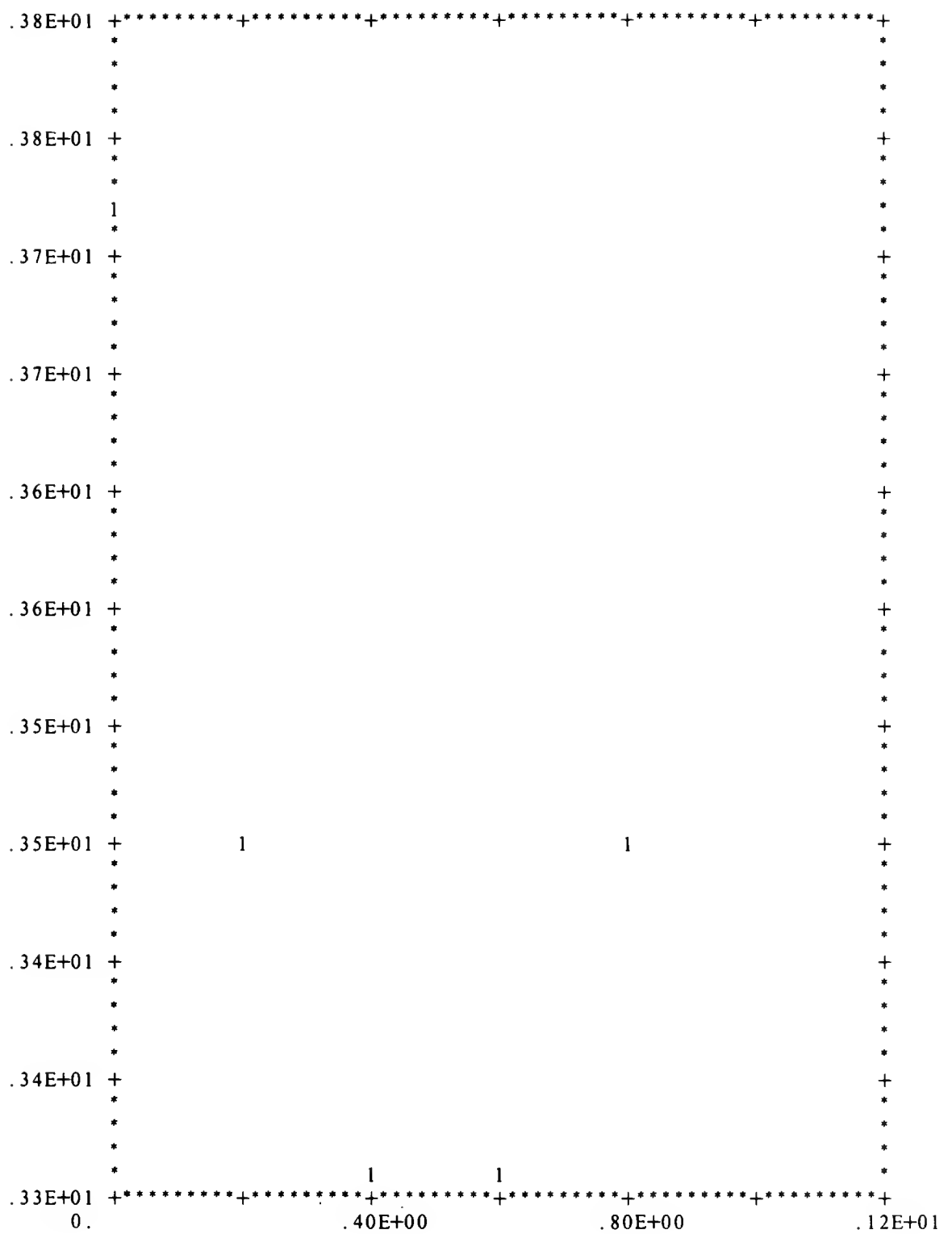
```
GRAB,PROTRAN.
PROTRAN,xxx.
LGO.
```

where *xxx* is the local file containing the PROTRAN program. The PROTRAN,*xxx* statement causes the program to be translated into FORTRAN and compiled (with FTN5), and the LGO causes it to be run; the output can then be found in local file TAPE6.

Now suppose we try a more complicated model of the data: a sum of exponentials. Let's try the nonlinear model

```
Y = C(1)*EXP(C(2)*X) + C(3)*EXP(C(4)*X)
```

PLOT



(The symbol '1' represents each data point.)

Since this is nonlinear, we can not use $APPROXIMATE; we must use $NONLSQ, the procedure for nonlinear least squares. The problem is set up as the minimization of

$$\sum_{I=1}^{I=5} (Y(I)-(C(1) * e^{C(2) * X(I)}+C(3) * e^{C(4) * X(I)}))^2$$

The program might look like this

```
$    DECLARATIONS
     VECTOR X(5), Y(5), F(5), C(4)
$    ASSIGN  X = (0.0, 0.2, 0.4, 0.6, 0.8)
             Y = (3.72011, 3.44630, 3.31467, 3.31412, 3.44796)
     C = (0.,0.,0.,0.)
$    NONLSQ F(C); GUESS = C; SOLUTION=C
     DEFINE
     = = = =
     DO 10 I=1,5
       F(I) = Y(I) - (C(1)*EXP(C(2)*X(I)) + C(3)*EXP(C(4)*X(I)))
10   CONTINUE
     = = = =
$    PRINT X,C,F
$    END
```

The NONLSQ procedure is told that the vector F is to contain the terms the sum of whose squares we are minimizing; the size (5) of F tells NONLSQ how many observations we have. The C in the NONLSQ procedure tells it what vector contains the parameters we seek, and how many there are (4). GUESS is used to tell it what vector (C) contains initial guesses for the parameters (this is always necessary for nonlinear least squares); and SOLUTION tells it in what vector it should store the best parameters it finds. The DEFINE tells PROTRAN that lines of FORTRAN will follow which calculate the "residuals" F(I) for a given set of parameters C; these FORTRAN lines may not include any PROTRAN statements, they must be pure FORTRAN. The "= = = =" tells PROTRAN exactly where the FORTRAN begins and ends. When $NONLSQ is done, the program prints the final values of the parameters C and the final residuals or errors F(I); compare these with the errors we found with a quadratic. The output from the PRINT is

```
X

          0.00000        .20000        .40000
           .60000        .80000
C

          1.03607        .97734       2.68392
         -1.02356
F

           .00012       -.00051        .00078
          -.00052        .00013
```

You may be interested to know that the data was in fact generated from the exact model y = exp(x) + exp(1-x) by adding small random normal perturbations to the exact y values.

FEATURE ARTICLES

FORTRAN FUTURES

This article is part of a series on the work of the FORTRAN Standards Committee X3J3 in producing the next revision of the FORTRAN standard. The reader is reminded that the features described in this article are not a part of any FORTRAN compiler currently available, but rather are proposed requirements for FORTRAN processors produced in the late 1980's through the mid 1990's. Although every effort has been made to accurately describe the current position of X3J3 on these matters, the development of a revision to the FORTRAN standard is an evolutionary process, and these proposals may be subject to refinement, revision, or even retraction. Comments on these proposals may be given to Kurt Hirchert of the CSO Systems Consulting staff, who is a member of X3J3.

The 85th meeting of X3J3 was held February 7-11, 1983 in Chapel Hill, North Carolina. Activity at that meeting included the following:

- The primary focus of this meeting was on the merging of text describing the proposals already passed with the text of the 1978 FORTRAN standard. This process is expected to help identify conflicts and holes in the proposals. In addition, it is hoped that the resulting text may serve as the basis for the text of the new standard. There were both extensive discussion and some formal votes on some of the problem areas already identified.

- Previously, X3J3 had been working under a model in which some language features might be standardized in "language extension modules" rather than the "core" language. Standard conforming FORTRAN processors would not be required to implement the features in language extensions modules, but processors which did implement such features would be expected to follow the modules. More recently, the committee has found that the new abstraction tools being added to the language may now be powerful enough to express these extensions in terms of features in the core language. Consequently, the committee has voted to drop the concept of a language extension module. (Modules of features expressed in the core language may still be supported.)

- The SHARED DATA subprogram was renamed the BUNDLE subprogram with changes that make it a better tool for packaging abstractions. Similarly, the INHERIT statement was renamed the USE statement and extended in several ways to make it a more useful tool for accessing a packaged abstraction.

- Initial proposals on abstraction tools such as user-defined operators were presented, but not acted upon. Formal votes on proposals in this area are expected at future meetings.

- The concept of putting names on constructs such as the block-IF, previously adopted and then omitted from the list of features to be included in the next revision of the standard, was restored to the list of new features for the next standard.

- In an effort to make internal procedures more like external procedures, several restrictions that applied only to internal procedures were removed.

- A proposal for creating one-dimensional array constants was adopted. Action on creating multi-dimension arrays was deferred.

- Array assignment statements under the control of a WHERE statement were constrained not to involve transformational functions or operators. The WHERE statement may be subject to a major reworking in the next few meetings.

- Because of the facilities in the PACK and UNPACK statements were expected to have relatively limited usage, the PACK and UNPACK statements were replaced by PACK and UNPACK functions.

- There was further discussion of the event/error/exception handling proposal being developed for FORTRAN, with emphasis in this presentation on arithmetic errors like overflow.

- Possible future array extensions work was discussed. In addition to the reworking of the WHERE statement, this included cleanup on the IDENTIFY statement and the addition of explicitly allocated arrays.

- A proposal on providing a facility for specifying aliases for variable, parts of structures, sections of arrays, etc. was considered, but tabled because of problems in the text of the specific proposal brought forward.

- Proposals were discussed to provide INQUIRE keywords corresponding to OPEN keywords already added.

- Brief progress reports were heard on other work in progress including specifying minimum requirements on the accuracy of arithmetic in FORTRAN.

- Several additional interpretations of the 1978 FORTRAN standard were formally adopted.

X3J3 TO MEET IN CHAMPAIGN-URBANA

The 86th meeting of X3J3 will be held May 9-13, 1983 on the University of Illinois campus. X3J3 meetings are open to the public and all interested parties are invited to attend. If you are planning on attending some or all of these sessions, it would be appreciated if you would contact Kurt Hirchert of the CSO systems consulting office, who is making the meeting arrangements. Since much of the work of X3J3 takes place over a period of several meetings, it will be helpful if you familiarize yourself with the work done by X3J3 at its last few meetings. Kurt Hirchert can help provide you with this background.

MISCELLANEOUS

ATTENTION: TEKTRONIX EQUIPMENT USERS

For the first time, the University will have a maintenance contract with Tektronix. It will start at 15% discount and will grow with added equipment. In order to help it grow, we need to update our records of users of Tektronix terminals (not oscilloscopes or test equipment). Whether or not you are interested in the maintenance contract, please identify the terminal(s) you have, their options, and their locations.

We will send details of the arrangement for maintenance if you are interested. Address note to:

Cliff Carter
195 DCL

(PLEASE NOTE: NO phone calls, please!)

HELP WANTED

HALF-TIME PROGRAMMER POSITION

The High Energy Physics Group, Department of Physics, is looking for an advanced undergraduate or graduate student to work approximately 20 hours per week. At least one year of programming experience with both FORTRAN and Assembler language is required. (Students who have completed Computer Science courses where Fortran or Assembler are heavily used MAY be able to substitute courses for the experience requirement.)

Computers used by the group include a DEC KI10 running the TOPS-10 monitor and DEC LSI-11/2's and LSI-11/23's running RT-11. A VAX-11/780 is on order. VMS will be used on it. Members of the programming staff maintain monitors, system programs and libraries for these computers, do consulting with the users and are also involved with some microcoded special hardware for the data taking activities of the group.

Applicants should be able to commit to at least 18 months in the position.

To be considered for the position, submit a resume (it may be an informal one) along with a completed questionnaire which may be obtained from D. Schoeps or J. Adomaitis, 437 Loomis Laboratory of Physics.

ASSISTANT STATISTICIAN IN BIOMETRY

Duties: Full time academic position for planning, writing, documentation, and maintenance of FORTRAN computer programs. One half time will be spent on developing programs for and the analysis of data from the International Soybean Program (INTSOY), and half time will be spent serving other programs in the Agronomy Department, with emphasis on the research programs of the departmental statisticians. This work is primarily in regard to statistical techniques useful in Agronomic research, teaching, and extension programs. Person must also provide consulting assistance to graduate students and faculty in data preparation and use of the SAS and other statistical software.

Training: B.S. or M.S. in Applied Statistics or a closely related area, or B.S. or M.S. in Agricultural Science with some training in Applied Statistics. Candidates should have sufficient knowledge of computer programming languages, statistics, agriculture, and writing skills to permit the logical analysis of computing problems, the selection and implementation of appropriate computing algorithms, and the documentation of computer programs. Actual experience using SAS or other statistical software and in writing FORTRAN computer programs is necessary. Fluent English (reading, writing, and speaking) is required.

Salary: Commensurate with qualifications.

Starting Date: March 1, 1983, or as soon as possible.

Application: Applications will be accepted until February 1, 1983 or thereafter until a qualified candidate is obtained. Candidates should send their curriculum vitae, copies of transcripts, and three letters of recommendation as soon as possible to:

Dr. D. A. Holt, Head
Department of Agronomy
Turner Hall
University of Illinois
1102 South Goodwin Avenue
Urbana, IL 61801
Telephone: (217) 333-3420

Please cite announcement number, Announcement 183-F, when applying.

The University of Illinois is an Equal Opportunity Employer.

OFF-LINE's Mailing List

If you wish to be placed on our mailing list for future issues of *OFF-LINE*, if you wish to be removed from the list, or if you wish to enter an address correction, please complete and return this page. **(Current subscribers are kept on the mailing list until a specific request for removal is received, or until a mailing is returned as undeliverable.)**

Check one:

- ☐ Place my name on mailing list
- ☐ Make the following corrections or changes
- ☐ Delete my name from mailing list

_____

First name -- Initial -- Last Name

Campus Address:

_____

Room - Building

_____

Department

Off-campus Address:

_____

Organization or Company (if applicable)

_____

Street Address

_____

City -- State -- Zip Code

If address correction, give old address and zip code below.

_____

_____

_____

SEND TO:

OFF-LINE
150 Digital Computer Laboratory
University of Illinois at Urbana-Champaign
1304 West Springfield Avenue
Urbana, Illinois 61801

Computing
Services
Office

Off-Line

University of Illinois at Urbana-Champaign VOL. 11, NO. 4 April 1983

EDITOR: Lynn Bilger
PHONE: (217) 333-6236
120 Digital Computer Lab
1304 W. Springfield Ave.
Urbana, Illinois 61801

CSO DIRECTORY - STAFF AND SERVICES

Administrative

Director	George Badger	150	DCL	333-4103
Business Manager	Stanley Rankin	150	DCL	333-6530
Secretary	Joyce McCabe	150	DCL	333-1637

User Services and Hardware/Software Support

User Accounting		**1208**	**W Springfield**	**333-7752**
Distribution Office		**1208**	**W Springfield**	**333-6760**
Systems Consulting		**1208**	**W Springfield**	**333-6133**
Statistical Services Consulting		**85**	**Comm West**	**333-2170**
Text Processing Consulting		**118**	**DCL**	**333-7318**
Terminal Repair Service		**150**	**DCL**	**333-0969**
CYBER Dial-up Numbers	**CYBER 175**	**110-300**	**baud**	**333-4000**
	CYBER 175	**1200**	**baud**	**333-4001**
	CYBER 174	**110-300**	**baud**	**333-4004**
Asst Dir User Services	Robert Penka	173	DCL	333-4709
Asst Dir Systems and Operations	Sandra Moy	177	DCL	333-4703
Asst Dir Development	J. M. Randal	175	DCL	333-9772
Asst Dir Engineering and Hardware Selection	Cliff Carter	195	DCL	333-3723
Manager Accounting Services	Gary Bouck	1208	W Springfield	333-7752
Manager Statistical Services	Beth Richardson	189	DCL	333-6276
Documentation	Lynn Bilger	120	DCL	333-6236
CYBER-IBM Operations	Jack Knott	194a	DCL	333-6562
UNIX Operations	Debbie Hudson	123	DCL	333-8150
Laboratory Support Project	Mike Gardner	18	DCL	333-7904
RJE Operations North	Rex Duzan	162	DCL	333-6285
RJE Operations South	Don McCabe	1208	W Springfield	333-2171 333-7752

RJE Sites (see Reference Guide RF-0.3 for operating hours)

Agriculture	M103	Turner Hall	333-8170
Chemistry	153	Noyes Lab	333-1728
Commerce West	70	Comm West	333-4500
CRH Snack Bar	120	Snack Bar	333-1851
DCL Routing Room	14	DCL	333-6203
Electrical Engineering	146	EEB	333-4936
Florida Ave Res Hall		FAR	333-2695
Illinois St Res Hall		ISR	333-0307
Mechanical Engineering	65	MEB	333-1430
Psychology	453	Psych Bldg.	333-7815
Social Science	202	Lincoln Hall	333-0309

CSO operates a CDC CYBER 175 with 262K words of central memory and a CDC CYBER 174 with 196K words of central memory. The 175 and 174 run under the NOS Operating System and share 512K words of ECS. The 175 serves over 200 simultaneously active text and graphics terminals and the 174 serves over 100 simultaneously active terminals. CSO also operates an IBM 4341 with 4 million bytes of memory running HASP-OS/MVT under VM. In addition, CSO operates a DEC VAX 11/780 with 4 million bytes of memory, running under the UNIX Operating System and driving a GSI CAT-8 phototypesetter.

Operating Hours (see HEARYE,SCHEDUL for exceptions):

	CYBERS 174/175	IBM
M-F	8 am to 6 am	8 am to 6 am
SAT	8 AM - Midnight	8 AM - 6 AM
SUN	Noon - 6 AM	Noon - 6 AM

POLICY

MACHINE NAMES

With the acquisition of a second CYBER and a second IBM, it became obvious that a logical naming scheme was necessary to differentiate the various machines in our documentation and for purposes of access via various local and national networks. The staff members of CSO have had intensive discussions about the choice of names for the various machines, and have adopted the following naming conventions as a result of these discussions:

Name	*Machine Referenced*	*Operating System*
NOSA	CDC CYBER 175	NOS
NOSB	CDC CYBER 174	NOS
UXC	VAX 11/780	UNIX
VMD	IBM 4341 Model M2	VM
VME	IBM 4341 Model L10	VM

The attempt has been to reference each machine by its operating system followed by an alphabetical letter which could be used as a short-hand form of address in some instances (e.g., if only the CYBERs were being discussed, one could talk about A and B rather than always using the terms CYBER 175 and CYBER 174, respectively).

For persons across the country who access our machines via a network system, the naming scheme should work equally as well. The only difference would be that they would need to prefix the name of the machine they wish to use with the letters UIUC (e.g., to access the IBM 4341 Model M2, they would use UIUCVMD).

In conjunction with the adoption of this naming scheme, CSO is encouraging persons using the Gandalf switch to immediately begin using the above names to access the machine they want. For example, when using a terminal on the switch, type **NOSA** instead of **175** to connect to the CYBER 175 or **NOSB** instead of **174** to connect to the CYBER 174. The old switch names will remain in effect for a period of time to allow all users to become familiar with the new names.

SOFTWARE REGISTRY

CSO is trying to establish a registry of privately acquired or developed software on its machines as a means of encouraging the sharing of such software. If you have software that you think might be of use to others, please tell us

- its name (including version number if appropriate)
- a brief description of its function
- its availability (which machine(s) it runs on, whether there are legal restrictions on who may use it)
- the name and address of someone to contact for more information

You can send this information to Kurt Hirchert by campus mail (129 DCL) or the Cyber message system (TELL,HIRCHERT). If you have any questions about the registry, you can call Kurt at 333-8627.

CYBER

NEW SUBROUTINES FOR FULL ASCII INPUT AND OUTPUT TO TERMINALS

New subroutines for full ASCII input and output to terminals are now available via UOILIB (see the article about updates to UOILIB under the MATHEMATICAL SERVICES section). They may be obtained by entering:

GRAB,UOILIB.

These ASCIO subroutines are used to output the full ASCII character set to a terminal. Specifically, they are intended for use when control characters are to be used, and interline characters must be controlled (interline characters are normally carriage return and line feed).

These requirements are often necessary when communicating with a graphics terminal, or performing cursor addressing on a standard terminal, or sending certain control sequences to a microcomputer.

Note that these subroutines will write ASCII characters to a terminal only. They cannot be used in any way to create a print file. If written to disk, the file cannot be edited.

These subroutines require the files INPUT and OUTPUT to be listed on your program statement (FTN only), and not be overridden on the LGO control statement. The subroutines may be called from FTN or FTN5. Other languages will probably not be able to use them due to the use of FORTRAN I/O statements.

The ASCII input and output is done using transparent mode input and output. Although this is described in the NOS Reference Manual, Volume 2, Chapter 12, the description provided there is not easily understood by the casual programmer.

These are four subroutines in this package. Three are used for output, and one is used for input. The routines are:

Subroutine ASC1CH (ICHAR)
Subroutine ASCNCH (NCH, ICHARS)
Subroutine ASCOUT
Subroutine ASCIN (NCH, ICHARS, IRECV)

The routines all use ASCII characters stored as integer constants, variables or array elements. One ASCII character is stored in each constant, variable or array element. The ASCII values must be integers from 0 to 127 inclusive (or octal values from 0 to 177).

Variables may be assigned ASCII values as decimal or octal constants. For example, to assign the JAY variable, the ASCII value of the capital letter J, you may use

```
JAY = 74                (using the standard FTN or FTN5 decimal constant)
  or
JAY = 112B              (using the FTN form of an octal constant)
  or
JAY = 0"112"            (using the FTN5 form of an octal constant)
```

ASCII Output

You may output one character with a call to ASC1CH, or several characters by calling ASCNCH. The characters passed to either subroutine are saved until you call ASCOUT. (In fact, the saved-up characters are automatically output after 72 are collected, but you shouldn't count on this.) You should always call ASCOUT when

1. you are about to switch to standard FORTRAN input or output,
2. you are done outputting ASCII characters, or
3. at some point before your program terminates.

See the individual writeups on ASC1CH and ASCNCH for additional details on their usage and parameters. These writeups are available through the MATH procedure (see the end of this article for information on how to obtain them).

Examples of ASCII Output

To output one single ASCII character, say a bell, pass the ASCII value of the character to ASC1CH, then call ASCOUT, as in:

```
IBELL = 7
CALL ASC1CH (IBELL)
CALL ASCOUT
```

To output several ASCII characters, say escape and form feed, you may call ASC1CH twice, or ASCNCH once, as in:

```
CALL ASC1CH (27)
CALL ASC1CH (12)
CALL ASCOUT
```

or

```
INTEGER ESCFF(2)
DATA ESCFF / 17,12 /

CALL ASCNCH (2, ESCFF)
CALL ASCOUT
```

ASCII Input

To input a line of full ASCII characters from a terminal, call ASCIN. The three parameters are:

NCH	The maximum number of characters you wish to input in this call to ASCIN. NCH must be less than or equal to 72. Input will be accepted until a carriage return is entered, or until NCH characters have been entered.
ICHARS	An integer array of length NCH or greater. This array will receive the ASCII characters one per word. If fewer than NCH characters are entered, the remaining elements of ICHARS will contain ASCII blanks (=32).
IRECVD	An integer variable which will contain the actual number of characters typed, not including the carriage return (if any).

The user will receive no prompt from the system (no question mark). You may issue your own prompt using ASC1CH or ASCNCH.

See the writeup on ASCIN for additional details on usage and parameters. This writeup is available through the MATH procedure (see the end of this article for more information on how to obtain the writeup).

Examples of ASCII Input

To read up to 20 characters from the terminal, use

```
INTEGER LINE(20)

CALL ASCIN (20, LINE, IRECVD)
```

When this call is executed, suppose the following was typed:

```
DO RE ME carriage return
```

The program would find IRECVD=8, the number of characters entered, and LINE(1)=68 (a capital D); LINE(2)=79 (a capital O); and so on, with LINE(9) through LINE(20)=32 (a space).

To obtain source code and a writeup for a UOILIB routine, you should enter the following CYBER commands:

```
GRAB,MATH.
MATH,UOILIB,routine.
```

This will provide you with two local files, SOURCE (containing the complete source, usually FORTRAN), and DOC (containing a short writeup on how to call the routine).

STATISTICAL SERVICES

SPSSX ON THE IBM

A new release of the SPSS batch system, SPSSX, is now available on the IBM 4341 computer. It may be accessed by using the following statements:

```
/*ID REGION=512K
// EXEC SPSSX
```

SPSSX stands for SPSS Extended and it represents a significant evolutionary change in the package. SPSS Inc. began work on the X release in 1977 and it required a rewrite of virtually all central system facilities. Among its many new features SPSSX can accommodate intricate file and record structures including nested or hierarchical files, free-field records, and records of variable length. The AGGREGATE, FACTOR, and FREQUENCIES procedures have been completely redesigned and there is a new LOGLINEAR procedure. "Productivity raised to the highest power" is the theme for the SPSSX software package.

MANUALS

SPSSX User's Guide

This comprehensive manual contains complete operational documentation for the SPSSX system. It is designed to be both a reference to all SPSSX facilities and a guide to their use. For the person who wants to use this manual for a reference there are:

- syntax summaries at the start of every chapter
- a thumb index with the key on the back cover
- a detailed table of contents and a very thorough index
- a removable, pocket-sized reference card

For the person who is new to SPSSX there are:

- introductory chapters on files and syntax rules
- annotated examples in each chapter

For users of SPSS Release 9 there is an appendix entitled "Help for Old Friends" which explains how to make the transition from SPSS to SPSSX. Copies of this manual have been ordered from McGraw-Hill and will be available soon at the CSO Distribution Office at 1208 W. Springfield.

SPSSX Introductory Statistics Guide

This is a revision of the SPSS Introductory Guide which is currently in wide use. It contains statistical overviews and limited operational instructions. Each chapter features a statistical problem and the SPSSX output useful for its solution, followed by detailed information about the SPSSX commands

needed to obtain the analysis. The manual contains exercises and an answer key, and it may be a useful supplement in courses that integrate the teaching of statistics and computing. This manual will be available later this spring.

SPSSX Statistical Algorithms

This manual documents the computational methods and statistical algorithms used in SPSSX procedures. Users of Release 9 will also be able to use this manual, where algorithms differ both the old and the new are provided. Copies of this manual have been ordered from SPSS Inc. and will be available soon at the CSO Distribution Office, 1208 W. Springfield.

Other SPSSX Manuals

SPSS Inc. has plans for more manuals including an advanced statistics guide and a basic get-acquainted manual. As new manuals become available from SPSS Inc. they will be announced in future issues of OFF-LINE.

NOTE

The remainder of this article describes the differences between SPSS Release 9 and SPSSX and is intended for current SPSS users who wish to make the transition to SPSSX.

SYNTAX

SPSS Inc. made every attempt to keep the SPSSX language natural and straightforward. The rule about the specification field beginning in column 16 was abandoned. Program statements are called commands in SPSSX, and commands begin with a command keyword and most require specifications. Commands may be entered in any case you wish, SPSSX preserves upper and lower case within labels and literals. Command keywords begin in column 1, while specifications begin at least one space after the command keyword. Continuation lines are indented at least one column. There is no end-of-command punctuation, a new command keyword beginning in column 1 signifies the end of the previous command. The first word of every command keyword must be spelled out in full, while all subsequent keywords (except WITH) that make up the command can be truncated to three characters. SPSS allowed many special characters in variable names but SPSSX is more restrictive. Variable names must start with one of the 26 letters A-Z or with one of the characters @, or #, or $. The remaining characters in the name can be any letter, any digit, a period, @, #, or $. A new feature of SPSSX is the ability to define scratch variables. If you begin a variable name with a # SPSSX understands that this is a scratch variable to be used only for convenience in defining the file or transforming the data. Scratch variables are not available in procedures and are not saved on system files. SPSSX provides several system variables that you can use in transformation commands. A $ in the first position of a variable name indicates that this is a system variable. For example, the system variable $CASENUM for each case is the number of cases read up to and including that case. The SPSSX $CASENUM variable is the SEQNUM variable of SPSS. The special system variables are listed on page 106 of the SPSSX User's Guide. You can have blank lines in the SPSSX program file and in-line comments are a new feature. In-line comments are enclosed within the symbols /* and */ in any command line wherever a blank would be valid. Of course the most reasonable place for the command is at the end of the line in which case the closing */ is optional as in:

```
IF (RACE EQ 1 AND SEX EQ 1) SEXRACE=1                    /* WHITE MALES
```

DEFINING DATA

The SPSS VARIABLE LIST and INPUT FORMAT commands have been dropped from SPSSX and the specifications of these two commands are combined on the SPSSX DATA LIST command. In SPSS when raw data is to be read in from disk or tape it needs to be specified on a DD statement named FT08F001. In SPSSX the DDNAME may be any 1 to 8 character name, and the FILE subcommand on the DATA LIST command is used to indicate the DDNAME of the file to be read in. For example, if the dataset named USER.P9999.ILL.CORN resides on an IBM disk it can be read into SPSSX as follows:

```
// EXEC SPSSX
//CORN DD DSN=USER.P9999.ILL.CORN, ...
//SYSIN DD *
DATA LIST  FILE=CORN/ ...
```

By pointing to DD statements on the DATA LIST command SPSSX has done away with the restriction of one file in a run, and you can have several DATA LIST commands in one program file. You can omit the FILE subcommand only when the data are included in the same file with the SPSSX commands. In this case the commands BEGIN DATA and END DATA are required to separate data lines from SPSSX command lines. These two commands replace the READ INPUT DATA and END INPUT DATA commands of SPSS Release 9. The BEGIN DATA command must be placed immediately before the first line of data and the END DATA command must be placed immediately after the last line of data. If there are several SPSSX procedure commands included in the job, the order of commands is the same as in SPSS Release 9. That is, the BEGIN DATA command followed by the lines of data and the END DATA command follow the first procedure command, and additional procedure commands follow the END DATA command. In SPSS Release 9 numeric variables are stored in single precision and alphanumeric variables are restricted to 4 characters. SPSSX stores numeric variables in double precision and stores alphanumeric variables in a series of contiguous double words. Variables which have alphanumeric or character values are called string variables in SPSSX and the maximum length of a string variable is 255 columns. The limit on the number of variables in SPSSX is 32,767, in SPSS Release 9 the limit is 1000 variables. When reading numbers SPSSX ignores blanks to the left or right of the number which means that numbers typed without decimal points no longer need to be right justified in their respective fields. When a value in the data is undefined SPSS stores a blank, while SPSSX assigns the "system missing value" which is a number of the magnitude –.7D+76. The SPSS INPUT MEDIUM command has been dropped from SPSSX because its function is replaced by the FILE subcommand on the DATA LIST command, and the N OF CASES command is unnecessary. The SPSS VAR LABELS command is named VARIABLE LABELS in SPSSX. The label is specified as a literal enclosed in either apostrophes or quotation marks and can consist of any printable character as in:

```
VARIABLE LABELS  SALARY 'SALARY IN 1982'
```

A label cannot be continued from one command line to the next as the same literal, but literals can be concatenated with the + sign as in:

```
VARIABLE LABELS  SALARY 'EMPLOYEE YEARLY SALARY'
  + ' IN 1982'  YRHIRED 'YEAR OF FIRST HIRING'
```

The label on the VALUE LABELS command is also enclosed in apostrophes or quotation marks as in:

```
VALUE LABELS  SEX 1 'MALE' 2 'FEMALE'
```

The Release 9 syntax for VARIABLE LABELS and VALUE LABELS is also accepted. One of the changes in SPSSX is that if you assign value labels to a variable that already has value labels assigned to it, the new assignment completely replaces the old assignment. In SPSS variable values were printed by default as integers and you needed to use the PRINT FORMATS statement to override this. However in

SPSSX the default formats are based on the format type used on the DATA LIST command. The default format is F8.2 for free format data and for variables created by transformation commands. To change the default formats use the FORMATS command. When SPSSX reads in the data it creates an "active file" which consists of the data and a dictionary containing variable definitions. The active file is the file that you modify using transformation commands and that you save as an SPSSX system file. You can refer to the active file with an * substituting for the file DDNAME on other commands.

SYSTEM FILES

The SPSSX commands to save and read system files are the SAVE command and the GET command. These commands replace the SAVE FILE and GET FILE commands of SPSS Release 9. In SPSS Release 9 a system file to be saved required a DD statement named FT04F001, a system file to be read required a DD statement named FT03F001, and system files had an internal file name which was specified on the SAVE FILE and GET FILE commands. SPSSX system files don't have an internal file name and the DDNAME of the DD statement which describes a system file can be any 1 to 8 character name. The OUTFILE subcommand on the SAVE command and the FILE subcommand on the GET command specify the DDNAME of the DD statement which describes the system file to saved or read.

Example of saving an SPSSX system file:

```
necessary ID statements

// EXEC SPSSX
//SURVEY DD DSN= ...
//SYSIN DD *
    .
    .
    .
SAVE  OUTFILE=SURVEY
    .
    .
    .
/*
```

Example of reading an SPSSX system file:

```
necessary ID statements

// EXEC SPSSX
//TREES DD DSN= ...
//SYSIN DD *
GET  FILE=TREES
    .
    .
    .
/*
```

The SAVE command can be placed in the program file anywhere that a procedure can be placed. When your system file contains many variables and you only want to use a few you can use the KEEP subcommand on the GET command to select a subset of variables. When the KEEP subcommand is used only those variables selected will be written to the active file. SPSSX can read system files created by SPSS Release 9. However SPSSX system files include more dictionary information and have a different

structure from SPSS system files, so SPSSX reads SPSSX system files more efficiently than old format system files. If you intend to use your old format system file or archive file with SPSSX you should use the SAVE command during your first SPSSX run to save it as an SPSSX system file. If you have alphanumeric variables on your old format file and did not use a PRINT FORMATS command when you created the system file, you will need to use a REFORMAT command in SPSSX. This is explained on page 757 of the SPSSX User's Guide. If you used the keyword THRU on MISSING VALUES to declare a range of missing values for an alphanumeric variable in an old format file, the missing values may not be handled correctly in SPSSX, and you may need to redeclare the missing values in SPSSX. SPSS was more liberal than SPSSX in allowing special characters in variable names. When an old format file is read by SPSSX special characters in variable names (except a period, or @, or #, or $) are changed to periods. SPSSX stores numeric values in double precision while Release 9 used single precision, therefore the space required for storage of a system file is increased. However, SPSSX has a new feature that allows you to compress your system files and scratch work files. Use the keyword COMPRESSED on the SAVE command to write a compressed system file and the COMPRESSION=YES specification on the SET command to write the scratch file in compressed form. Compression reduces the storage requirements for files that contain small integer values because it maximizes the number of values stored in each computer word. Compressed files will be considerably smaller than uncompressed files if most of the data are small integers. There is an overhead with compression however, because the processing time for saving and retrieving compressed files is greater than for uncompressed files.

TRANSFORMATIONS

Transformations in SPSSX are performed much faster than in SPSS Release 9. The ASSIGN MISSING command has been dropped in SPSSX. Propagation of missing values in SPSSX is similar to the ASSIGN MISSING command except that the value propagated is always the "system missing value". If a case is missing on any of the variables used in an expression or if an operation cannot be evaluated then SPSSX returns the system missing value. New variables in SPSS are initialized to zero, in SPSSX they are initialized to the system missing value which is a number of the magnitude -.7D+76. By default SPSSX assigns the print formats of F8.2 for new numeric variables created by the RECODE, COMPUTE, COUNT or IF commands. If this is not appropriate you can use the FORMATS command to reset it. You can use the INTO keyword on the RECODE command to create a new variable as a recoded version of another variable. The following example stores into VOTER the recoded values of AGE.

```
RECODE  AGE(18 THRU HI=1)(0 THRU 18=0) INTO VOTER
```

There are a large number of new functions available on the COMPUTE command. Some are statistical functions that allow you to compute statistics such as the mean or variance across variables for each case. The following example adds together 100 variables and stores the result in the variable X.

```
COMPUTE  X=SUM(V1 TO V100)
```

Implied operators or expressions on the IF and SELECT IF commands are allowed in SPSS but they are not allowed in SPSSX. The following kind of statement is illegal in SPSSX:

```
IF (X EQ 3 OR 5 OR 7) Y=5
```

The new functions ANY and RANGE are useful alternatives to implied operators. For example, the following two commands are equivalent.

```
IF (X GT 0 AND X LT 10) Y=5
IF (RANGE(X,1,9)) Y=5
```

Likewise the next two commands are equivalent.

```
IF (X EQ 3 OR X EQ 5 OR X EQ 7) Y=5
IF (ANY(X,3,5,7)) Y=5
```

You can conditionally execute one or more transformation commands with the new DO IF-END IF structure, and the structure can be further defined with the ELSE AND ELSE IF commands. Parentheses enclosing logical expressions are optional in SPSSX on the IF, SELECT IF, and DO IF commands. Temporary transformations are no longer signified with an asterisk in column 1 of the command. Instead, use the new TEMPORARY command to signal the beginning of temporary transformations that are only in effect for the next procedure. The new LEAVE command retains the values of variables from one case to the next allowing you to build cumulative indexes. There are many new functions available for manipulating string variables. There are functions for concatenating strings, padding them on the left or right, and converting strings to numbers or numbers to strings. The following example uses the new SUBSTR function to select cases for which the first two characters of the state variable are IL.

```
SELECT IF  (SUBSTR(STATE,1,2) EQ 'IL')
```

The new STRING command is available when you want to create new string variables which are not defined on a DATA LIST command.

PROCEDURES

In SPSS Release 9 when statistical results such as Z scores, regression residuals, discriminant scores, or factor scores are produced they are written to a file described on a DD statement named FT09F001, and they are not available for use in the same job. In SPSSX when Z scores are requested from the CONDESCRIPTIVE procedure they are written to the active file as fully defined new variables, and they can be used by other SPSSX procedures in the same job. The same thing can be done in SPSSX with residuals and predicted values from REGRESSION, discriminant scores from DISCRIMINANT, and factor scores from FACTOR. The LOGLINEAR procedure is a new procedure in SPSSX which does model fitting, hypothesis testing and parameter estimation for a model that has categorical variables as its major components. The NEW REGRESSION procedure in SPSS Release 9 has been renamed REGRESSION in SPSSX. The AGGREGATE procedure has been completely redesigned for SPSSX and its syntax completely differs from its SPSS syntax. Among its many new features AGGREGATE can now produce a system file which can be used by other SPSSX procedures in the same job. To accomplish this you specify after the OUTFILE subcommand the DDNAME of the DD statement which describes the aggregated system file. The function of the SPSS RAW OUTPUT UNIT command has been replaced by the PROCEDURE OUTPUT command in SPSSX. To write out a correlation matrix from the PEARSON CORR procedure specify OPTIONS 4, and specify the file to which the matrix is written on a PROCEDURE OUTPUT command. In the following example a correlation matrix is written to the file referenced by the DDNAME CORREL.

necessary ID statements

```
// EXEC SPSSX
//GRADS DD DSN= ...
//CORREL DD DSN= ...
//SYSIN DD *
GET  FILE=GRADS
PROCEDURE OUTPUT  OUTFILE=CORREL
PEARSON CORR  AGE TO OCCUP
OPTIONS  4
```

The syntax of the FREQUENCIES procedure has been completely revised. Results that were formerly requested on OPTIONS and STATISTICS commands are now requested on subcommands. For example, the following command requests a median and a histogram on the job prestige variable.

```
FREQUENCIES VARIABLES=PRESTIGE/
 STATISTICS=MEDIAN/HISTOGRAM
```

Both histograms and bar graphs are now available in the FREQUENCIES procedure and the normal curve can be superimposed on a histogram. The FACTOR procedure has been completely redesigned for SPSSX. Among the many new features three new extraction methods are available: unweighted least squares, generalized least squares, and maximum likelihood. The numerous changes in procedures are documented in the "Help for Old Friends" appendix of the SPSSX User's Guide.

MATHEMATICAL SERVICES

UPDATES TO UOILIB

UOILIB is a local library containing mathematical and utility routines collected from various sources. Though it is highly eclectic, this is no reflection on the quality of the software it contains -- some excellent software, such as sparse linear system solvers and differential equations routines, is contained therein. UOILIB is described in Reference Guide RF-4.10. It can be accessed on the CYBERs by the command

```
GRAB,UOILIB.
```

following which you can compile and run a CYBER FORTRAN program. The UOILIB routines have been compiled with FTN, so calling them from FTN5 programs may not work. An FTN5 version of the library is in the works.

Several changes have been made in the UOILIB library within the last month, as follows:

1. The catalog has been re-organized according to functional areas, which should make it much easier to see if UOILIB has a routine for a particular function. To print a copy of the new catalog, enter the following commands:

   ```
   GRAB,MATH.
   MATH,UOILIB.
   PRINT,DOC/RJE=xxx.
   ```

   where *xxx* designates some print site.

2. A set of 4 routines for doing 7-bit ASCII input/output with a terminal has been added. These are described in another article (under the CYBER section) in this issue of OFF-LINE.

3. The permanent file routine PF has been renamed to XPF, to avoid conflict with the new system PF routine. The new PF is called the same way as the old. The old one was renamed so that source code could remain conveniently available for those who wish to tailor their own permanent file routine or to study the techniques used by PF.

4. Two differential equations algorithms published in Transactions on Mathematical Software were added:

    a. Algorithm 497 – Automatic integration of functional differential equations, by Kenneth W. Neeves, published in the December 1975 issue – was added as routine DMRODE. This algorithm is intended to solve equations involving such things as lags, e.g., $y' = y(t-1)$.

    b. Algorithm 504 – Global error estimation for ordinary differential equations, by L. F. Shampine and H. A. Watts, published in the June 1976 issue – was added as routine GERK. It is based on the Runge-Kutta-Fehlberg algorithm, which appears in several other packages which we have (RKF45 in UOILIB; FORSIM; SLAM).

To obtain source code and a writeup for a UOILIB routine, you should enter the following CYBER commands:

```
GRAB,MATH.
MATH,UOILIB,routine.
```

This will provide you with two local files, SOURCE (containing the complete source, usually FORTRAN), and DOC (containing a short writeup on how to call the routine).

TRANSACTIONS ON MATHEMATICAL SOFTWARE

We now have on tape all algorithms published in Transactions on Mathematical Software up to December 1982 (from algorithm 493 published in the first issue of March 1975, up to algorithm 591). The algorithms from the March 1983 issue have been ordered, but have not yet arrived.

All TOMS algorithms are kept on a public tape; file TOMS/UN=MATHLIB contains a directory of this tape, plus directions for reading particular files on the tape. To print a copy of this directory, enter:

```
GET,TOMS/UN=MATHLIB.
PRINT,TOMS.
```

The "Math Note" series described in the March 1981 issue of OFF-LINE includes a note summarizing all algorithms published in TOMS to date. To print this note (it is note number 6), enter the following:

```
GET,MNOTES/UN=MATHLIB.
MNOTES,6.
PRINT,OUT/AS/CC/EJ.
```

(To print instructions on the use of MNOTES, substitute a 0 (zero) for the 6 in the example above.)

The algorithms in the December 1982 and the March 1983 issues are as follows (the March 1983 algorithms have not yet arrived):

The algorithms published in the December 1982 issue are:

588: Fast Hankel Transforms Using Related and Lagged Convolutions, by Walter L. Anderson

589: SICEDR: A FORTRAN Subroutine for Improving the Accuracy of Computed Matrix Eigenvalues, by Jack J. Dongarra

590: DSUBSP and EXCHQZ: FORTRAN Subroutines for Computing Deflating Subspaces with Specified Spectrum, by P. Van Dooren

591: A Comprehensive Matrix-Free Algorithm for Analysis of Variance, by William J. Hemmerie

535: remark on algorithm 535. The QZ algorithm to solve the generalized eigenvalue problem.

539: remark on algorithm 539. Basic Linear Algebra Subprograms for FORTRAN Usage

580: remark on algorithm 580. QRUP: A Set of FORTRAN Routines for Updating QR Factorizations

The algorithms in the March 1983 issue are:

592: A FORTRAN subroutine for Computing the Optimal Estimate of f(x), by P. W. Gaffney

593: A Package for the Helmholtz Equation in Nonrectangular Planar Regions, by Wlodzimierz Proskurowski

594: Software for Relative Error Analysis, by John L. Larson, Mary E. Pasternak, and John A. Wisniewski

595: An Enumerative Algorithm for Finding Hamiltonian Circuits in a Directed Graph, by Silvano Martello

573: remark on Algorithm 573. NL2SOL - An Adaptive Nonlinear Least-Squares Algorithm

The following articles which appeared in these issues may be of interest to some of our users:

From the December 1982 issue:

Checking the Calculation of Gradients, by Phillip Wolfe

Fast Hankel Transforms Using Related and Lagged Convolutions, by Walter L. Anderson (This is the support paper for algorithm 588.)

From the March 1983 issue:

A Functional Description of ANALYZE: A Computer-Assisted Analysis System for Linear Programming Models, by Harvey J. Greenberg

A Reduced Gradient Algorithm for Nonlinear Network Problems, by P. Beck, L. Lasdon, and M.Engquist

If you are looking for software, Transactions on Mathematical Software is one of the journals to check, along with "Numerische Mathematik", "BIT", the "Computer Journal", and various publications of ACM (the Association for Computing Machinery) and SIAM (the Society for Industrial and Applied Mathematics).

In the Systems Consulting Office at 1208 W. Springfield, we have bound volumes containing all the algorithms published in Communications of the ACM from 1960 to 1975 (when the algorithms department of CACM became Transactions on Mathematical Software). The loose-leaf service of ACM (which sends us updates of the algorithms) includes an index going back to 1960. This index includes many algorithms from journals other than CACM.

SOLVING DIFFERENTIAL EQUATIONS WITH PROTRAN

The January-February issue of *OFF-LINE* introduced a new problem-solving language called PROTRAN, from IMSL, Inc. The March issue showed how PROTRAN might be used in a curve-fitting problem. In this article we will show a few simple examples of using PROTRAN to solve differential equations.

As we remarked in the last article, there is now an on-line writeup on PROTRAN, which can be obtained and printed by entering the following commands

```
WRITEUP,PROTRAN.
PRINT,PRODOC/AS/CC/EJ/RJE=xxx.
```

where *xxx* designates some print site on campus (see Reference Guide RF-0.3 Job Entry Sites for a list of site codes). Due to copyright restrictions, the writeup does not include a full description of all PROTRAN features; this appears in the *MATH/PROTRAN User's Manual*, published by IMSL, which can be purchased at the CSO Distribution Office for $15 or ordered from IMSL. Reference Guide RF-4.32 PROTRAN summarizes how to access PROTRAN and includes a one-sheet summary of all PROTRAN procedures; once one is acquainted with PROTRAN, this summary sheet should be all that is required to set up a PROTRAN program.

Let us solve the differential equation

$$y'(t) = y(t) \quad y(0) = 1 \quad 0 \leq t \leq 1.0$$

This is admittedly simple, but it will show how easy it is to express a differential equation with PROTRAN. In solving this, we will save information about the solution and use it to see how closely the numerical solution satisfies the equation. There are two ways of saving the solution of a differential equation in PROTRAN: as an array of points representing the solution at equally spaced values of the independent variable; or as an actual function routine which can be called to evaluate the solution at an arbitrary point. (Of course, in this latter case, what PROTRAN does is to save solution information in a "hidden" place -- when you call the solution function at some later point, this hidden information is used in an interpolation scheme; the interpolation scheme is based on the order of the Runge-Kutta method used to solve the equations, not on cubic splines.) The solution will be saved both ways.

Here is the first program. The solution is saved in the array SOL, and $DERIVATIVE is used to estimate the derivatives of this solution using cubic splines.

(The dollar signs are in column 1.)

```
      PROGRAM MAIN(OUTPUT,TAPE6=OUTPUT)
$     DECLARATIONS
      VECTOR  SOL(21), A(21) , DER(21) , DIFF(21)
$     DIFEQU Y'=F(X,Y); ON(0.0,1.0); INITIAL=1.0; SOLUTION=SOL
      NOUTPUT = 20
      DEFINE
      =====
      F = Y
      =====
$     ASSIGN  A(I) = (I-1)/20.0
$     DERIVATIVE SOL; VS A; AT A; IS DER
$     ASSIGN  DIFF = SOL - DER
$     PRINT DIFF ; FORMAT=EMAXP
$     END
```

(Recall that, to run this, you enter

```
GRAB,PROTRAN.
PROTRAN,xxx.
```

where *xxx* is the program file; then LGO.)

The differential equation itself is specified by the FORTRAN statement between the ===== lines. The vector SOL is dimensioned to 21 points because NOUTPUT specifies that 20 solution points are to be saved; the initial point counts as an extra point, hence 21. NOUTPUT is taken to be 100 if it is omitted. If SOL were NOT dimensioned at all, it would instead construct a function routine named SOL, as the second program below does. After the solution is calculated, vector A is filled with values of the independent variable corresponding to the solution points, then $DERIVATIVE is called to find the derivative of the spline through these points; the derivative values are returned in DER. Then DIFF is calculated as the difference between the derivative and the solution; notice the simplicity of this statement -- many vector and matrix computations can be written simply in PROTRAN. The PRINT statement displays the values of DIFF; the name "DIFF" is written on the output file, followed by its contents, neatly formatted; the option "FORMAT=EMAXP" tells PROTRAN to print the contents with an E-format with as many places as possible.

Here is the output of the first program

```
DIFF

 -.23560854437221E-04    .63589048764356E-05   -.16557226629743E-05
  .49425852211016E-06   -.79232357563797E-07    .77161651290680E-07
  .38124376544602E-07    .51596657613118E-07    .51165351067084E-07
  .54577682817580E-07    .57296333011436E-07    .59763728188500E-07
  .64788387987846E-07    .60738187812603E-07    .91380272237984E-07
 -.66743837123795E-08    .37641441963387E-06   -.10352708130767E-05
  .42521558327735E-05   -.15460871097162E-04    .58130086017627E-04
```

So the calculated solution seems pretty good.

Here is the second program

```
            PROGRAM MAIN(OUTPUT,TAPE6=OUTPUT)
$           DECLARATIONS
            VECTOR  A(IA=101), DER(IDER=101), DIFF(IDIFF=101)
$           ASSIGN  NOUT=20
                    IA = NOUT+ 1; IDER=NOUT+ 1; IDIFF=NOUT+ 1
$           DIFEQU Y'=F(X,Y); ON(0.0,1.0); INITIAL=1.0; SOLUTION=SOL
            NOUTPUT = NOUT
            DEFINE
            =====
            F = Y
            =====
$           ASSIGN  A(I) = (I-1)/FLOAT(NOUT)
$           DERIVATIVE SOL(X,1); D(X); AT A ; IS DER
$           ASSIGN  DIFF(I) = SOL(A(I),1) - DER(I)
$           PRINT DIFF ; FORMAT=EMAXP
$           END
```

In this program, SOL is not dimensioned; thus, PROTRAN creates a function routine named SOL which can be called to evaluate the solution. This function is called as SOL(X,N), where N is the component of the solution being evaluated and X is the independent variable; in this case there is only one solution component, so N=1. The $DERIVATIVE procedure is called to differentiate SOL(X,1) with respect to X; since a function form is given, a finite difference technique is used to evaluate the derivative rather than cubic splines. This example also shows the use of "range variables" in PROTRAN: in the VECTOR declaration, each vector has been declared to have 101 elements, and a variable (the range variable) has been set equal to that number. Later, the assumed size of the vector can be changed by simply changing the value of the range variable, as is done in the following $ASSIGN. This way, the program can be instructed to save more points simply by changing the value of NOUT; other things that need changing are then altered automatically (up to the dimensioned maximum of 101). In this case, the output is

DIFF

-.37806636427717E-04	.59455572056777E-05	.17413884947359E-05
-.17670593308594E-05	.41109661452765E-06	.43217392686756E-06
-.36829720926335E-05	-.23362028613860E-05	-.16881828628357E-05
-.14676179560524E-05	-.14660843987713E-05	-.13616141032458E-05
-.71287357883421E-06	.64472799721216E-06	.67778356083181E-06
.71253457178955E-06	.75424566148286E-07	.87165824709245E-05
.26144113803639E-04	-.26406849684690E-04	.81739174120798E-04

Lastly, we will solve a very simple boundary value problem using a shooting technique. PROTRAN is not really to be recommended in general for boundary value problems; the current release lacks a specific procedure for such problems. For 2-point boundary value problems, one might use the routines DTPTB (shooting method) and DVCPR (finite difference method) in the IMSL library, or DD04AD in the Harwell library. This will demonstrate, however, that it is not difficult to formulate a simple shooting method using PROTRAN.

The problem is

$$y'' (t) = y (t) \quad y (0) = 0 \quad y (1) = 1$$

The solution is y(t) = a*(exp(t) - exp(-t)) where a = exp(1)/(exp(2)-1). We try to guess what y'(0) is; if this is known, the solution is determined from its initial condition. We set it up as the solution of a nonlinear equation: the independent variable is the guess G of y'(0); the equation is y(1)-1 = 0; we use $DIFEQU to calculate y(1) given the value of G. Notice that the system is written as two first-order equations. The program also shows how a FORTRAN routine can be included in a PROTRAN program.

```
      PROGRAM MAIN(OUTPUT,TAPE6=OUTPUT)
$     DECLARATIONS
      VECTOR  FUN(1) , G(1) , GSTART(1)
$     ASSIGN  GSTART = (0)
$     NONLIN  FUN(G)=0; GUESS = GSTART
      DEFINE
      =====
      CALL SUB(G,FUN)
      =====
$     PRINT G,FUN; FORMAT=EMAXP
$     FORTRAN
      PRINT *,'EXACT SOLUTION=',2*EXP(1.)/(EXP(2.)-1)
$     END
      SUBROUTINE SUB(G,FN)
```

```
C
C  SOLVE  Y"=Y  Y(0)=0  Y'(0)=G AND  RETURN FN = Y(1) - 1
C
$          DECLARATIONS
           VECTOR  Y(2), F(2)
           MATRIX  SOL(2,2)
$          ASSIGN  Y = (0 , G)
$          DIFEQU Y'=F(X,Y); ERRTARGET=.000001
           INITIAL = Y
           ON(0,1)
           DEFINE
           ====
           F(1) = Y(2)
           F(2) = Y(1)
           ====
           NOUTPUT = 1
           SOLUTION = SOL
$          ASSIGN  FN = SOL(2,1) - 1
$          PRINT G,SOL; FORMAT=EMAXP
$          END
```

$NONLIN (the procedure for solving a nonlinear equation) has a slight peculiarity: even when you have but one equation to solve, it must be formulated as a vector equation of dimension 1; thus the dimensions of 1 in the VECTOR declaration. The output from this program is:

```
G

               0.
SOL
                         1                        2

 1             0.                      0.

 2             0.                      0.
G

               -.68005380911811E-08
SOL
                         1                        2

 1             0.                      -.68005380911811E-08

 2             -.79919999781486E-08    -.10493778176751E-07
G

                .85091818482430E+ 00
SOL
                         1                        2

 1             0.                       .85091818482430E+ 00

 2              .10000000561812E+ 01    .13130353637998E+ 01
```

```
G

                    .85091813701868E+00
SOL
                          1                        2

 1                  0.                        .85091813701868E+00

 2                  .10000000000000E+01       .13130352900319E+01
NONLIN SOLUT

                    .85091818482430E+00
G

                    .85091813701868E+00
FUN

                    0.
EXACT SOLUTION=.8509181282393
```

Each time the differential equation is solved by routine SUB, it prints the solution; notice that PRINT displays SOL neatly as a table. SOL(2,1) is y(1), hence the result of SUB is given by SOL(2,1)-1.

FEATURE ARTICLES

FORTRAN PROGRAMS AND FILES

One essential aspect of effective FORTRAN usage is the means by which input/output statements are associated with specific files and devices. On the CYBERs, this issue is surprisingly complex, not because such associations are difficult, but because there are so many ways they may be accomplished. Since each method has its own advantages and disadvantages, the wise FORTRAN programmer will be aware of them all, so the right method may be chosen for each application.

No matter which methods are used, each FORTRAN input/output statement transfers data between the program and a single file. Thus, none of these methods can be used to do things like changing a program that writes only to the terminal into one which writes both to the terminal and to a local file to be printed. One must either modify the program or run it twice, once to produce the terminal output and once to produce the disk file output.

Each input or output statement references some unit name. For statements with a unit parameter, the unit name is determined by the value of the unit parameter. If it is an integer value in the range 0 to 99 for FTN4 or 0 to 999 for FTN5, the unit name is TAPEn where n is the integer value. For example, the statement

```
READ(5,100) I,J
```

references the unit name TAPE5. Unit names may be specified more directly as left justified Hollerith data with binary zero fill.

For example, the FTN4 statements

```
IUNIT=6LINDATA
READ(IUNIT,*) NUMBER
```

and the FTN5 statement

```
READ(L"INDATA",*) NUMBER
```

both reference the unit name INDATA. Input statements that do not have a unit specified, such as

```
READ 10000, SAMPLE
```

or the FTN5 statement

```
READ(*,10000) SAMPLE
```

reference the unit name INPUT. Similarly, output statements that do not have a unit specified, such as

```
PRINT *, X
```

or the FTN5 statement

```
WRITE(*,*) X
```

reference the unit name OUTPUT. Finally, run-time error messages are written referencing the unit name OUTPUT and output from the C$ DEBUG package (FTN4 only) is written referencing the unit name DEBUG. In FTN4, unit names are limited to six characters, but in FTN5 they may have seven.

The unit name is used to look up an associated record manager control block called a file information table or FIT. The FIT is used to control the actual input or output. The association between a unit name and a FIT can be established at compile time by including the unit name on the PROGRAM statement. A new FIT is created if the unit name is listed alone or with a parameter specifying the buffer size and/or maximum line length. Alternatively, a unit name may be specified to be associated with the same FIT as some unit name appearing earlier on the PROGRAM statement, thus effectively making it a synonym for the unit. For example, the statement

```
PROGRAM MAIN(INPUT,OUTPUT=/300,TAPE5=INPUT,TAPE6=OUTPUT)
```

associates unit names INPUT and OUTPUT with FITs created for the them (the latter having a maximum line length of 300 rather than the default of 150) and associates unit names TAPE5 and TAPE6 with the FITs already created for INPUT and OUTPUT respectively.

In FTN4, this is the only way unit names can be associated with FITs. In FTN5, association may be established (and, where appropriate, new FITs created), when an OPEN statement is executed or a statement references a unit name not already associated with a FIT.

The FIT contains a field which normally contains the local file name to be read or written through that FIT. When FITs are created by FORTRAN, this field is initialized to be the same as the unit name to be associated with that FIT, but this field can be changed before it is actually used.

When execution of a FORTRAN begins, a FORTRAN library routine looks to see if any parameters were specified on the statement that began execution of the program. If so, the first parameter is substituted for the local file name in the FIT for the first unit name on the PROGRAM statement, the second

parameter for the local file name in the FIT for the second unit name, and so on. For example, if the statement

```
LGO,DATA,RESULTS.
```

were used to begin a program having the PROGRAM statement in our example above, DATA would become the local file name in the FIT associated with unit names INPUT and TAPE5 and RESULTS would become the local file name in the FIT associated with unit names OUTPUT and TAPE6. If the local file name of a particular FIT is not to be changed, the corresponding parameter may either be omitted (if it appears at the end of the parameter list) or explicitly indicated as missing by successive commas. Thus, in our example,

```
LGO,DATA.
```

would change the local file name only for INPUT and TAPE5 and

```
LGO,,RESULTS.
```

would change the local file name only for OUTPUT and TAPE6. Note that there is no requirement that INPUT precede OUTPUT on the PROGRAM statement, so if you think that you will be substituting a different local file name for OUTPUT more often than INPUT, it may be sensible to put OUTPUT first.

The local file name field of a FIT may also be affected by the FTN5 OPEN and CLOSE statements. The FILE= parameter of the OPEN statement can be used to directly specify the local file name to be place in the FIT. (If there is a FIT containing this local file name already, the unit name for the OPEN will simply be associated with that FIT rather than with a newly created FIT. Thus unit name "synonyms" may also be created dynamically.) The CLOSE statement ends the association between its unit name and the currently associated FIT. Thus, if that unit name is used again, it will be with a new FIT with a new local file name (possibly set by default). The CLOSE statement also ends the association between the FIT being closed and any other unit names currently associated with it, not just the unit name referenced by the CLOSE statement.

Record manager performs initialization related to a FIT when an FNT5 OPEN statement is executed or when the first input/output statement using that FIT is executed. Part of that initialization is a check to see whether a record manager FILE statement has been issued for the local file name specified in the FIT. The LFN= parameter on the FILE statement can be used to specify a local file name to replace the one in the FIT before initialization is complete. Caution: Various portions of the FORTRAN library check the local file name field of the FIT without considering whether the name there is the one before or after record manager initialization, so the use of this feature can produce confusing results when names are not unique.

Finally, one can control the specific file or device accessed through a particular local file name. Most standard commands creating local files have some means of controlling the local file name used. For example, the first parameter of a LABEL command specifies the local file name to be used to access the tape. The permanent file commands, such as GET and ATTACH, use the permanent file name as the local file name by default, but have an alternate form where the permanent and local file names can be specified separately. If local file name is used which doesn't correspond to any existing local file, an empty temporary disk file will normally be created for it. There are a few names, like INPUT and OUTPUT, which are handle specially. In a batch job, INPUT is a local file containing the batch input and OUTPUT is a local file which will be printed automatically when the job completes. In a timesharing session, INPUT and OUTPUT will normally both refer to the terminal. (By convention, one normally reads from the terminal using INPUT and writes to it using OUTPUT, but a program which read using OUTPUT and writes using INPUT will work just as well.) The ASSIGN statement can be used to override these defaults. For example,

```
ASSIGN,TT,INFILE.
```

causes the local file name INFILE to be associated with the terminal and

```
ASSIGN,MS,OUTPUT.
```

causes the local file OUTPUT to be a temporary disk file. If a local file already exists, but you need it to have a different local file name, the RENAME command can be used.

The principles are illustrated by the following procedure:

```
.PROC,TEST.
$GET,LFN=PFN.
$LABEL,TAPLFN,VSN=MYTAPE,F=S,CV=EB,LB=KU.
$FILE,FITLFN,LFN=TAPLFN,BT=E,RT=F,FL=80,MBL=3200,MNB=80,MNR=80.
$ASSIGN,MS,OUTPUT.
$FTN5,I=TESTPGM,L=0,REW.
LGO,LFN.
$RENAME,OUT=OUTPUT.
$REWIND,OUT.
$COPY,OUT,OUTPUT.

.DATA,TESTPGM
1       PROGRAM MAIN(INPUT,TAPE5=INPUT)
2       CHARACTER*80 LINE
3       READ(5,'(A)',END=6)LINE
4       PRINT *,LINE
5       GO TO 3
6       PRINT *,'--EOF ON TAPE5--'
7       REWIND 5
8       READ(L"INPUT",'(A)',END=11)LINE
9       WRITE(L"OUTPUT",*)LINE
10      GO TO 8
11      WRITE(L"OUTPUT",*)'--EOF ON INPUT--'
12      CLOSE(UNIT=5)
13      OPEN(UNIT=L"INPUT")
14      READ(*,'(A)',END=17)LINE
15      WRITE(*,*)LINE
16      GO TO 14
17      WRITE(*,*)'--EOF ON INPUT--'
18      REWIND(UNIT=L"INPUT")
19      READ(L"TAPE5",'(A)',END=22)LINE
20      PRINT *,LINE
21      GO TO 19
22      PRINT *,'--EOF ON TAPE5--'
23      CLOSE(UNIT=L"INPUT")
24      OPEN(UNIT=L"INPUT",FILE='FITLFN')
25      READ(L"INPUT",'(A)',END=28)LINE
26      WRITE(L"OUTPUT",*)LINE
27      GO TO 25
28      WRITE(L"OUTPUT",*)'--EOF ON INPUT--'
29      CLOSE(UNIT=L"INPUT")
30      END
```

A typical execution of this procedure at a terminal produces (after a long wait because of the tape mount) the following output:

```
THESE LINES ARE CONTAINED IN PERMANENT FILE PFN.  IN OUR EXAMPLE,
THE GET COMMAND MAKES A LOCAL COPY OF PFN UNDER THE LOCAL FILE NAME
LFN.
--EOF ON TAPE5--
THESE LINES ARE CONTAINED IN PERMANENT FILE PFN.  IN OUR EXAMPLE,
THE GET COMMAND MAKES A LOCAL COPY OF PFN UNDER THE LOCAL FILE NAME
LFN.
--EOF ON INPUT--
THESE LINES ARE BEING ENTERED FROM THE TERMINAL, WHICH IS ASSOCIATED WITH
THE LOCAL FILE NAME INPUT.
--EOF ON INPUT--
--EOF ON TAPE5--
THESE LINES ARE CONTAINED ON THE TAPE WITH VSN MYTAPE.  IN OUR
EXAMPLE, MYTAPE IS ASSOCIATED WITH THE LOCAL FILE NAME TAPLFN.
--EOF ON INPUT--
```

In statements 3 to 6, the program copies lines read from unit name TAPE5 to unit name OUTPUT. Since the PROGRAM statement associated TAPE5 with the same FIT as INPUT and program initialization copied the name LFN into that FIT (replacing the name INPUT that was put there at compile time), statement 3 reads from local file LFN. Since the GET statement made the permanent file PFN available under the local file name LFN, what we actually see in the output is a copy of the contents of PFN. Note that the first time statement 4 was executed, there was no FIT associated with unit name OUTPUT, so the FORTRAN library created one with local file name OUTPUT. Normally this output would have gone to the terminal directly, but because of the ASSIGN statement, it goes to a local disk file. Statement 7 then rewinds the file and statements 8 to 11 perform a similar copy between unit names INPUT and OUTPUT. Because INPUT is associated with the same FIT as TAPE5, we are reading the same file and thus get essentially the same output. Statement 12 then breaks the association between unit names TAPE5 and INPUT and the FIT created by the PROGRAM statement, so statement 13 creates a new FIT for unit name INPUT. Since no FILE= parameter was specified, the local file name used was also INPUT and statement 13 thus reads from the terminal. The first execution of statement 19 creates a new FIT for unit name TAPE5 and thus local file name TAPE5. Since there was no local file named TAPE5, the system creates an empty one and the program sees an immediate end of file. Finally, unit name INPUT is once again closed and reopened, this time with an explicit local file name of FITLFN. When the record manager initialization associated with statement 24 was executed, record manager saw the FILE statement for FITLFN was executed, so it replaced the local file name in the FIT by TAPLFN. Since this was the local file name associated with our tape by the LABEL command, the last copy loop produced lines from the tape. Note that although three different forms of output statements were used in our program, all reference the unit name OUTPUT and thus the same FIT and the same output local file. When our FORTRAN program completed, the local disk file named OUTPUT was renamed to OUT and then copied to local file name OUTPUT, which once again became associated with the terminal, thus producing our terminal output.

To summarize, the basic methods of associating a FORTRAN input/output statement with a specific file or device are as follows:

- using an appropriate unit name
- using the PROGRAM statement to make the unit name used a "synonym" for the appropriate unit name

- specifying a replacement local file names a parameter to the program execution (on the LGO statement)
- specifying the local file name using the FILE= parameter of the FTN5 OPEN statement
- changing the local file name using the LFN= parameter of the record manager FILE statement
- using NOS commands to insure that the specific file or device has the local file name being used

MISCELLANEOUS

X3J3 TO MEET IN CHAMPAIGN-URBANA

The 86th meeting of X3J3 will be held May 9-13, 1983 on the University of Illinois campus. X3J3 meetings are open to the public and all interested parties are invited to attend. If you are planning on attending some or all of these sessions, it would be appreciated if you would contact Kurt Hirchert of the CSO systems consulting office, who is making the meeting arrangements. Since much of the work of X3J3 takes place over a period of several meetings, it will be helpful if you familiarize yourself with the work done by X3J3 at its last few meetings. Kurt Hirchert can help provide you with this background.

HELP WANTED

PART-TIME PROGRAMMER NEEDED

Programmer needed to assist with horticulture data. Must have working knowledge of BASIC and experience with IBM Personal Computer. Part-time, starting as soon as possible; salary negotiable. Please contact:

James C. Schmidt or Floyd Giles
104 Ornamental Horticulture
1107 W. Dorner Drive
Urbana, IL 61801

Telephone 217-333-2125

Computing Services Office

Off-Line

University of Illinois at Urbana-Champaign VOL. 11, NO. 5 May 1983

EDITOR: Lynn Bilger
PHONE: (217) 333-6236
120 Digital Computer Lab
1304 W. Springfield Ave.
Urbana, Illinois 61801

Page *Contents*

CSO DIRECTORY - STAFF AND SERVICES

Administrative

Director	George Badger	150	DCL	333-4103
Business Manager	Stanley Rankin	150	DCL	333-6530
Secretary	Joyce McCabe	150	DCL	333-1637

User Services and Hardware/Software Support

User Accounting		**1208**	**W Springfield**	**333-7752**
Distribution Office		**1208**	**W Springfield**	**333-6760**
Systems Consulting		**1208**	**W Springfield**	**333-6133**
Statistical Services Consulting		**85**	**Comm West**	**333-2170**
Text Processing Consulting		**118**	**DCL**	**333-7318**
Terminal Repair Service		**150**	**DCL**	**333-0969**
CYBER Dial-up Numbers	**CYBER 175**	**300**	**baud**	**333-4000**
	CYBER 174	**300**	**baud**	**333-4004**
	CYBER DIAL-UP	**1200**	**baud**	**333-4008**
	IBM 4341	**300**	**baud**	**333-4006**
Asst Dir User Services	Robert Penka	173	DCL	333-4709
Asst Dir Systems and Operations	Sandra Moy	177	DCL	333-4703
Asst Dir Development	J. M. Randal	175	DCL	333-9772
Asst Dir Engineering and Hardware Selection	Cliff Carter	195	DCL	333-3723
Manager Accounting Services	Gary Bouck	1208	W Springfield	333-7752
Manager Statistical Services	Beth Richardson	189	DCL	333-6276
Documentation	Lynn Bilger	120	DCL	333-6236
CYBER-IBM Operations	Jack Knott	194a	DCL	333-6562
UNIX Operations	Debbie Hudson	123	DCL	333-8150
Laboratory Support Project	Mike Gardner	18	DCL	333-7904
RJE Operations North	Rex Duzan	162	DCL	333-6285
RJE Operations South	Don McCabe	1208	W Springfield	333-2171 333-7752

RJE Sites (see Reference Guide RF-0.3 for operating hours)

Agriculture	M103	Turner Hall	333-8170
Chemistry	153	Noyes Lab	333-1728
Commerce West	70	Comm West	333-4500
CRH Snack Bar	120	Snack Bar	333-1851
DCL Routing Room	14	DCL	333-6203
Electrical Engineering	146	EEB	333-4936
Florida Ave Res Hall		FAR	333-2695
Illinois St Res Hall		ISR	333-0307
Mechanical Engineering	65	MEB	333-1430
Psychology	453	Psych Bldg.	333-7815
Social Science	202	Lincoln Hall	333-0309

OFF-LINE is the monthly newsletter of the Computing Services Office at the University of Illinois at Urbana-Champaign. Information in this issue is current as of April 22, 1983.

CSO operates a CDC CYBER 175 with 262K words of central memory and a CDC CYBER 174 with 196K words of central memory. The 175 and 174 run under the NOS Operating System and share 512K words of ECS. The 175 serves over 200 simultaneously active text and graphics terminals and the 174 serves over 100 simultaneously active terminals. CSO also operates an IBM 4341 Model 2M with 8 million bytes of memory running HASP-OS/MVT under VM and an IBM 4341 Model L10 with 4 million bytes of memory running VM. In addition, CSO operates a DEC VAX 11/780 with 4 million bytes of memory, running under the UNIX Operating System and driving a GSI CAT-8 phototypesetter.

Operating Hours (see HEARYE,SCHEDUL for exceptions):

	CYBERS 174/175	IBM
M-F	8 am to 6 am	8 am to 6 am
SAT	8 AM - Midnight	8 AM - 6 AM
SUN	Noon - 6 AM	Noon - 6 AM

POLICY

EARLY WARNINGS -- UPCOMING CHANGES

Since this is the last issue of *OFF-LINE* during the current school year, we are publishing a set of brief "early warnings" of upcoming changes. These changes are not all definite, but because they may affect your planning, we want to let you know that they are likely. Details and schedules will appear in later issues of *OFF-LINE* and/or in HEARYE.

1. The last CS introductory course will be moving to terminals this fall. Twenty-four new terminals will be added for this purpose. We will retire almost all keypunches at that time.

2. There should be a major downward revision of CPU time costs on the CYBER systems this summer.

3. We hope to have a cluster of personal computers (about 20 to 25) available for use by August.

4. We plan to discontinue the availability of punched card output by May 1984. The lack of available spare parts, and the high cost of this service, force this action upon us.

5. An updated release of the CYBER operating system will be installed during the summer.

6. We plan to add, by August, a dot matrix printer capable of producing plots at a rate approximately 15 times faster than the current pen plotters. While the plots will be of lower quality than plots produced on the pen plotters, they will be acceptable in most cases. (This matrix printer will not replace any of the current pen plotters.)

- Telenet prices to us have increased substantially, and our charge will be increased to recover this.

FREE STUDENT ACCOUNTS

Beginning with the 1983 Fall Semester, students will be able to apply for a **Free Student Account** number at the following locations:

CSO South Operations Room 70 Commerce West	or	CSO Accounting Office 1208 W. Springfield, Urbana

The Free Student Account is available to students currently enrolled at the University of Illinois. Each student is required to present a valid University I.D. to obtain the student account.

Current restrictions on obtaining a Free Student Account are as follows:

- Persons having an active SARA account (graduate students in the process of a Masters or Doctoral Thesis) are not eligible.

- Students currently enrolled in 100 level CS courses are not eligible.

CHANGE AND EXPANSION IN 1200-BAUD DIAL-UP SERVICE

CSO has been aware for some time that the CYBER 1200-baud dial-up service does not meet the demand for the service. This has been caused by two factors, a rapid increase in the demand and separate telephone numbers for each system. A separate telephone number for each system causes CSO to statically commit a certain number of 1200-baud modems and telephone lines to a particular machine. Since the load on our machines varies by time of day and day of semester, the demand for 1200-baud service floats from system to system with the load. This means that people dialing in to a heavily loaded system will receive busy signals, even though the other systems are available. A more appropriate solution to the problem would be to have a pool of 1200-baud modems which could be allocated to the systems with the demand.

Pooling of modems has not been feasible in the past due to the lack of any suitable hardware to do the pooling. This has changed, however, with CSO's doubling of the size of the GANDALF PACX port switch. Many people have used terminals connected to the switch at various sites around campus to specify which CSO system they want to use. CSO will be doing an analogous operation with 1200-baud dial-up terminals.

CSO has purchased eight additional 1200-baud modems and telephone lines reachable through the number 3-4008. These additional modems are connected to the Gandalf PACX. Therefore, when dialing the above number, you will be connected to the Gandalf PACX switch and must then specify which system you wish to use. To do this, depress the carriage return to elicit the prompt ENTER CLASS. Respond with NOSA, NOSB, UXC, VMD, or VME, as appropriate.

As usage begins to shift from the old telephone numbers (e.g. 3-4001) to 3-4008, we will remove modems from the old numbers and add them to 3-4008 until, by summer, 1200-baud service will be available only through that number.

SOFTWARE REGISTRY

CSO is trying to establish a registry of privately acquired or developed software on its machines as a means of encouraging the sharing of such software. If you have software that you think might be of use to others, please tell us

- its name (including version number if appropriate)
- a brief description of its function
- its availability (which machine(s) it runs on, whether there are legal restrictions on who may use it)
- the name and address of someone to contact for more information

You can send this information to Kurt Hirchert by campus mail (129 DCL) or the CYBER message system (TELL,HIRCHERT). If you have any questions about the registry, you can call Kurt at 333-8627.

CYBER

SIR USER LIST TO BE COMPILED

The Survey Research Laboratory is trying to assess the extent of user interest in software produced by Scientific Information Retrieval, Incorporated (SIR). Currently, SRL provides user support for SIR/DBMS, a multipurpose data base management system designed for use with scientific and business data sets. SIR/DBMS 2.1.1 is available to users on the CYBER 175 via the SHARED facility. (See the January-February edition of *OFF-LINE* for details.)

As part of that support function, Survey Research Laboratory is compiling a list of local users of software produced by SIR. SRL will use this list to notify SIR users of additions, changes and enhancements to software products, training seminars, and useful information transmitted by SIR, Inc. If you wish to be added to this list, please specify your

University ID
Name
Campus address
Office telephone number
Staff position or title
Department or unit
SIR product(s) being used

in a message to the SIR Site Coordinator via the MESSAGE facility on the CYBER by entering

TELL,3SIRSRL

or by mailing the information to

Richard Williams
Survey Research Laboratory
1005 W. Nevada
Campus

If you are a SIR/DBMS user, please remember to specify which version you employ (V1.1, V2.1, or both).

SRL would also like to assess user interest in other software products marketed by SIR, Inc. The following products have been released by SIR or will become available in the second quarter of 1983:

SIR/FORMS	An integrated forms language for screen oriented data entry. SIR/FORMS directly interfaces with SIR/DBMS.
SIR/HOST	SIR/HOST provides a direct interface between a SIR/DBMS data base and a program written in FORTRAN or some other "host" language. The SIR/HOST subprogram library supplies all the capabilities of the SIR/DBMS Retrieval Language.
SIR/QUERY	SIR/QUERY is an interactive package for retrieving information from a SIR/DBMS data base using "English-like" sentences and the data is displayed in simple tabular form.

SIR/CENSYS — SIR/CENSYS is a "turnkey" system for processing data from the 1980 U.S. Census. It is a self-contained package built on top of SIR/DBMS and enables the user to extract census data in any form, including Census Bureau table formats.

These products are not licensed by the Computing Services Office, nor is SRL responsible for their acquisition, maintenance, or support. Users interested in software products such as those described should contact the SIR Site Coordinator for more details about a product, and to indicate the types of applications for which these products might be useful.

SIR UPDATES: DO YOU USE SIR VERSION 1.1?

On April 1, 1983, SIR 2.1.1 became the default version of the system available through the SHARED facility. To access SIR/DBMS 2.1.1, enter the command

```
GRAB,SIR/S
```

or

```
SHARED,SIR
```

To aid in transition to the new release, SIR 1.1 will be available via the command

```
GRAB,SIRV1/S
```

or

```
SHARED,SIRV1
```

As announced in the January-February issue of *OFF-LINE*, vendor support for the SIR Version 1.1 (release 0) has been reduced to maintenance-only status. The reduction in support status is part of an eventual phase-out of the Version 1.1 software product in favor of the recently released SIR/DBMS 2.1.1. Data bases created using SIR 1.1 are **NOT** compatible with the new version; however, built-in procedures exist for transporting such data bases to the newer SIR/DBMS.

The Survey Research Laboratory is currently assessing the feasibility of continued licensing and support of the older version of the system, and we would appreciate users who currently rely on SIR Version 1.1 to contact Richard Williams at SRL via the CYBER MESSAGE facility

```
TELL,3SIRSRL
```

or at

Survey Research Laboratory
1005 W. Nevada
Campus
Telephone: 333-3494

IBM

OS/SAS82.2: NEW RELEASE AVAILABLE ON IBM

The 1982 release of OS/SAS is now available on the IBM for batch processing. The current default version of SAS is OS/SAS79.6. To access the new version, use the following JCL statement:

```
// EXEC SAS822
```

instead of: // EXEC SAS (which invokes the default: SAS79.6). (OS/SAS82.2 will become the default batch SAS processor around May 1st.)

OS/SAS82.2 is a new major release of SAS with many enhancements and new features which are documented in the two new SAS user's guides:

"SAS User's Guide: BASICS" and "SAS User's Guide: STATISTICS"

which are available at the CSO Distribution Office for $15.50 each. As a complete revision of the "1979 SAS User's Guide", these manuals are a better organized, more extensively indexed, and more thorough documentation of the SAS system.

New features of special note are:

- A new MACRO FACILITY
- A HELP facility and improved NEWS option
- New PROCS: TABULATE, CALENDAR, APPEND, FORMS, TREE
- A variety of performance and procedure enhancements

A REGION SIZE USAGE NOTE FOR OS/SAS82.2

A few weeks ago in HEARYE we announced that jobs running under OS/SAS82.2 were requiring 15-20% more region than similar jobs run under OS/SAS79.6. We have now learned from SAS Institute that this increased region requirement is a fix amount rather than a percentage. The SAS Supervisor, the primary processing unit in SAS, has grown in the amount of central memory region it occupies in OS/SAS82.2 as compared with what it occupied in OS/SAS79.6 by about 45K. This means that a job requiring a region of "nnnK" to run under OS/SAS79.6 will now require a region of "(nnn+45)K" to run under OS/SAS82.2. Therefore if you plan to run a job under OS/SAS82.2 that you've run previously under OS/SAS79.6, be sure to add 45K to the region specified for the REGION= ID card parameter.

(Note: If you specify too small a region, your job ABENDs with a code of S804.)

ACCESSING THE NEW SAS MACRO FACILITY

If you intend to use the new MACRO facility under OS/SAS82.2, you will need to add the MACRO option to your EXEC SAS822 JCL statement. In this case, the statement should look as follows:

```
// EXEC SAS822,OPTIONS='MACRO'
```

The documentation indicates that you need to increase your REGION= parameter on the ID card in your job to allocate additional region for the macro language processor which is loaded when the MACRO option is specified. SAS Institute has indicated that the macro language processor requires 48K of region. Some preliminary tests on our part show that you should initially figure an additional 100K in your region estimate for the REGION= ID card parameter if you use the MACRO facility.

After the initial job run, you can more accurately specify the region required by examining the "region usage" reports following the DATA and PROC steps in the SASLOG output. For example, if the highest region used by any of the DATA or PROC steps is 380K, specify at least that much in any future job runs by coding: REGION=380K .

There may also be other SAS system options you will wish to specify. For further details, please refer to the System Options section (p. 323) and the SAS Under OS Batch appendix (p. 855) of the "SAS User's Guide: BASICS".

MATHEMATICAL SERVICES

SLIGHT CHANGE TO ACSL TEKTRONIX PLOTTING

A small adjustment has been made to the ACSL simulation package which affects plotting on Tektronix terminals. By default, ACSL had assumed (since the latest version was installed last summer) that the terminal transmission speed is 9600 baud (960 characters per second). In doing Tektronix plots, this caused a large number of delay characters to be sent to the screen to allow time for it to finish clearing. Since none of our terminals is this fast, this caused a very long delay before a plot actually appeared on one's screen. To remedy this, the assumed baud rate has been changed to 1200, which produces a much smaller delay. The ACSL user can, at run time, tell ACSL what baud rate to assume, by entering the command

```
SET TBRCPL=nnn
```

where *nnn* is the baud rate to assume (e.g., 300, 1200, 9600); this SET command must be done before the first PLOT command is entered. The ACSL manual states that the default value of TBRCPL is 9600, so your manual should be marked to show this change.

INTEGRATION WITH MATH/PROTRAN

In previous issues of *OFF-LINE*, we announced the MATH/PROTRAN package from IMSL, and gave various examples of using it. In this issue we will give yet more examples of using MATH/PROTRAN, this time in the area of integration.

Remember that PROTRAN is accessed on the CYBER systems by the command

```
GRAB,PROTRAN.
```

and that a PROTRAN program is first translated by entering the command

```
PROTRAN,prog,out
```

where *prog* designates your program file and *out* a listing file. *out* may be omitted (e.g., PROTRAN,prog) in which case the listing appears on your terminal. Once the program is translated, it can be run with LGO. Output from PROTRAN PRINT statements goes to local file TAPE6 unless you have a PROGRAM statement to redirect it elsewhere, such as in the examples below.

A writeup describing PROTRAN briefly with examples can be obtained on the CYBER systems by entering the commands

```
WRITEUP,PROTRAN.
PRINT,PRODOC/AS/CC/EJ.
```

and complete PROTRAN manuals can be purchased for $15.50 from the CSO Distribution Office at 1208 W. Springfield.

Integration is accomplished in PROTRAN using the procedure $INTEGRAL. There are two ways of using this procedure:

1. If a function is given by a series of points (X(I),Y(I)), you can give the arrays containing these values, and request that a cubic spline be fitted through them and integrated.

2. If the function can be calculated for an arbitrary argument, you can give the function directly, either as a simple expression (if the function can be so expressed), or by means of a subroutine. In this case, the integration is performed using the algorithm of IMSL routine DCADRE, which adaptively decides where to sample the function so as to obtain an accurate integral.

These approaches to integration may, of course, be combined with other PROTRAN procedures. For instance, a given set of data points may need to be smoothed first using the $APPROXIMATE procedure; the output of $APPROXIMATE might be a set of points on the smoothing curve (allowing use of integration method 1 above) or an actual FORTRAN function routine which could be used with method 2.

Suppose, for example, we have with great effort calculated the following 11 points on a curve

x	*y*
0.0	0.0
0.1	0.01
0.2	0.04
0.3	0.09
0.4	0.16
0.5	0.25
0.6	0.36
0.7	0.49
0.8	0.64
0.9	0.81
1.0	1.00

If we want to fit a spline through this set of points and integrate it from x=0 to x=1, then the following PROTRAN program would do this.

```
            PROGRAM MAIN(OUTPUT,TAPE6=OUTPUT)
$           DECLARATIONS
            VECTOR  X(11) , Y(11)
$           ASSIGN
            X = (0,.1,.2,.3,.4,.5,.6,.7,.8,.9,1.)
            Y = (0,.01,.04,.09,.16,.25,.36,.49,.64,.81,1.)
$           INTEGRAL  Y; VS X; ON(0.,1.); IS Z
$           PRINT Z
$           END
```

Note that the ON option tells what the limits of integration are to to be. If we wanted only to integrate from x=.1 to x=.9, then the ON option would be ON(.1,.9). The IS option tells PROTRAN that we want the result stored in variable Z.

Now if we can evaluate the integrand for an arbitrary argument, say y=f(x)=x**2, we can use the second form of $INTEGRAL. There are actually three variations on this form:

a. We can give the integrand directly, if it is a simple expression:

```
$           INTEGRAL  X**2; FOR(X=0,1); IS Z
```

b. We can give the integrand in terms of a subroutine which we will supply:

```
$           INTEGRAL  F(X); FOR(X=0,61); IS Z
```

and insert a FORTRAN FUNCTION named F to evaluate the integrand.

c. We can simply name the integrand, and use the DEFINE option to calculate it immediately by a series of FORTRAN statements:

```
$           INTEGRAL F; FOR(X=0,1); IS Z
            DEFINE
            ====
            F = X**2
            ====
```

Here are the complete PROTRAN programs illustrating each of these variations.

a.
```
            PROGRAM ONE(OUTPUT,TAPE6=OUTPUT)
$           INTEGRAL X**2; FOR(X=0,61); IS Z
$           PRINT Z
$           END
```

b.
```
            PROGRAM TWO(OUTPUT,TAPE6=OUTPUT)
$           INTEGRAL F(X); FOR(X=0,1); IS Z
$           PRINT Z
$           END
            FUNCTION F(X)
            F = X**2
$           END
```

```
c.          PROGRAM THREE(OUTPUT,TAPE6=OUTPUT)
      $     INTEGRAL F; FOR(X=0,1); IS Z
            DEFINE
            ====
            F = X**2
            ====
      $     PRINT Z
      $     END
```

Here are some points to note about these programs:

- As a rule, a PROTRAN procedure is called by placing a dollar sign in column 1 and the procedure name in column 7 or after. The procedure "call" may extend over several lines.

- Options on a PROTRAN procedure call are separated by placing them on separate lines or by separating them with semicolons on the same line.

- When the DEFINE option is used to introduce a function needed for a particular procedure, it must be be immediately followed by a line with = = = = in column 7,8,9,10; this delimiter must be followed by lines of FORTRAN statements (no PROTRAN allowed here) and the necessary FORTRAN must be terminated by another line with = = = = in columns 7,8,9,10.

- The last line of a PROTRAN program must be "$ END", unless the last PROTRAN procedure invoked was $FORTRAN, in which case a regular FORTRAN END is sufficient.

- As with all PROTRAN procedures which some control over the accuracy of a calculation, there is an ERRTARGET option of $INTEGRAL in this case. ERRTARGET defaults to .001, meaning it will attempt to find the integral to three significant figures, or within .001 (if the value is less than 1.0).

Last of all, we give an example of combining $INTEGRAL with other PROTRAN procedures. Suppose we must calculate the solution of

$$y' = y^2 \qquad y\ (0) = .01 \qquad 0 \leqslant x \leqslant 1.0$$

and then integrate the solution from x=0 to x=1. We will use the $DIFEQU procedure to solve the equation. Two forms of output are possible for DIFEQU: a table of values or a function (which will do interpolation on a hidden table stored in PROTRAN's "stack"). Suppose that in either case we want it to save 50 solution points (in addition to the initial point x=0,y=1); these points will be saved at equally spaced x-values.

The PROTRAN program for the first form of output is as follows

```
            PROGRAM FIRST(OUTPUT,TAPE6=OUTPUT)
$           DECLARATIONS
            VECTOR  XS(51) , YS(51)
$           DIFEQU Y'=F(X,Y); ON(0,1); SOLUTION=YS; NOUTPUT=50
            INITIAL = .01
            DEFINE
            ====
            F = Y**2
            ====
$           ASSIGN  XS(I) = (I-1)*1./50.
$           INTEGRAL  YS; VS XS; ON(0,1); IS Z
$           PRINT Z
$           END
```

The PROTRAN program for the second type of DIFEQU output is as follows:

```
            PROGRAM SECOND(OUTPUT,TAPE6=OUTPUT)
$           DIFEQU Y'=F(X,Y); ON(0,1); INITIAL=.01; NOUTPUT=50
            SOLUTION = FUN
            DEFINE
            ====
            F = Y**2
            ====
$           INTEGRAL FUN(X,1); FOR(X=0,1); IS Z
$           PRINT Z
$           END
```

The integrand is given as FUN(X,I) because the function FUN created by DIFEQU has two arguments: the independent variable, and the number of the solution component (DIFEQU can handle a system with many components).

BOUNDARY VALUE SOLVER ADDED TO UOILIB

COLSYS, a routine for solving multipoint boundary value problems in ordinary differential equations using collocation techniques, has been added to the UOILIB subroutine library. This routine was published in the June 1981 issue of *Transactions on Mathematical Software* by U. Ascher, J. Christiansen and R. D. Russel.

To use COLSYS, you must write a FORTRAN program which calls it; your program must include three auxiliary subroutines, defining the ordinary differential equations, their partial derivatives with respect to the unknowns, and the boundary conditions. COLSYS computes the solution in terms of spline functions defined at special points called collocation points. Two auxiliary routines, COLSAS and COLSAP are called after COLSYS is called, to generate values of the solution at other points.

A writeup explaining its use from the UOILIB library can be obtained and printed by entering:

```
GRAB,MATH.
MATH,UOILIB,COLSYS.
PRINT,DOC.
```

Other routines for solving boundary value problems in ordinary differential equations are:

1. DTPTB (in IMSL). This routine handles only 2-point boundary value problems, i.e., those whose boundary conditions are defined only at the endpoints of the interval of interest, not at intermediate points. A multiple shooting technique is used. (The basic idea of shooting is to guess what the missing initial conditions at the left endpoint are, then integrate -- "shoot" -- across the interval and revise one's initial guesses according to the observed error in the boundary conditions. With multiple shooting, one shoots from several intermediate points simultaneously, hoping by the shortening of the intervals of integration to achieve faster convergence.) The "shots" are performed using the Runge-Kutta algorithm of IMSL routine DVERK, making this routine inappropriate for stiff systems.

2. DVCPR (in IMSL). This routine also handles only 2-point problems, using a finite difference technique.

3. DD04AD (in HARWELL). This routine uses the same finite difference technique as DVCPR in IMSL.

Presently, COLSYS is the only boundary value routine we have which purports to solve multipoint problems.

DOCUMENTATION

NEW AND REVISED REFERENCE GUIDES

Following is a list of the new and revised Reference Guides now available at the RJE sites. Note that with the addition of another IBM, it has become necessary to add a new IBM/CMS category. To keep the two IBM categories together, we have moved the Microcomputer Software category from 20 to 30, and made the IBM/CMS category number 20. Also note that Reference Guide RF-01 REFERENCE GUIDE LIST has been updated; we recommend that users pick up this latest version to keep abreast of recent additions and/or changes.

Reference Guides

RF-0.1	Reference Guide List	Revised 04/15/83
RF-0.3	Job Entry Sites (RJE)	Revised 04/15/83
RF-0.11	LocalNet	Revised 04/07/83
RF-0.14	Names for CSO Computer Systems	New 04/15/83
RF-1.2	CYBER Rates	Revised 03/10/83
RF-4.1	ACSL	Revised 04/08/83
RF-4.19	GRG	Revised 04/08/83
RF-4.32	MATH/PROTRAN	New 02/15/83
RF-11.2	IBM Rates	Revised 03/10/83
RF-17.8	RESTORE	Revised 04/12/83
RF-20.1	IBM/CMS LOGON PROCESS	New 04/11/83

MISCELLANEOUS

X3J3 TO MEET IN CHAMPAIGN-URBANA

The 86th meeting of X3J3 will be held May 9-13, 1983 on the University of Illinois campus. X3J3 meetings are open to the public and all interested parties are invited to attend. If you are planning on attending some or all of these sessions, it would be appreciated if you would contact Kurt Hirchert of the CSO systems consulting office, who is making the meeting arrangements. Since much of the work of X3J3 takes place over a period of several meetings, it will be helpful if you familiarize yourself with the work done by X3J3 at its last few meetings. Kurt Hirchert can help provide you with this background.

SPECIAL TEAROFF SHEET

UPDATE OF LIST OF MINI, MICRO, AND PERSONAL COMPUTER USERS

CSO is planning to update its list of Mini, Micro, and Personal Computer Users. The list includes the use of specialized computer-related equipment, software packages, or databases not provided by CSO.

The impetus for this documentation stemmed from the frequent questions directed to CSO in these areas. A researcher who is facing a problem involving computers often suspects that someone else on campus has already dealt with a similar problem. CSO is the logical place to start trying to contact such a person.

This document, available on-line via WRITEUP,SURVEY, will continue to be updated periodically. Minimal effort will be spent editing the information received, and inclusion of an item will not imply further knowledge or support by CSO. We are simply trying to provide a useful starting place for those users seeking to contact others.

Please fill in and return the form on the reverse side if

- you are currently on the list and wish to remain on it (entries currently on the list for persons who do not respond, or let us know that they wish to remain on the list, will be deleted);
- you are using a minicomputer, microcomputer, or personal computer in your work;
- you have acquired experience using equipment for automated control, measurement, data collection, or data conversion; or
- you have acquired or developed software packages or data bases which could be of interest to others and which you are willing to share.

If you are a CYBER user you can use the APPEND command to send us the information through a CYBER file, COMPUSE, rather than using the form. To do this, simply put your responses into a local file (named *filex* in the example below) and then enter the following control statement:

APPEND,COMPUSE,*filex*/UN=DOCUMNT.

Or, fill out and return the form to:

Editor, Documentation
Computing Services Office
120 Digital Computer Laboratory
University of Illinois at Urbana-Champaign
1304 W. Springfield Avenue
Urbana, IL 61801

If you are on campus, use campus mail to return the form.

Name:_____

Address: _____ Phone:_____

If you are using a personal minicomputer or microcomputer, please supply the following information:

Type of Computer_____

Peripheral hardware_____

Operating system_____

How are you using the equipment?_____

_____

If you've developed or acquired special purpose equipment which is used in conjunction with a computer, please tell us about it._____

_____

_____

_____

If you've developed or acquired software packages or databases which could be useful to someone else, please tell us about it._____

_____

_____

_____

Other comments or information_____

_____

_____

_____

OFF-LINE's Mailing List

If you wish to be placed on our mailing list for future issues of *OFF-LINE*, if you wish to be removed from the list, or if you wish to enter an address correction, please complete and return this page. **(Current subscribers are kept on the mailing list until a specific request for removal is received, or until a mailing is returned as undeliverable.)**

Check one:

- ☐ Place my name on mailing list
- ☐ Make the following corrections or changes
- ☐ Delete my name from mailing list

_____

First name -- Initial -- Last Name

Campus Address:

_____

Room - Building

_____

Department

Off-campus Address:

_____

Organization or Company (if applicable)

_____

Street Address

_____

City -- State -- Zip Code

If address correction, give old address and zip code below.

_____

_____

_____

SEND TO:

OFF-LINE
150 Digital Computer Laboratory
University of Illinois at Urbana-Champaign
1304 West Springfield Avenue
Urbana, Illinois 61801

Computing
Services
Office

Off-Line

University of Illinois at Urbana-Champaign VOL. 11, NO. 6 June 1983

EDITOR: Lynn Bilger
PHONE: (217) 333-6236
120 Digital Computer Lab
1304 W. Springfield Ave.
Urbana, Illinois 61801

Page *Contents*

CSO DIRECTORY - STAFF AND SERVICES

Administrative

Director	George Badger	150	DCL	333-4103
Business Manager	Stanley Rankin	150	DCL	333-6530
Secretary	Joyce McCabe	150	DCL	333-1637

User Services and Hardware/Software Support

User Accounting		**1208**	**W Springfield**	**333-7752**
Distribution Office		**1208**	**W Springfield**	**333-6760**
Systems Consulting		**1208**	**W Springfield**	**333-6133**
Statistical Services Consulting		**85**	**Comm West**	**333-2170**
Text Processing Consulting		**118**	**DCL**	**333-7318**
Terminal Repair Service		**150**	**DCL**	**333-0969**
CYBER Dial-up Numbers	**CYBER 175**	**300**	**baud**	**333-4000**
	CYBER 174	**300**	**baud**	**333-4004**
	CYBER DIAL-UP	**1200**	**baud**	**333-4008**
	IBM 4341	**300**	**baud**	**333-4006**
Asst Dir User Services	Robert Penka	173	DCL	333-4709
Asst Dir Systems and Operations	Sandra Moy	177	DCL	333-4703
Asst Dir Development	J. M. Randal	175	DCL	333-9772
Asst Dir Engineering and Hardware Selection	Cliff Carter	195	DCL	333-3723
Manager Accounting Services	Gary Bouck	1208	W Springfield	333-7752
Manager Statistical Services	Beth Richardson	189	DCL	333-6276
Documentation	Lynn Bilger	120	DCL	333-6236
CYBER-IBM Operations	Jack Knott	194a	DCL	333-6562
UNIX Operations	Debbie Hudson	123	DCL	333-8150
Laboratory Support Project	Mike Gardner	18	DCL	333-7904
RJE Operations North	Rex Duzan	162	DCL	333-6285
RJE Operations South	Don McCabe	1208	W Springfield	333-2171 333-7752

RJE Sites (see Reference Guide RF-0.3 for operating hours)

Agriculture	M103	Turner Hall	333-8170
Chemistry	153	Noyes Lab	333-1728
Commerce West	70	Comm West	333-4500
CRH Snack Bar	120	Snack Bar	333-1851
DCL Routing Room	14	DCL	333-6203
Electrical Engineering	146	EEB	333-4936
Florida Ave Res Hall		FAR	333-2695
Illinois St Res Hall		ISR	333-0307
Mechanical Engineering	65	MEB	333-1430
Psychology	453	Psych Bldg.	333-7815
Social Science	202	Lincoln Hall	333-0309

OFF-LINE is the monthly newsletter of the Computing Services Office at the University of Illinois at Urbana-Champaign. Unless otherwise indicated, permission to reprint is freely granted, provided that the author, if named, and the Computing Services Office (CSO) are credited. Information in this issue is current as of May 22, 1983.

CSO operates a CDC CYBER 175 with 262K words of central memory and a CDC CYBER 174 with 196K words of central memory. The 175 and 174 run under the NOS Operating System and share 512K words of ECS. The 175 serves over 200 simultaneously active text and graphics terminals and the 174 serves over 100 simultaneously active terminals. CSO also operates an IBM 4341 Model 2M with 8 million bytes of memory running HASP-OS/MVT under VM and an IBM 4341 Model L10 with 4 million bytes of memory running VM. In addition, CSO operates a DEC VAX 11/780 with 4 million bytes of memory, running under the UNIX Operating System and driving a GSI CAT-8 phototypesetter.

Operating Hours (see HEARYE,SCHEDUL for exceptions):

	CYBERS 174/175	IBM
M-F	8 am to 6 am	8 am to 6 am
SAT	8 AM - Midnight	8 AM - 6 AM
SUN	Noon - 6 AM	Noon - 6 AM

POLICY

OVERHEAD CHARGES

All sponsored research accounts operating under the overhead structure of "modified total direct cost" are handled and flagged automatically through the UFAS accounting system. Charges made on a ledger 5 account (sponsored project) will have an additional charge of 9.4% added by the UFAS accounting system against an object code of 5569 for FY 84. Questions concerning this matter should be directed to the Grants and Contracts Office.

SHORT COURSES - SUMMER SESSION 1983

As announced in past issues of *OFF-LINE*, some of the manuals formerly distributed free by CSO are now being sold. Access to the following manuals is strongly recommended for certain of the courses listed below. These documents may be purchased at the Illini Union Bookstore (Reference Section), 715 South Wright Street, Champaign.

1.	Introduction to the CYBER Systems	$2.00
2.	A Tutorial Guide to the ICE Text Editor	$1.25
3.	ICE Reference Manual	$3.25
4.	RNF Documentation: Tutorial, Macros and Reference	$4.00
5.	An Index to Software on the CYBER	$3.25

CSO is offering the following free, noncredit short courses during the Summer Session 1983 to acquaint potential users with our computing systems, facilities and services.

Those planning to attend short courses numbered 1 through 6 may register by telephoning 333-6630 during office hours. Registrants for all other short courses are assumed to know the fundamentals of using a CYBER terminal and are requested to use the interactive on-line registration procedure:

```
BATCH.          (if you are not already in this subsystem)
GET,SIGNUP/UN=COURSES.
SIGNUP.
```

Please note that:

REGISTRATION IS REQUIRED for all courses.

Use of the SIGNUP procedure assumes that you have the current short course listing before you for reference. This document may be examined on-line via:

```
TYPE,COURSES/AS/UN=COURSES.
```

A brief list of course titles and meeting times is available via:

```
TYPE,TITLES/AS/UN=COURSES.
```

Due to circumstances beyond the control of CSO, some room assignments may have to be changed; users should check HEARYE for updates.

CSO reserves the right to cancel courses or sections with insufficient enrollment.

Updates (changes) since the printed short course listing was issued may be examined on-line via:

```
TYPE,CHANGES/AS/UN=COURSES.
```

If you wish to drop (cancel your registration in) a short course, you may use this on-line procedure in the BATCH subsystem:

```
GET,DROP/UN=COURSES.
DROP.
```

Questions, comments and suggestions should be addressed to the CSO short course coordinator: Ron Szoke, 333-8630 or TELL,SZOKE from a CYBER terminal.

Short Course Listing

1. Orientation to CSO Facilities and Services

A brief, nontechnical presentation for prospective users on the following topics: the mission and organization of CSO; computing equipment; available software; locations of facilities and hours of operation; where to obtain documentation; short courses and video tapes; consulting and other services; how to set up a computer account. Prerequisites: none. Recommended manual: 5, above. Four sections will be offered:

June 10	12N-1pm	168 Elect Engr	[Szoke]
June 13	12N-1pm	168 Elect Engr	[Richardson]
June 15	3pm-4pm	168 Elect Engr	[Alster]
June 20	12N-1pm	168 Elect Engr	[Edwards-Iwe]

2. Computing for Poets

An orientation to computers and computing for those in the humanities and fine arts. Intended especially for those with "computer anxiety" and those who are "anti-computer" or "anti-technology." The course assumes no background in computing, mathematics, or any other technical field.

July 11,13,15	12N-1pm	131 English	[Szoke & Kyle]

3. The Writer and the Computer

A practical, laboratory/demonstration course showing how the writer can use a microcomputer and word processing package to produce publication-ready manuscripts. Covers revision skills for all forms of composition: creative writing, critical essay, historical report, etc. Also how to access the on-line library catalog. Prerequisite: none, but Computing for Poets (above) or equivalent knowledge of basic computer terminology is recommended.

July 18,19,20,21	12N-1pm	131 English	[Szoke & Kyle]

4. Basic Concepts in Computer Information Processing

An introductory survey of computer information processing concepts and terminology, intended especially for those thinking of acquiring a small computer or word processing equipment. The needs of UI administrative, clerical and nonacademic personnel have been central to the planning of the course. Prerequisites: none.

June 27,28,29,30	3pm-4pm	269 Elect Engr	[Szoke & others]

5. Introduction to the CYBER System

This course is intended for the first time CYBER system user. The emphasis is on time-sharing usage and the ICE text editor. Batch usage may also be discussed. Required manuals: 1 and 2 (above). Assumes: Orientation (above). Four classes will be offered.

June 13,14,15,16,17	12N-1pm	241 Elect Engr	[Scheid]
June 20,21,22,23,24	12N-1pm	241 Elect Engr	[Edwards-Iwe]
June 27,28,29,30,July 1	8am-9am	168 Elect Engr	[Primer]
July 11,12,13,14,15	3pm-4pm	168 Elect Engr	[Scheid]

6. Introduction to Statistical Package Computing

The purpose of this course is to equip the novice computer user or the newcomer to this campus with the basic knowledge necessary to utilize CSO's computing resources to perform statistical analyses. Topics covered: an overview of the computer system and the statistical packages available, file concepts, program and data entry using the ICE text editor, and a glimpse of the four major statistical packages available: SPSS, SAS, SOUPAC, BMDP. (Note: this course is NOT intended to replace the regular introductory courses for each package.) Two 2-hour lectures followed by a 2-hour lab session covering basic file concepts and time-sharing terminal usage. Prerequisite: Orientation to CSO (above). This course will be offered twice in the Summer Session.

June 27,29	7pm-9pm	168 Elect Engr	[Mills]
Lab: Time and place to be arranged			
July 5,6	7pm-9pm	168 Elect Engr	[Alster]
Lab: July 7, 7pm-9pm; place to be arranged			

7. Introduction to RNF

A beginning level discussion of the RNF text formatter, used in the preparation of documents such as papers, theses and manuals. Topics will include tabbing, margins, spacing, paragraphing, and justification. Familiarity with the CYBER system and ICE is assumed. Required manual: 4 (above). Two sections are offered:

June 20,22,24	11am-12N	168 Elect Engr	[DeWan]
July 11,13,15	12N-1pm	168 Elect Engr	[DeWan]

8. Intermediate ICE (Illinois Central Editor)

A course on the use of the ICE text editor, covering techniques for solving a variety of editing problems: reading from and writing to local files, copy and transfer of text lines, extended modify, case conversion, tab fields, windows, loops, stripe, procedures, and others. Assumes: Introduction to CYBER or Introduction to Statistical Packages, or the equivalent. Required manual: 3, above.

July 25,27,29	12N-1pm	168 Elect Engr	[DeWan]

9. Use of MANAGE

How an instructor or project director allocates computer resources to individual users. Assumes familiarity with simple use of the CYBER.

June 21	12N-2pm	168 Elect Engr	[Primer]

10. Using Magnetic Tapes

Covers the use of magnetic tapes on the CYBER. Assumes familiarity with use of the CYBER. Registrants should obtain the free manual "Magnetic Tape Guide for the CYBER" at 1208 West Springfield or 85 Commerce West before attending.

July 18,19,20,21	3pm-4pm	168 Elect Engr	[Wetzel]

11. Introduction to Data Communications

This class will discuss: connecting a terminal to a computer both locally and remotely; long distance communication media; EIA RS232 standard interface wiring; modems; multiplexing; and local area networks. The discussion will be general and not necessarily deal with any particular hardware. Assumes: Basic Concepts (above) or equivalent knowledge.

June 20,21,22,23	3pm-4pm	168 Elect Engr	[Lathrop & Krol]

12. Easy Graphing

Elementary presentation of a high-level interactive plotting program for X-Y plots, bar and pie charts. Its English-like commands require no programming experience to generate plots. Assumes: Introduction to the CYBER or equivalent knowledge.

July 18,19,20	3pm-4pm	169 Elect Engr	[Tuchman]

13. Introduction to the DI-3000 Graphics Package

DI-3000 is a library of FORTRAN callable subroutines for computer graphics. It is an implementation of the 1979 ACM/SIGGRAPH CORE graphics system proposed standard. This course is for those with little or no experience using DI-3000. Covers capabilities, structure, and basic use of the package. No high level or X-Y plotting capabilities will be presented. Assumes a working knowledge of CYBER FORTRAN.

July 5,6,7	3pm-4pm	168 Elect Engr	[Tuchman]

14. Survey of Mathematical Software

This course provides a brief summary of the mathematical (non-statistical) software available at CSO. The lecture will cover subroutine libraries and software for linear systems, eigenanalysis, differential equations, data fitting, and other uses. Assumes a working knowledge of CYBER FORTRAN.

June 16	12N-1pm	168 Elect Engr	[Kerr]

15. Using the PROTRAN Package

This course describes a new package from IMSL, Inc., which can greatly ease the burden of mathematical problem solving by permitting one to input a simple, English-like description of a problem. Assumes: Intro. to CYBER or equivalent.

July 5,7	3pm-4pm	169 Elect Engr	[Kerr]
July 19,21	1pm-2pm	168 Elect Engr	[Kerr]

16. CYBER FORTRAN-77 (FTN5) for the FTN User

A systematic presentation of the principal differences between CYBER FORTRAN 1V (FTN) and the more recent FORTRAN-77 standard, as implemented by the CYBER FTN5 compiler. (This new compiler will become the default at some time in the future.) Assumes a good working knowledge of FTN.

June 27,28,29,30,July 1	1pm-2pm	168 Elect Engr	[Hirchert]

17. CYBER Control Language (CCL)

CCL can be used in batch jobs or procedures to control the flow of execution, alter the control statements to be executed, issue messages to the terminal, and generate text files which can be used as input to programs within a batch job or CCL procedure. This course will cover most CCL statements, CCL procedures, and, as time permits, related commands and examples which illustrate the various aspects of CCL. Assumes: Introduction to CYBER or equivalent.

July 5,7	3pm-4pm	163 Elect Engr	[Pommert]

18. Introduction to SPSS (Statistical Package for the Social Sciences)

This is an introductory course on the SPSS statistical package on the CYBER system. Topics covered: data preparation, basic components of the SPSS language, data modification, running SPSS programs at terminals, and using SPSS to obtain basic statistics and contingency tables. The course, consisting of two 3-hour lectures and one 2-hour lab (to be arranged), will be offered once in the Summer Session. Prerequisite: Intro to Stat Package Computing or equivalent knowledge.

July 11,12	6:30pm-9:30pm	241 Comm West	[Richardson]

19. Introduction to SAS (Statistical Analysis System)

An introductory course on the basics of SAS processing on the IBM 4341. The core content of the course is a series of six videotapes produced by SAS Institute covering the following topics:

1. Introduction to SAS
2. Getting Data Into SAS Data Sets -- the Data Step
3. Program Processing in the Data Step
4. Working With SAS Data Sets
5. Basic SAS Procedures
6. Report Writing

Lectures are included to relate how the material presented can be used in the local CYBER-IBM computing environment. The SAS videotape material is relevant to either OS/SAS or CMS/SAS usage. However, the lectures discuss only OS/SAS batch processing via the CYBER or CMS time-sharing systems. (A separate course will discuss local CMS/SAS usage.) The course consists of four 2.5 hour sessions. Prerequisite: Introduction to Statistical Package Computing (above) or equivalent knowledge of local system usage. Recommended manual: SAS Video Training Workbook (available at the CSO Distribution Office, 1208 W. Springfield, $8.00.)

June 27,30,July 7,11 6:30pm-9pm 115 DCL [Kesner]

20. Introduction to IBM Time-sharing (CMS and XEDIT)

This course introduces CMS (Conversational Monitor System), the IBM time-sharing system, and the XEDIT text editor for CMS. Special emphasis is placed on using CMS and XEDIT to run SAS and SAS/GRAPH programs, though the content is general enough to help any user begin working with the IBM time-sharing system. (All SAS users are encouraged to make the transition from CYBER time-sharing to CMS to perform their SAS processing because of the advantages CMS offers for such work.) Topics include: general CMS and virtual machine concepts, useful standard and local CMS and XEDIT commands, sending files between the CYBER and CMS, suggestions for utilizing the IBM CMS documentation available, and how to use the full-screen simulator under CMS.

This course is a prerequisite for the "Introduction to CMS/SAS and CMS/SAS/GRAPH" course. The course consists of two 2.5-hour lectures and one 2-hour lab.

June 14,16 6:30pm-9pm 115 DCL [Kesner]
Lab: June 18, 9am-11am; place, 85 Comm West

21. Introduction to CMS/SAS and CMS/SAS/GRAPH

An introduction to SAS and SAS/GRAPH processing using the IBM CMS time-sharing system. Topics include: using SAS interactively and non-interactively, creating and maintaining SAS data libraries under CMS, submitting batch OS/SAS jobs from CMS, using SAS/GRAPH under CMS, and obtaining hard-copy ZETA plots from CMS. Prerequisites: "Introduction to IBM Time-sharing" and "Introduction to SAS" or equivalent knowledge.

June 20,21 6:30pm-9pm 168 Elect Engr [Kesner]

22. Introduction to BMDP (Biomedical Statistical Programs)

BMDP, a widely-used package of statistical programs distributed by UCLA's Department of Biomathematics, is oriented toward the biological, medical, nutritional, agricultural and veterinary sciences. This is an introductory course on the use of BMDP on the CYBER system. Topics covered: data preparation, elements of the BMDP language, running BMDP programs at terminals, data editing, creation and use of system files, and examples of using BMDP for descriptive statistics, regression and analysis of variance. The course, consisting of two 2-hour lectures and a lab session (to be arranged), will be offered once in the Summer Session. Prerequisite: Intro to Stat Package Computing or equivalent knowledge.

July 26,27	7pm-9pm	140 Comm West	[Szoke]

23. Introduction to MANOVA (Multivariate Analysis of Variance)

Basic steps in the coding of multivariate analysis of variance designs using the general linear model, with examples from the SPSS MANOVA and SAS GLM procedures. Emphasizes elementary aspects of factorial, repeated measures and mixed designs. Missing values, special error terms and contrasts are covered as time permits. Detailed theory is not presented; some familiarity with the basics of MANOVA and one of the packages is assumed.

July 11,12,13,14	12N-1pm	160 Comm West	[Mills]

FREE STUDENT ACCOUNTS

Beginning with the 1983 Fall Semester, students will be able to apply for a **Free Student Account** number at the following locations:

CSO South Operations Room 70 Commerce West	or	CSO Accounting Office 1208 W. Springfield, Urbana

The Free Student Account is available to students currently enrolled at the University of Illinois. Each student is required to present a valid University I.D. to obtain the student account.

Current restrictions on obtaining a Free Student Account are as follows:

- Persons having an active SARA account (graduate students in the process of a Masters or Doctoral Thesis) are not eligible.
- Students currently enrolled in 100 level CS courses are not eligible.

SOFTWARE REGISTRY

CSO is trying to establish a registry of privately acquired or developed software on its machines as a means of encouraging the sharing of such software. If you have software that you think might be of use to others, please tell us

- its name (including version number if appropriate)

- a brief description of its function
- its availability (which machine(s) it runs on, whether there are legal restrictions on who may use it)
- the name and address of someone to contact for more information

You can send this information to Kurt Hirchert by campus mail (129 DCL) or the CYBER message system (TELL,HIRCHERT). If you have any questions about the registry, call Kurt at 333-8627.

TAPE SERVICE AT CSO

Magnetic tapes to be used on the CYBER or IBM must be checked-in to the machine room and assigned a label. This function is performed in Room 123 DCL. There is a desk immediately to your left upon entering Room 123; on the desk is a switch box and a logbook. When you turn the switch box "on" an operator will be prompted to assist you. While you're waiting for a person to respond, complete a new line entry in the logbook for each of your tapes. Tapes can also be checked-out of the machine room or cleaned in Room 123.

Room 123 is open only between the hours of 8:00 AM and 5:00 PM, Monday through Friday. We believe that these hours should be sufficient to accommodate almost everyone's tape handling needs. Also, these hours enable us to staff this service with full-time CSO employees, which provides us an opportunity to expand the functions of this service. If you have any questions or comments about this service, please feel free to stop by Room 127 DCL (333-8640) and talk with Greg German or Debbie Hudson.

STATISTICAL SERVICES

SAS SAMPLE LIBRARY AVAILABLE

We have now installed the SAS82 Sample Library on the IBM. This is a collection of SAS programs you may run which illustrate a variety of SAS programming applications. These examples are useful for learning how SAS is used to solve various computing problems. The sample programs may also save you some programming work if you find some of them useful in your particular application.

To access the sample library, you must use the %INCLUDE statement in a SAS job. The general format of the statement is:

```
%INCLUDE SAMPLES (member-name);
```

where SAMPLES is the ddname (JCL data-definition statement name) of the DD-statement (JCL data-definition statement) which makes the sample library available to your SAS job. This DD-statement is already included in the SAS IBM catalogued procedure which you invoke by the // EXEC SAS JCL statement in your SAS job.

(A catalogued procedure, like the one for SAS, is a collection of essential JCL statements which we have already stored on disk as a convenience for you. All you need do is call the catalogued procedure

by using the // EXEC statement, as you do to execute a SAS job. If you wish to see the contents of a catalogued procedure you are invoking, add the following ID card to your job: /*ID MSGLEVEL=(1,1). The contents will be listed at the beginning of your output.)

The *member-name* parameter of the %INCLUDE statement indicates the sample library program member you wish to execute. You may run the the following SAS job to obtain an index of all the members of the sample library:

```
//SAMPINDX JOB
/*ID PS=
/*ID CODE=
/*ID PRINT=CYBER,NAME='SAMPOUT(??????????)',EJECT=YES,OVFL=YES
/*ID LINES=3000,REGION=300K,TIME=(,15),IOREQ=1000
//*
// EXEC SAS
//*****************************************************************************
//*  JOB TO GENERATE INDEX FOR MEMBERS OF SAS SAMPLE LIBRARY FOR
//*  THE SAS BASE PRODUCT.
//*****************************************************************************
//SYSIN DD *
   %INCLUDE SAMPLES(INDEX);
```

NOTE: *You must modify the ID cards for the job to run successfully. Include your PS number and CODEWORD. Substitute your University ID for the question marks to send the output to the FETCH queue.*

(A copy of this SAS code is available in a CYBER file which you may obtain by entering: GET,SAMPSAS/UN=STATCON. To obtain a copy of the output listing from this job, enter: GET,SAMPOUT/UN=STATCON.)

From the index listing you may find sample programs you would like to run. Just replace the member-name INDEX in the %INCLUDE statement with the member-name of the program you want.

For more information about the SAS Sample Library, please see Vol. VIII, Num. 2 (Fall 1982) of "SAS Communications" which is available in the CSO South Consulting Office. For further details on the %INCLUDE statement, please see pages 316-317 of the *SAS User's Guide: Basics.*

OS/SAS82.2 IS NOW THE DEFAULT SAS PROCESSOR

On Wednesday, May 18th, OS/SAS82.2 replaced OS/SAS79.6 as the default SAS processor version. This means that when you use:

```
// EXEC SAS
```

to run a SAS job, you will be using OS/SAS82.2 rather than OS/SAS79.6. (Prior to this, OS/SAS82.2 could only be invoked by coding: // EXEC SAS822)

Please note that since the SAS supervisor is 45K larger in SAS82.2 than it was in SAS79.6, you will need to increase the region request on the REGION= ID card parameter to at least 45K more than what you have been specifying for SAS jobs run with SAS79.6.

To use the SAS79.6 processor, use the following EXEC statement:

```
// EXEC SAS796
```

REGION AND TIME CONSIDERATIONS FOR USING PROC GLM

Since questions about how to estimate the region and time needed for analyzing models using PROC GLM occur often, SAS Institute has attempted to address these considerations in the new *SAS User's Guide: Statistics*. On pages 157-158 of the chapter on PROC GLM, the major factors affecting the region size and time requirements are discussed. Users are encouraged to refer to this discussion when planning their analyses and specifically when designing their MODEL statement. PROC GLM is a powerful procedure for analyzing various linear models, particularly unbalanced designs. However the power of PROC GLM is not without cost; it is more expensive to use than PROC ANOVA. Therefore use PROC GLM if your design will not allow you to use PROC ANOVA or PROC REG for analysis. If you must use PROC GLM, weigh carefully the considerations discussed in the PROC GLM reference as you construct your MODEL statement. You may also want to refer to the Reference Guide, RF-11.2 IBM Rates/Charge Estimator, to calculate approximately how much your job will cost given your resources estimate. This may help avoid some unpleasant surprises when you receive your output.

SAS USERS -- PLEASE TELL US WHO YOU ARE!

We are continuing to compile a list of local SAS users which we will use to notify SAS users of changes or enhancements to SAS, SAS training seminars, and new information we receive from SAS Institute. If you have not already contacted us and would like to be added to the list, please specify:

University ID#,
Office Telephone Number,
Name,
Staff Position/Title
Campus address

in a message to Greg Kesner via the CYBER Message Facility by entering:

```
TELL,UN=KESNER
```

or by mailing the information to:

Greg Kesner
150 DCL (CSO)
Campus

MATHEMATICAL SERVICES

CONVERSION OF SOME CYBER PACKAGES TO FTN5

Since FTN5 has become CDC's "official" FORTRAN compiler and support has been dropped on FTN, CSO has been encouraging users to develop new software using FTN5 rather than FTN. We still have a large number of software packages which have been generated using FTN, and would like at some point to convert these to FTN5. This cannot be done without warning, however, for the following reasons:

1. If a subroutine library (e.g., IMSL) has been compiled with FTN, it can be used with an FTN5 main program, with no problems in most cases. (There are problems in a few areas, typically related to infrequently used input/output features such as the "=" format specification.) The reverse is not true, though: if a library is compiled with FTN5, an FTN main program may have difficulties using it; the difficulties come from library routines which do input/output. These difficulties can be alleviated in some cases by using the system command ADDLIB,FTN5LIB before running one's program, but this does not cover all cases.

   Subroutine libraries in question here include: IMSL, UOILIB, LINPACK, EISPACK, MINPACK, LSODE, FISHPAK, AMOSLIB, XMP.

2. Some packages (e.g., SLAM) involve a main program and a number of subroutines. In some instances, a user may supply replacements for particular routines in the package, and in other instances, the user may actually supply their own version of the main program. This latter case is difficult to convert: if, say, SLAM is compiled with FTN5, and a user supplies a main program compiled with FTN, the program fails completely; it will not run at all. The main program MUST be compiled with FTN5 in this case. (SLAM is not alone in this regard. Other packages in the same boat are GRG and GASP.)

It is thus not possible to have a "smooth", "invisible" transition. Although it will require a large amount of disk space to maintain parallel FTN and FTN5 versions of these packages, we feel it is necessary for the smoothest transition. Thus, for some time prior to the conversion, there will be FTN5 versions of many packages available via the GRAB/F and FUTURE commands.

We are setting a target date of January 9, 1984 (a Monday) for the transition to default FTN5 versions of our GRAB products which are built with FTN. Past this date, FTN versions of the various packages will not be available on a casual basis; it will be possible to access the old FTN versions, but it will be relatively inconvenient and done on a case-by-case basis.

CSO will begin to announce FTN5 versions of some packages very soon. The next issue of *OFF-LINE* should see the first announcements of FTN5 versions to be accessed via GRAB/F.

Some packages with limited markets can be converted sooner than others. FORSIM, for instance, has a limited community of users, so a full semesters notice will not be necessary.

Please watch *OFF-LINE* in the coming months for notices about packages being converted. We will run a notice about FTN5 conversion in each issue throughout the Fall semester.

SOFTWARE FROM NATIONAL BUREAU OF STANDARDS

Recently Professor Gear of the Department of Computer Science received a magnetic tape from the National Bureau of Standards with a number of items of software in the public domain. He has kindly allowed CSO to have a copy of this tape. There are 88 files on the tape, all with fixed length records of length 80, and 20 records per tape block. The tape is 9-track, unlabeled, and is recorded in the ASCII character code. Half of the files are documents describing the software in the remaining files; the document files are in upper and lower case, so the COPYCH utility is needed on the CYBER to read the tape.

A file named NBS in UN=MATHLIB on the CYBER systems contains a directory of the tape and instructions for reading individual files on it. Over the course of time, we expect that some of the packages on this tape will be integrated into our regular offerings of software (some already are, e.g. LINPACK, EISPACK); for the present, we are making the tape available to interested parties on a use-at-your-own-risk basis. If you desire assistance in obtaining particular items of software from the tape, please talk to Stan Kerr (179 DCL, phone 333-4715, or TELL,UN=MATHLIB on the CYBER systems).

Following is a list of the 88 files on the tape, and a short description of the software in each file. After this list is an alphabetical listing of the packages by name, with an index to the files containing them.

NOTE: An asterisk (*) in the left margin indicates that the item is already on CSO systems in some form.

file	*record count*	*source/ doc*	*contents*
1	106	doc	description of the tape
* 2	2205	source	Basic Linear Algebra Subroutines (part of LINPACK on our CYBERs. also in IMSL.)
3	787	doc	documentation on BLAS in file 2
4	2063	source	FFTPKG--a package of fast fourier transform routines
5	954	doc	documentation on file 4
6	31	doc	documentation on RKF45 routine
* 7	146	source	ZEROIN--a routine to solve a nonlinear equation in one variable
8	113	doc	documentation on file 7
* 9	3659	source	Yale Sparse Matrix Package (this is contained in UOILIB on the CYBERs, under names "YALEx" for various x.)
10	801	doc	documentation on file 9
11	13466	source	FNLIB--portable special function library
12	462	doc	documentation on file 11
*13	16115	source	FISHPAK--routines for various types of separable elliptic partial differential equations. Available on CYBERs under the name FISHPAK.
14	6014	doc	documentation on file 13
15	4685	source	FC--routines for constrained linear least squares
16	1856	doc	documentation on file 15
17	705	source	XERROR--error handling routines used in CMLIB
18	86	doc	documentation on file 17
19	804	source	SSORT--fast in-core sorting of arrays
20	194	doc	documentation on file 19
21	2539	source	SDRIV--stiff differential equation solver
22	1124	doc	documentation on file 21
23	2534	source	DDRIV--double precision version of SDRIV (file 21)

24	1119	doc	documentation on file 23
*25	6870	source	LINPACK -- single precision routines These are available on our CYBERs under the name LINPACK (which comprehends the single precision, double precision and complex routines, plus the basic liner algebra routines -- BLAS -- referred to earlier.)
26	2951	doc	documentation on file 25
*27	6868	source	LINPACK -- double precision routines
28	2993	doc	documentation on file 27
*29	8772	source	LINPACK -- complex routines
30	3513	doc	documentation on file 29
31	940	source	routines to return machine constants. They include comment statements which specify constants for a wide variety of machines.
32	149	doc	documentation on file 31
33	268	source	XBLAS--extended BLAS--functions not in regular BLAS
34	123	doc	documentation on file 33
35	752	source	LICEPACK--interface to EISPACK package--includes drivers which make it easier to use EISPACK
36	447	doc	documentation on file 35
*37	11472	source	EISPACK eigenanalysis package (available on the CYBERs under the name EISPACK)
38	4389	doc	documentation on file 37
39	411	source	X-SDRIV
40	411	source	X-DDRIV
*41	2168	source	BSPLINE routines from Carl de Boor's book "A Practical Guide to Splines" (These are available via "SHARED,BSPLINE" on our CYBERs; source is available through the MATH procedure.)
42	1059	doc	documentation on file 41
*43	2205	source	BSPLINE routines -- double precision
44	1088	doc	documentation on file 43
*45	2296	source	AMOSLIB special function library (This is available on the CYBERs under the name AMOSLIB.)
46	379	doc	documentation on file 45
47	5066	source	LINDRIVES--an easy to use interface to LINPACK for solving linear systems of equations
48	3323	doc	documentation on file 47
*49	473	source	SLVBLK--a routine for solving linear systems of equations which are "almost block diagonal".
50	122	doc	documentation on file 49
51	621	source	SLRPACK--simple linear regression package
52	1075	doc	documentation on file 51
53	167	source	SQRLSS--solves linear least squares problem in the matrix form AX=B, using LINPACK. Easy to use.
54	161	doc	documentation on file 53
55	4960	source	SNLSE--solves nonlinear least squares problems and nonlinear systems of equations.
56	1952	doc	documentation on file 55
57	2571	source	CDRIV--complex version of SDRIV.
58	1107	doc	documentation on file 57
59	9124	source	QUADSP--routines for definite integrals of functions of one variable; including singular integrands and infinite intervals.
60	4970	doc	documentation on file 59
61	8881	source	QUADDP--double precision version of QUADSP
62	4507	doc	documentation on file 61

63	194	source	CPQR79
64	135	doc	documentation on file 63
65	251	source	CPZERO
66	131	doc	documentation on file 65
67	3708	source	PCHIP--routines for 1-dimensional interpolation; specially adapted to produce nice-looking fits.
68	1206	doc	documentation on file 67
69	733	source	LOTPS--2 dimensional interpolation of scattered data using thin-plate splines.
70	498	doc	documentation on file 69
71	1056	source	TENSORBS--2 and 3-dimensional interpolation using tensor products of 1-dimensional splines.
72	599	doc	documentation on file 71
73	1064	source	DTENSORBS--double precision version of TENSORBS
74	604	doc	documentation on file 73
75	2052	source	BVSUP--routine for 2-point boundary value problems in ordinary differential equations
76	437	doc	documentation on file 75
77	5394	source	DEPAC--a set of routines for solving initial value problems in ordinary differential equations.
78	1859	doc	documentation on file 77
79	1233	source	SUDSSODS--solves over and under-determined systems of linear algebraic equations.
80	204	doc	documentation on file 79
81	1792	source	FCNPAK--a collection of special function routines, including the associated Legendre functions (ferrers functions) and the normalized Legendre polynomials.
82	751	doc	documentation on file 81
83	5243	source	SPLP--solves linear programming problems.
84	1549	doc	documentation on file 83
85	973	source	Q1DA--1 dimensional integration
86	301	doc	documentation on file 85
87	103	source	UNI--generates uniform random numbers
88	78	doc	documentation on file 87

Following is an alphabetical list of the various packages on the tape. The number to the left of each entry is the file number in which the FORTRAN source of the package can be found. For most packages, the immediately following file contains the documentation on the package.

file	*package*	*description*
45	AMOSLIB	a collection of special function routines with particular emphasis on the special functions of statistics.
2	BLAS	basic linear algebra subroutines. perform various elementary matrix and vector operations.
41	BSPLINE	subroutines for computing with piecewise polynomials (b-splines). includes interpolation, differentiation and integration with b-splines.
75	BVSUP	solves linear two-point boundary value problems using superposition coupled with an orthonormalization procedure and a variable-step integration scheme.
57	CDRIV	solves initial value problems for systems of ordinary differential equations including stiff equations.

complex diff. eqs., real independent variable.

43 DBSPLINE .....subroutines for computing with piecewise polynomials (b-splines). double precision version of BSPLINE package.

77 DEPAC.....a suite of programs for solving initial value problems for ordinary differential equations.

23 DDRIV.....solves initial value problems for systems of ordinary differential equations, including stiff systems. (double precision version of SDRIV).

73 DTENSORBS.....interpolation of two and three dimensional gridded data using tensor products of one dimensional b-spline basis functions. double precision version of TENSORBS.

37 EISPACK.....solves various linear algebraic eigenvalue problems.

15 FC.....solves constrained least squares problems.

81 FCNPAK.....a collection of special function routines, including the associated Legendre functions (ferrers functions) and the normalized Legendre polynomials.

4 FFTPKG.....subroutines for computing the fast fourier transform in various forms.

11 FNLIB.....portable special function routines (e.g., bessel functions, the error function, etc.)

13 FSHPK.....FISHPAK solves separable elliptic boundary value problems in two and three dimensions using a variety of coordinate systems.

35 LICEPACK.....solves various linear algebraic eigenvalue problems. (provides an interface to the EISPACK package).

47 LINDRIVES.....programs to solve linear systems of algebraic equations in a number of forms. Provides an easy to use interface to the LINPACK package.

29 LINPACKC.....analyze and solve various systems of linear algebraic equations. (complex precision version of LINPACK).

27 LINPACKD.....analyze and solve various systems of linear algebraic equations. (double precision version of LINPACK).

25 LINPACKS.....analyze and solve various systems of linear algebraic equations. (single precision version of LINPACK).

69 LOTPS.....a set of programs for smooth interpolation of scattered data in two dimensions using thin-plate splines.

31 MACHCONST....functions that return machine-dependent constants.

67 PCHIP .....is a set of programs for interpolation of univariate data and which is specially adapted to producing fits which are are aesthetically pleasing.

85 Q1DA .....evaluates one dimensional integrals automatically, easy to use but very powerful.

61 QUADDP.....is a set of programs for evaluating definite integrals of functions of one variable; the double precision version of QUADSP.

59 QUADSP.....is a set of programs for evaluating definite integrals of functions of one variable; including singular integrands and infinite intervals .

21 SDRIV.....solves initial value problems for systems of ordinary differential equations, including stiff systems.

51 SLRPACK .....a collection of FORTRAN subprograms for simple linear regression

49 SLVBLK.....solves linear systems of algebraic equations where the coefficient matrix is in "almost block diagonal" form

55	SNLSE	solves non-linear least squares problems and non-linear systems of equations.
83	SPLP	solves linear programming problems (minimize a linear function of n variables subject to linear equality constraints).
53	SQRLSS	solves linear least squares problem in the matrix form AX=B. Uses LINPACK routines. Easy to use.
19	SSORT	fast in-core sorting of arrays.
79	SUDSSODS	solves overdetermined and underdetermined systems of linear algebraic equations. The least squares solution is provided for problems with full rank.
71	TENSORBS	interpolation of two and three dimensional gridded data using tensor products of one dimensional b-spline basis functions.
87	UNI	generates uniform random numbers. Portable, reproducible, and with a long cycle.
33	XBLAS	extended basic linear algebra subroutines. Perform various matrix and vector operations not found in the BLAS.
17	XERROR	error handling utilities.
9	YSMP	The Yale Sparse Matrix package. Solves large sparse systems of linear algebraic equations in core with no pivoting.
7	ZEROIN	finds zeros of a function of one variable.

MATH NOTES

We would like to remind users at the outset of the summer semester that there is an informal set of notes on various topics of mathematical software on the CYBER systems. These notes are accessed by means of a CCL procedure called MNOTES, which is accessed by the command:

```
GET,MNOTES/UN=MATHLIB.
```

The available notes are numbered; to retrieve note *nnn*, you would enter the following command:

```
MNOTES,nnn.
```

This would produce a file named OUT which you could print as follows:

```
PRINT,OUT/AS/CC/EJ/RJE=xxx.
```

where *xxx* designates some valid print site on campus. The currently available note numbers are as follows:

Number	***Contents***
0	General information about MNOTES
1	Brief summary of math software available on the CYBERs
2	Description of user number MATHLIB
3	Curve fitting software
4	Summary of math libraries with descriptions
5	Description of IMSL products and services

6	Description of "TOMS" (Transactions on Math Software) and a summary of algorithms published to date
7	Simulation software
8	A summary of possible sources of software
9	Routines for ordinary differential equations
10	(unused)
11	(unused)
12	Linear programming software
13	Software for sparse linear problems
14	Software for partial differential equations
15	Combined library catalogs -- about 100 pages long !

TRANSACTIONS ON MATHEMATICAL SOFTWARE

We now have on tape all algorithms published in *Transactions on Mathematical Software* up to March 1983 (from algorithm 493 published in the first issue of March 1975, up to algorithm 595).

All TOMS algorithms are kept on a public tape; file TOMS/UN=MATHLIB contains a directory of this tape, plus directions for reading particular files on it. To print a copy of this directory, enter the following CYBER commands:

```
GET,TOMS/UN=MATHLIB.
PRINT,TOMS.
```

The "Math Note" series described in the March 1981 issue of *OFF-LINE* includes a note summarizing all algorithms published in TOMS to date. To print this note (it is note number 6), enter the following:

```
GET,MNOTES/UN=MATHLIB.
MNOTES,6.
PRINT,OUT/AS/CC/EJ.
```

(To print instructions on the use of MNOTES, substitute a 0 (zero) for the 6 above.)

The algorithms published in the December 1982 and March 1983 issues are as follows :

In the December 1982 issue:

588:	Fast Hankel Transforms Using Related and Lagged Convolutions by Walter L. Anderson
589:	SICEDR: A FORTRAN Subroutine for Improving the Accuracy of Computed Matrix Eigenvalues by Jack J. Dongarra
590:	DSUBSP and EXCHQZ: FORTRAN Subroutines for Computing Deflating Subspaces with Specified Spectrum by P. Van Dooren
591:	A Comprehensive Matrix-Free Algorithm for Analysis of Variance by William J. Hemmerie
535:	remark on algorithm 535. The QZ algorithm to solve the generalized eigenvalue problem
539:	remark on algorithm 539. Basic Linear Algebra Subprograms for FORTRAN Usage

580: remark on algorithm 580. QRUP: A Set of FORTRAN Routines for Updating QR Factorizations

In the March 1983 issue:

592: A FORTRAN Subroutine for Computing the Optimal Estimate of f(x)
by P. W. Gaffney

593: A Package for the Helmholtz Equation in Nonrectangular Planar Regions
by Wlodzimierz Proskurowski

594: Software for Relative Error Analysis
by John L. Larson, Mary E. Pasternak, and John E. Wisniewski

595: An Enumerative Algorithm for Finding Hamiltonian Circuits in a Directed Graph
by Silvano Martello

573: Remark on algorithm 573: NL2SOL -- an adaptive nonlinear least squares algorithm
by David M. Gay

The following articles which appeared in the December and March issues may be of interest to some of our users:

In the December 1982 issue

Checking the Calculation of Gradients
by Phillip Wolfe

Fast Hankel Transforms Using Related and Lagged Convolutions
by Walter L. Anderson
(This is the support paper for algorithm 588.)

In the March 1983 issue:

A Functional Description of ANALYZE: A Computer-Assisted Analysis System for Linear Programming Models
by Harvey S. Greenberg

A Reduced Gradient Algorithm for Nonlinear Network Problems
by P. Beck, L. Lasdon, and M. Engquist

If you are looking for software, *Transactions on Mathematical Software* is one of the journals to check, along with *Numerische Mathematik, BIT, Computer Journal*, and various publications of ACM (the Association for Computing Machinery) and SIAM (the Society for Industrial and Applied Mathematics).

We have in the systems consulting office at 1208 W. Springfield bound volumes containing all the algorithms published in *Communications of the ACM* from 1960 to 1975 (when the algorithms department of CACM became Transactions on Mathematical Software).

IMSL NEWSLETTER / IMSL BOOK

We have received the March 1983 issue of IMSL User News, IMSL's periodic newsletter telling of new IMSL products or information about current products. A copy is on display in the CSO System

Consulting Office at 1208 W. Springfield. Additional copies are available from Stan Kerr, 179 DCL, 333-4715.

One particular item of interest is the announcement of LP/PROTRAN, a linear programming member of IMSL's PROTRAN family of problem- oriented applications packages. LP/PROTRAN will work similarly to MATH/PROTRAN and STAT/PROTRAN (both of which CSO now has; STAT/PROTRAN has arrived but not been installed yet). LP/PROTRAN will allow a linear programming problem to be described in two different formats, an algebraic format like that used in MPOS and LINDO, and an array format like that used in IMSL routine ZX4LP. LP/PROTRAN will be released this summer.

We have also received copies of a new book published by McGraw-Hill:

Numerical Methods, Software, and Analysis (IMSL Reference Edition)
by John R. Rice

This book describes all the basic mathematical techniques used by the non-statistical routines of the IMSL library, and includes extensive exercises. There are several appendices describing the IMSL library and the PROTRAN package in detail. List price of the book is about $40. If you are interested in seeing one of our copies, please contact Stan Kerr (333-4715) or Beth Richardson (333-6276).

THE PROTRAN PROBLEM-SOLVING LANGUAGE

CSO would like to encourage users to acquaint themselves with the PROTRAN problem-solving language from IMSL. Designed as a family of similar languages directed toward different problem areas, the PROTRAN family now includes MATH/PROTRAN and STAT/PROTRAN, both of which CSO has acquired, and will include LP/PROTRAN, a language for linear programming.

PROTRAN is available on the CYBER systems by entering the command:

```
GRAB,PROTRAN.
```

This places several files used for PROTRAN into your local file space. Running PROTRAN involves first translating it with the PROTRAN command, much as you use the FTN or FTN5 command to compile a FORTRAN program.

For example, to evaluate the integral of EXP(X**2) from X=0 to X=1, the following PROTRAN program might be used:

```
      PROGRAM MAIN(OUTPUT,TAPE6=OUTPUT)
$     INTEGRAL  EXP(X**2); FOR(X=0,1); IS XINT
$     PRINT XINT
$     END
```

If you place this in, say, file XYZ, then the following command causes PROTRAN to translate this program into FORTRAN (FTN5 FORTRAN, to be exact), and then compile it into an LGO file:

```
PROTRAN,XYZ.
```

Then, the command

```
LGO.
```

causes the program to run (just as for FORTRAN). The integral is evaluated and the value stored in the FORTRAN variable XINT; this is then printed by the $PRINT command in the program. You will see the output displayed on your terminal.

Documentation on PROTRAN is available in several forms:

1. A manual from IMSL can be purchased at our Distribution Office at 1208 W. Springfield fro $15.50. This includes a binder. Updates to the manual will cost extra; an update is expected within the next several months. We have no price schedule for the cost of manual updates presently.

2. A reference guide (RF-4.32) summarizes how to access PROTRAN, and includes a 1-page summary of all the math procedures in MATH/PROTRAN.

3. An on-line writeup summarizes PROTRAN and includes a number of examples of using the various math procedures. This is not intended to replace the full IMSL manual, only to supplement it. An experienced person may find this writeup sufficient to learn from .

This summer, two 2-hour courses on PROTRAN will be offered, at the following dates and times:

July 5 and 7, 3pm-4pm, 169 Electrical Engineering (Tuesday and Thursday)
July 19 and 21, 1pm-2pm, 168 Electrical Engineering (Tuesday and Thursday)

You can enroll in these courses the week before each is offered by signing on to the CYBER and entering the commands:

```
GET,SIGNUP/UN=COURSES.
SIGNUP.
```

This will run a program which will ask you what courses you wish to enroll in.

PLOTTING SERVICES

NEW RATES FOR PLOTTING

Over the past few years the cost of producing quality plot output has risen while the revenue, because it has been calculated in service units, has declined. Also, we are dissatisfied with the present charging formula because it does not always reflect the real costs of producing a plot. For example, we have learned that plots which take a long time to run often have to be restarted several times, because the probability of problems occurring (e.g., pens drying out, or CYBER service interrupted) is increased. Obviously, such restarts increase the cost of producing a plot, but this has not been accountable within the present charging formula. On the other hand, although we've charged more for India ink pen setups, we've found that the type of pen being used is not a significant factor in our cost of producing a quality plot. In fact, our costs for the type of plotting medium used and even for the type of plotter (large or small) are small compared with other costs of producing special plots.

For these reasons we will be using a new charge formula, beginning July 1, 1983, for all special plots:

$$\text{plot cost in dollars} = \max(23.0*t*(1.5 - 0.5/(1+t)),1)$$

where *t* is the plotter time in hours. There will be no charge for paper or pen setups; there will be no extra charge for use of the large plotter. Also note that each special plot job will be charged a minimum of $1.00.

People with short plots will generally find their costs reduced. Those with long plots will find their costs will increase.

The present methods of charging for special plots (hard money PS number, real-money University account number, or personal check) will remain in effect. See Reference Guide, RF-7.31 PLOT or PLOTZ, for more information about submitting plot jobs.

NETWORK SERVICES

SYTEK CAMPUS NETWORK UPDATE

In the January-February issue of *OFF-LINE*, there was a discussion of the Sytek LocalNet network. Since that time, the cost of purchasing a complete Tbox or Tmux has decreased. The equipment costs are now as follows:

- A two-port Tbox now costs $981, or approximately $500 per port.
- An eight-port Tmux now costs $3375, or approximately $400 per port.

All other costs remain the same. We repeat them briefly here for your convenience.

- People who purchase a Tbox or Tmux will pay $15/port/month or $150/port/year to be connected to the network. (People who rent or purchase a "receive-only" port will not pay this monthly fee.)
- People who rent a Tbox or Tmux, or a single port, will pay $400/port/year.
- Installation of the Tbox or Tmux will be $70.00 and each EIA line will cost $50.00. These costs are reasonable estimates based on time and materials.

Finally, anyone may purchase a dedicated line for $3000 which provides you with the following:

- installation
- maintenance for three years
- reduced connect rates to CSO facilities for three years
- no network access fee for three years.

Questions regarding the campus network may be referred to Scott Lathrop , 193 DCL, 333-6618.

FEATURE ARTICLES

FORTRAN FUTURES

This article is part of a series on the work of the FORTRAN Standards Committee X3J3 in producing the next revision of the FORTRAN standard. The reader is reminded that the features described in this article are not a part of any FORTRAN compiler currently available, but rather are proposed requirements for FORTRAN processors produced in the late 1980's through the mid 1990's. Although every effort has been made to accurately describe the current position of X3J3 on these matters, the development of a revision to the FORTRAN standard is an evolutionary process, and these proposals may be subject to refinement, revision, or even retraction. Comments on these proposals may be given to Kurt Hirchert of the CSO Systems Consulting staff, who is a member of X3J3.

The 86th meeting of X3J3 was held May 9-13, 1983 on the campus of the University of Illinois at Urbana-Champaign. Activity at that meeting included the following:

- As the committee prepares to wind up the phase of the work in which new technical features have been developed and begin a document writing phase, there has been increased attention on the form the new standards document will take, including some of the first formal proposals on this topic.

- A number of minor proposals were adopted to correct minor problems in previously passed proposals or otherwise facilitate the writing of the working document intended as the textual basis for the next revision of the standard.

- Several formal interpretations of the 1978 standard were officially adopted. Additional interpretations were unofficially scheduled for official adoption at the August meeting.

- Event/exception/error handling was again considered. There was an extensive discussion on the issue of what arithmetic exceptions should be defined. A framework for handling events was formally adopted amidst great controversy. This topic seems likely to remain the focus of extensive debate for several meetings to come.

- The previously passed one-dimensional array constants feature was extended to be a general one-dimensional array constructor. A SHAPE intrinsic function was added to provide a facility for converting to arrays of higher dimensionality. As a side effect of this work, the SEQ (sequence generating) intrinsic function was deleted, as this functionality was included in the array constructor functionality.

- The FORALL statement was adopted. The FORALL statement simplifies the writing of array computations involving the position of the elements within the array.

- A proposal was adopted which provided for a class of dynamic arrays which would be explicitly allocated and freed by the program. This proposal is especially significant because it is the first feature requiring dynamic storage that cannot be managed as a stack. The adoption of this proposal is likely to trigger the submission of several other proposals requiring dynamic storage managed on a heap.

- Traditionally, the nature of the interface to a Fortran procedure has been derivable from any reference to that procedure. With the addition of new features like optional arguments and array valued functions, this was no longer true. A proposal was adopted which, in effect, requires the use of a procedure interface block whenever one of these new features is to be used, thus insuring that the processor will know the interface rather than having to derive it from the form of the reference to the procedure. Earlier mechanisms for specifying some of

this interface information were deleted.

- Proposals were adopted which extend Fortran data structuring facilities more towards being tools for implementing data abstractions. Assignment or argument association between two data structures now requires that they be specified to have the same form, rather than simply structurally equivalent forms. Also, a mechanism was adopted to define the meaning of operators like "+" when applied to such data structures.

- "**" was chosen to be the format indicator for name-directed input/output.

- A proposal was rejected that would have placed explicit restrictions on non-Fortran procedures included in standard conforming executable programs. An informal vote suggested that a more clearly worded proposal on the same subject might be acceptable.

- A proposal identifying the concept of "consistent" function (i.e., one which acts like a mathematical function) was rejected.

- Alternative syntax for the naming of block constructs was considered. A formal proposal in this area is likely to be forthcoming at a future meeting.

- The character sequence "__" (two underscores) was tentatively accepted for use in references to fields of a structure as the separator between the structure name and the field name. Symbolic names will be restricted to not contain consecutive underscores (but will still be allowed to contain singleton underscores for readability). A formal proposal on this topic is expected at a future meeting.

- The interaction between array subscripting (especially subsectioning) and structure qualification was again considered with no resolution of the issues involved.

- A proposal to allow parameterized forms was presented. Initial committee response was favorable.

- The possibility of overloading procedure names, distinguishing procedures by the types of their arguments, was discussed.

- A proposed module supporting high accuracy matrix and array computations received lukewarm response.

- Several irregularities in Fortran 77 were discussed and possible changes in the next revision suggested.

- The possibility of giving some kind of status as standard conforming to a processor that supports the core features, but not the deprecated (Fortran 77) features, was rejected.

DOCUMENTATION

NEW TECHNICAL NOTE SERIES

As an added service to our users, CSO is planning to produce a series of "Technical Note" documents which will be available at all RJE sites. Each "Technical Note" will be a short document pertaining to a specific topic; it will be more detailed than a Reference Guide, but shorter and more topic specific than a manual.

To start off the "Technical Note" series, we have completed six documents about the plotting device drivers at CSO. These documents, numbered TN-100 through TN-105, should be available in the near future at the various sites:

TN-100	DI-3000 Device Driver Configuration File at the University of Illinois
TN-101	DI-3000 Tektronix 4010 Device Driver at the University of Illinois
TN-102	DI-3000 Zeta 1453 Device Driver at the University of Illinois
TN-103	DI-3000 Tektronix 4027 Device Driver at the University of Illinois
TN-104	DI-3000 Line Printer Device Driver at the University of Illinois
TN-105	DI-3000 Alphanumeric Terminal Device Driver at the University of Illinois

SALES - HELP WANTED

SKYMNK ARITHMETIC PROCESSOR FOR SALE

Produced by Sky Computers, Inc., this array processor for DEC systems brings a 30-300 times speed-up in LSI-11 performance.

Included is the MNKSIM software simulator and all documentation.

Condition: NEW
Contact: Carl Moore, 105 Noyes Lab, 333-1236

The manufacturer's list price is $7,052, though it has been offered for $4,800. You can have ours immediately for $3,500.

OFF-LINE's Mailing List

If you wish to be placed on our mailing list for future issues of *OFF-LINE*, if you wish to be removed from the list, or if you wish to enter an address correction, please complete and return this page. **(Current subscribers are kept on the mailing list until a specific request for removal is received, or until a mailing is returned as undeliverable.)**

Check one:

- ☐ Place my name on mailing list
- ☐ Make the following corrections or changes
- ☐ Delete my name from mailing list

_____

First name -- Initial -- Last Name

Campus Address:

_____

Room - Building

_____

Department

Off-campus Address:

_____

Organization or Company (if applicable)

_____

Street Address

_____

City -- State -- Zip Code

If address correction, give old address and zip code below.

_____

_____

_____

SEND TO:

OFF-LINE
150 Digital Computer Laboratory
University of Illinois at Urbana-Champaign
1304 West Springfield Avenue
Urbana, Illinois 61801

Computing Services Office

Off-Line

University of Illinois at Urbana-Champaign VOL. 11, NO. 7 July 1983

EDITOR: Lynn Bilger
PHONE: (217) 333-6236
120 Digital Computer Lab
1304 W. Springfield Ave.
Urbana, Illinois 61801

CSO DIRECTORY - STAFF AND SERVICES

Administrative

Director	George Badger	150	DCL	333-4103
Business Manager	Stanley Rankin	150	DCL	333-6530
Secretary	Joyce McCabe	150	DCL	333-1637

User Services and Hardware/Software Support

User Accounting		**1208**	**W Springfield**	**333-7752**
Distribution Office		**1208**	**W Springfield**	**333-6760**
Systems Consulting		**1208**	**W Springfield**	**333-6133**
Statistical Services Consulting		**85**	**Comm West**	**333-2170**
Text Processing Consulting		**118**	**DCL**	**333-7318**
Terminal Repair Service		**150**	**DCL**	**333-0969**
CYBER Dial-up Numbers	**CYBER 175**	**300**	**baud**	**333-4000**
	CYBER 174	**300**	**baud**	**333-4004**
	CYBER DIAL-UP	**1200**	**baud**	**333-4008**
	IBM 4341	**300**	**baud**	**333-4006**
Asst Dir User Services	Robert Penka	173	DCL	333-4709
Asst Dir Systems and Operations	Sandra Moy	177	DCL	333-4703
Asst Dir Development	J. M. Randal	175	DCL	333-9772
Asst Dir Engineering and Hardware Selection	Cliff Carter	195	DCL	333-3723
Manager Accounting Services	Gary Bouck	1208	W Springfield	333-7752
Manager Statistical Services	Beth Richardson	189	DCL	333-6276
Documentation	Lynn Bilger	120	DCL	333-6236
CYBER-IBM Operations	Jack Knott	194a	DCL	333-6562
UNIX Operations	Debbie Hudson	123	DCL	333-8150
Laboratory Support Project	Mike Gardner	18	DCL	333-7904
RJE Operations North	Rex Duzan	162	DCL	333-6285
RJE Operations South	Don McCabe	1208	W Springfield	333-2171 333-7752

RJE Sites (see Reference Guide RF-0.3 for operating hours)

Agriculture	M103	Turner Hall	333-8170
Chemistry	153	Noyes Lab	333-1728
Commerce West	70	Comm West	333-4500
CRH Snack Bar	120	Snack Bar	333-1851
DCL Routing Room	14	DCL	333-6203
Electrical Engineering	146	EEB	333-4936
Florida Ave Res Hall		FAR	333-2695
Illinois St Res Hall		ISR	333-0307
Mechanical Engineering	65	MEB	333-1430
Psychology	453	Psych Bldg.	333-7815
Social Science	202	Lincoln Hall	333-0309

OFF-LINE is the monthly newsletter of the Computing Services Office at the University of Illinois at Urbana-Champaign. Unless otherwise indicated, permission to reprint is freely granted, provided that the author, if named, and the Computing Services Office (CSO) are credited. Information in this issue is current as of June 22, 1983.

CSO operates a CDC CYBER 175 with 262K words of central memory and a CDC CYBER 174 with 196K words of central memory. The 175 and 174 run under the NOS Operating System and share 512K words of ECS. The 175 serves over 200 simultaneously active text and graphics terminals and the 174 serves over 100 simultaneously active terminals. CSO also operates an IBM 4341 Model 2M with 8 million bytes of memory running HASP-OS/MVT under VM and an IBM 4341 Model L10 with 4 million bytes of memory running VM. In addition, CSO operates a DEC VAX 11/780 with 4 million bytes of memory, running under the UNIX Operating System and driving a GSI CAT-8 phototypesetter.

Operating Hours (see HEARYE,SCHEDUL for exceptions):

	CYBERS 174/175	IBM
M-F	8 am to 6 am	8 am to 6 am
SAT	8 AM - Midnight	8 AM - 6 AM
SUN	Noon - 6 AM	Noon - 6 AM

POLICY

FREE STUDENT ACCOUNTS

Beginning with the 1983 Fall Semester, students will be able to apply for a **Free Student Account** number at the following locations:

CSO South Operations Room 70 Commerce West	or	CSO Accounting Office 1208 W. Springfield, Urbana

The Free Student Account is available to students currently enrolled at the University of Illinois. Each student is required to present a valid University I.D. to obtain the student account.

Current restrictions on obtaining a Free Student Account are as follows:

- Persons having an active SARA account (graduate students in the process of a Masters or Doctoral Thesis) are not eligible.
- Students currently enrolled in 100 level CS courses are not eligible.

SOFTWARE REGISTRY

CSO is trying to establish a registry of privately acquired or developed software on its machines as a means of encouraging the sharing of such software. If you have software that you think might be of use to others, please tell us

- its name (including version number if appropriate)
- a brief description of its function
- its availability (which machine(s) it runs on, whether there are legal restrictions on who may use it)
- the name and address of someone to contact for more information

You can send this information to Kurt Hirchert by campus mail (129 DCL) or the CYBER message system (TELL,HIRCHERT). If you have any questions about the registry, call Kurt at 333-8627.

CYBER

UWRIM INSTALLED

The University of Washington Relational Information Management system (UWRIM) was recently installed on the Cyber. UWRIM is a database management system based on the relational algebra model for data management. The original version was developed under a NASA contract at the BOEING company. It has since undergone considerable enhancement at both BOEING and the University of Washington.

UWRIM is an easy to learn and easy to use database package. It can be used to solve a wide range of information handling problems from engineering and scientific applications to every day office management. The information in UWRIM is kept in tables with rows and columns.

The UWRIM system has two major parts. The first part is the self-contained program, **UWRIM** and the second part is the application program interface library **UWRMLIB.**

The self-contained part has its own language to describe the database, add, modify, and delete data, retrieve data and generate reports. The execution of UWRIM as "self-contained" or "stand-alone" can be effected in two modes, COMMAND or MENU mode. The command mode is used when UWRIM is executed in the batch environment or for interactive users who wish to bypass the menu dialogue. In MENU mode, UWRIM prompts you for the information for creating a new database or querying and updating an existing one. The interface library allows database manipulation from a FORTRAN or PASCAL program. An extensive "help" facility is also available with this system. To use the self-contained program enter the following control statement:

```
GRAB,UWRIM.
UWRIM.
```

To use the interface library enter the following control statement:

```
GRAB,UWRMLIB.
```

This adds UWRMLIB to your global library set. Following this, you need only compile and run a program using routines within this library.

Two manuals, a primer and a reference manual, are available for inspection at the systems consulting office, 1208 W. Springfield. The same manuals are available for sale at the Distribution Office in the same building.

STATISTICAL SERVICES

SAS USERS -- PLEASE TELL US WHO YOU ARE!

We are continuing to compile a list of local SAS users which we will use to notify SAS users of changes or enhancements to SAS, SAS training seminars, and new information we receive from SAS Institute. If you have not already contacted us and would like to be added to the list, please specify:

University ID#,
Office Telephone Number,
Name,
Staff Position/Title
Campus address

in a message to Greg Kesner via the CYBER Message Facility by entering:

TELL,UN=KESNER

or by mailing the information to:

Greg Kesner
150 DCL (CSO)
Campus

MATHEMATICAL SERVICES

CONVERSION OF SOME CYBER PACKAGES TO FTN5

In the last issue of *OFF-LINE*, we announced that a number of CYBER packages will be converted to FTN5, and gave a target date of

Monday, January 9, 1984

for the **final** conversion date. Prior to January 9, 1984, FTN5 versions will be available via the FUTURE and GRAB/F commands for user testing. After that date, the old versions will not be available except by special arrangement.

Here is the first group of products to become available as FTN5 versions under FUTURE:

IMSL
SLAM
FORSIM
GRG
FLEX
GMET
XMP

In the next few months, we will announce more. By early fall, there should be a complete line of the FTN5 versions available via FUTURE and GRAB/F. We strongly urge users of these packages to start now on converting to the use of FTN5.

SIMULATION SOFTWARE ANNOUNCEMENTS FROM PRITSKER AND ASSOCIATES

We have received two announcements of new software available from Pritsker and Associates, the vendor of the SLAM, GASP and QGERT simulation packages.

1. MicroNET [tm] is a powerful network simulation system developed specifically for microcomputers. Using a series of English-like commands, MicroNET models are easily created, edited, executed, and analyzed. MicroNET supports a full set of network simulation concepts, allowing a wide range of problem situations to be modeled and analyzed. MicroNET is designed for the IBM Personal Computer, the Apple II+ and many CP/M compatible microcomputers with as little as 48K RAM.

3. MAP/1 [tm] is a simulation-based Modeling and Analysis Program (MAP) used to design and evaluate batch manufacturing systems. It provides extensive capabilities for modeling and analyzing the various components of batch manufacturing systems, offering engineers and others an efficient means of understanding system productivity. Using MAP/1, in-depth simulation analyses may be performed to investigate alternative system configurations and operating procedures.

If you wish to see the brochures we received on these products, please call Stan Kerr at 333-4715, or write directly to

Pritsker and Associates
P.O. Box 2413
West Lafayette, Indiana 47906
Telephone: (317) 463-5557

DOCUMENTATION

NEW AND REVISED REFERENCE GUIDES

Following is a list of the new and revised Reference Guides now available at the RJE sites. Note that Reference Guides RF-0.1 and RF-0.2 have both been updated. We recommend that users pick up these revised guides to keep abreast of recent additions and/or changes.

Reference Guides

RF-0.1	Reference Guide List	Revised 06/09/83
RF-0.2	Documentation List	Revised 06/06/83
RF-1.6	RNF Ready Reference	Revised 06/09/83
RF-0.11	LocalNet	Revised 05/25/83

RF-0.15	LocalNet STATUS Command	New 05/25/83
RF-4.33	MATLAB	New 06/17/83
RF-30.5	Standard Terminal Settings for all CSO Computer Systems	New 06/06/83
RF-30.6	Transferring Files Between the CYBER and a Microcomputer	New 06/06/83

MISCELLANEOUS

STATISTICAL HELP IS AVAILABLE FOR THE SUMMER

The Mathematical and Statistical Consulting Committee (MSCC), under the directorship of Professor Robert Wijsman, Department of Mathematics, will continue its services through the summer. The MSCC office, 215 Altgeld Hall, is open from 3 pm - 5 pm, Monday through Friday. Appointments may be made by contacting Bruce Richardson at 333-9317 or via TELL,3KQQN4H.

SALES - HELP WANTED

PART-TIME POSITION AVAILABLE

Hourly position open for person to debug hardware and software for Z80 microcomputer based on data acquisition system. Hardware and software have been developed but not completely debugged. Assembly language programming experience and Z80 hardware knowledge would be helpful but not absolutely necessary. Hours are flexible according to the person's schedule but one could expect to average 10 hours per week. Start as soon as possible. Hourly rate will be determined by qualifications (ranging from $4.00 to $5.00 per hour). If interested contact:

Gary Cera
Room 191 RAL Box C-3
Telephone: 333-2274

GRADUATE ASSISTANTSHIP AVAILABLE

Assistant needed in August to work a minimum of 10 hours a week for a year or more. Duties include running an Apple II+ system for real-time data collection and using computer-aided drafting systems. Applicant must have a general knowledge of computer hardware and be familiar with Pascal and BASIC.

Please contact Ken Taylor, 103 Bevier Hall, 333-8488 or leave a message at 333-0737.

SKYMNK ARITHMETIC PROCESSOR FOR SALE

Produced by Sky Computers, Inc., this array processor for DEC systems brings a 30-300 times speed-up in LSI-11 performance.

Included is the MNKSIM software simulator and all documentation.

Condition: NEW
Contact: Carl Moore, 105 Noyes Lab, 333-1236

The manufacturer's list price is $7,052, though it has been offered for $4,800. You can have ours immediately for $3,500.

OFF-LINE's Mailing List

If you wish to be placed on our mailing list for future issues of *OFF-LINE*, if you wish to be removed from the list, or if you wish to enter an address correction, please complete and return this page. **(Current subscribers are kept on the mailing list until a specific request for removal is received, or until a mailing is returned as undeliverable.)**

Check one:
- ☐ Place my name on mailing list
- ☐ Make the following corrections or changes
- ☐ Delete my name from mailing list

_____

First name -- Initial -- Last Name

Campus Address:

_____

Room - Building

_____

Department

Off-campus Address:

_____

Organization or Company (if applicable)

_____

Street Address

_____

City -- State -- Zip Code

If address correction, give old address and zip code below.

_____

_____

_____

7/83

SEND TO:

OFF-LINE
150 Digital Computer Laboratory
University of Illinois at Urbana-Champaign
1304 West Springfield Avenue
Urbana, Illinois 61801

Computing Services Office

Off-Line

University of Illinois at Urbana-Champaign VOL. 11, NO. 8 August 1983

EDITOR: Lynn Bilger
PHONE: (217) 333-6236
120 Digital Computer Lab
1304 W. Springfield Ave.
Urbana, Illinois 61801

CSO DIRECTORY - STAFF AND SERVICES

Administrative				
Director	George Badger	150	DCL	333-4103
Business Manager	Stanley Rankin	150	DCL	333-6530
Secretary	Joyce McCabe	150	DCL	333-1637
User Services and Hardware/Software Support				
User Accounting		**1208**	**W Springfield**	**333-7752**
Distribution Office		**1208**	**W Springfield**	**333-6760**
Systems Consulting		**1208**	**W Springfield**	**333-6133**
Statistical Services Consulting		**85**	**Comm West**	**333-2170**
Text Processing Consulting		**118**	**DCL**	**333-7318**
Terminal Repair Service		**150**	**DCL**	**333-0969**
Dial-up Numbers	**CYBER 175 (NOSA)**	**300**	**baud**	**333-4000**
	CYBER 174 (NOSB)	**300**	**baud**	**333-4004**
	IBM 4341 L10 (VME)	**300**	**baud**	**333-4005**
	IBM 4341 M2 (VMD)	**300**	**baud**	**333-4006**
	Switch			**333-4008**
Asst Dir User Services	Robert Penka	173	DCL	333-4709
Asst Dir Systems and Operations	Sandra Moy	177	DCL	333-4703
Asst Dir Development	J. M. Randal	175	DCL	333-9772
Asst Dir Engineering and Hardware Selection	Cliff Carter	195	DCL	333-3723
Manager Accounting Services	Gary Bouck	1208	W Springfield	333-7752
Manager Statistical Services	Beth Richardson	189	DCL	333-6276
Microcomputer Laboratory	Jack Knott	86	Comm West	333-6562
Documentation	Lynn Bilger	120	DCL	333-6236
CYBER-IBM Operations	Myra Williams Ann Ryczer Rex Duzan	194a	DCL	333-6562
VAX Operations	Debbie Hudson	123	DCL	333-8150
Laboratory Support Project	Mike Gardner	18	DCL	333-7904
RJE Operations North	Sylvia Hansen Chuck Harnsberger	65	ME	333-6285
RJE Sites (see Reference Guide RF-0.3 for operating hours)				
Agriculture		M103	Turner Hall	333-8170
Chemistry		153	Noyes Lab	333-1728
Commerce West		70	Comm West	333-4500
CRH Snack Bar		120	Snack Bar	333-1851
DCL Routing Room		14	DCL	333-6203
Electrical Engineering		146	EEB	333-4936
Florida Ave Res Hall			FAR	333-2695
Illinois St Res Hall			ISR	333-0307
Mechanical Engineering		65	MEB	333-1430
Psychology		453	Psych Bldg.	333-7815
Social Science		202	Lincoln Hall	333-0309

OFF-LINE is the monthly newsletter of the Computing Services Office at the University of Illinois at Urbana-Champaign. Unless otherwise indicated, permission to reprint is freely granted, provided that the author, if named, and the Computing Services Office (CSO) are credited. Information in this issue is current as of July 20, 1983.

CSO operates a CDC CYBER 175 with 262K words of central memory and a CDC CYBER 174 with 196K words of central memory. The 175 and 174 run under the NOS Operating System and share 512K words of ECS. The 175 serves over 200 simultaneously active text and graphics terminals and the 174 serves over 100 simultaneously active terminals. CSO also operates an IBM 4341 Model M2 with 8 million bytes of memory running HASP-OS/MVT under VM and an IBM 4341 Model L10 with 4 million bytes of memory running VM. In addition, CSO operates a DEC VAX 11/780 with 4 million bytes of memory, running under the UNIX Operating System and driving a GSI CAT-8 phototypesetter.

Operating Hours (see HEARYE,SCHEDUL for exceptions):

	CYBERS 174/175	IBM
M-F	8 am - 6 am	8 am - 6 am
SAT	8 am - Midnight	8 am - 6 am
SUN	Noon - 6 am	Noon - 6 am

POLICY

RATE CHANGES

We plan to make a number of changes to our rate structure as noted below. All of the changes should be in place by the time this issue of *OFF-LINE* is distributed.

Most of the changes reduce the cost of computing. However, charges for prime time use of TELENET and the dialup facilities have been increased. In the case of TELENET, the increase is a pass through of the higher rates which CSO must pay TELENET. Our former TELENET charge was established several years ago and TELENET has since raised their prices significantly. The increase in the charge for dialup access was made to encourage use during off-peak hours and to recover some of the expense if we must expand dialup facilities to handle the prime time load. The changes are as follows:

- The cost of running a CYBER BULK job has been reduced from .18 to .12 of the normal rate.

- A 60% discount is now given to jobs run during low-cost periods. The previous discount was 40%. The discount applies to all charges except those for printing, plotting, and punching. Low-cost rates are in effect continuously every weekend between 4PM Friday and 8AM the following Monday morning and are sometimes announced during holiday periods.

- The hourly connect time fee has been reduced from 2 service units/hour to 1.6 service units/hour. Special terminals and terminals which are connected through TELENET or through the phone system are charged a higher rate as noted below.

- The hourly connect time fee for dialup terminals has been changed to 3 service units/hour between 9AM and 4PM daily. The fee remains at 2 service units/hour between 4PM and 9AM.

- TELENET charges have been increased to 14 service units/hour between the hours of 6AM and 9PM. Outside of this time the rate will be 7 service units/hour. The previous rate was 8 service units/hour.

- The charge for mounting a tape has been increased to 3 service units. The old rate was 2 service units.

- The low-cost discount has been extended to all account types. Previously the discount was available only to jobs funded through internal university account numbers, including contracts and grants.

- The formula used to calculate CYBER batch job execution units has been changed to

  $$SRU = 16(CP+IO + 0.012(CP+IO)CM) + CV$$

  where

  CP = CPU time in seconds
  IO = a measure of IO activity
  CM = kilowords of central memory
  CV = cover charge, presently set at 35 SRUs

  The previous formula was $SRU = 18(CP+IO + 0.015(CP+IO)CM + CV$

- The formula used to calculate CYBER time-sharing execution units has been changed to

  $$SRU = 20(CP+IO + 0.020(CP+IO)CM) + K$$

  where CP, IO, and CM are as above and K is the hourly connect time fee which varies as follows:

  K = 300 SRU/hour for dialup access between 9AM and 4PM
  200 SRU/hour for dialup access between 4PM and 9AM
  1000 SRU/hour for use of Tektronix 4014 or 4027 terminals
  1400 SRU/hour for access through TELENET between 6AM and 9PM
  700 SRU/hour for access through TELENET between 9PM and 6AM
  160 SRU/hour for all other kinds of access

  The previous formula was $SRU = 20(CP+IO + 0.028(CP+IO)CM + K$

SOFTWARE REGISTRY

CSO is trying to establish a registry of privately acquired or developed software on its machines as a means of encouraging the sharing of such software. If you have software that you think might be of use to others, please tell us

- its name (including version number if appropriate)
- a brief description of its function
- its availability (which machine(s) it runs on, whether there are legal restrictions on who may use it)
- the name and address of someone to contact for more information

You can send this information to Kurt Hirchert by campus mail (129 DCL) or the CYBER message system (TELL,HIRCHERT). If you have any questions about the registry, call Kurt at 333-8627.

STATISTICAL SERVICES

SAS USERS -- PLEASE TELL US WHO YOU ARE!

We are continuing to compile a list of local SAS users which we will use to notify SAS users of changes or enhancements to SAS, SAS training seminars, and new information we receive from SAS Institute. If you have not already contacted us and would like to be added to the list, please specify:

University ID#,
Office Telephone Number,
Name,
Staff Position/Title
Campus address (please include department)

in a message to Greg Kesner via the CYBER Message Facility by entering:

```
TELL,UN=KESNER
```

or by mailing the information to:

Greg Kesner
150 DCL (CSO)
Campus

MATHEMATICAL SOFTWARE

TRANSACTIONS ON MATHEMATICAL SOFTWARE

We now have on tape all algorithms published in Transactions on Mathematical Software up to March 1983 (from algorithm 493 published in the first issue of March 1975, up to algorithm 595). The algorithms from the June 1983 issue have been ordered, but have not yet arrived.

All TOMS algorithms are kept on a public tape; file TOMS/UN=MATHLIB contains a directory of this tape, plus directions for reading particular files on it. To print a copy of this directory, enter the following CYBER commands.

```
GET,TOMS/UN=MATHLIB.
PRINT,TOMS.
```

The "Math Note" series described in the March 1981 issue of *OFF-LINE* includes a note summarizing all algorithms published in TOMS to date. To print this note (it is note number 6), enter the following:

```
GET,MNOTES/UN=MATHLIB.
MNOTES,6.
PRINT,OUT/AS/CC/EJ.
```

(To print instructions on the use of MNOTES, substitute a 0 (zero) for the 6 above.)

The algorithms published in the March 1983 and June 1983 issues are as follows :

In the March 1983 issue:

592: A FORTRAN Subroutine for Computing the Optimal Estimate of f(x), by P. W. Gaffney

593: A Package for the Helmholtz Equation in Nonrectangular Planar Regions, by Wlodzimierz Proskurowski

594: Software for Relative Error Analysis, by John L. Larson, Mary E. Pasternak, and John E. Wisniewski

595: An Enumerative Algorithm for Finding Hamiltonian Circuits in a Directed Graph, by Silvano Martello

573: Remark on algorithm 573: NL2SOL -- an adaptive nonlinear least squares algorithm , by David M. Gay

In the June 1983 issue:

596: A Program for a Locally Parameterized Continuation Process, by Werner C. Rheinboldt and John V. Burkardt

597: Sequence of Modified Bessel Functions of the First Kind, by W. J. Cody

598: An Algorithm to Compute Solvents of the Matrix Equation $AX^2+BX+C=0$, by George J. Davis

599: Sampling from Gamma and Poisson Distributions, by J. H. Ahrens, K. D. Kohrt, and U. Dieter

600: Translation of Algorithm 507. Procedures for Quintic Natural Spline Interpolation, by John G. Herriot and Christian H. Reinsch

562: Remark on algorithm 562. Shortest Path Lengths., by U. Pape

The following articles which appeared in the March and June issues may be of interest to some of our users:

In the March 1983 issue:

A Functional Description of ANALYZE: A Computer-Assisted Analysis System for Linear Programming Models, by Harvey S. Greenberg

A Reduced Gradient Algorithm for Nonlinear Network Problems, by P. Beck, L. Lasdon, and M. Engquist

In the June 1983 issue:

Modeling Languages Versus Matrix Generators for Linear Programming, by Robert Fourer

The Efficient Solution of Linear Complementarity Problems for Tridiagonal Minkowski Matrices, by C. W. Cryer

A Locally Parameterized Continuation Process, by Werner C. Rheinboldt and John V. Burkardt

If you are looking for software, Transactions on Mathematical Software is one of the journals to check, along with "Numerische Mathematik", "BIT", the "Computer Journal", and various publications of ACM (the Association for Computing Machinery) and SIAM (the Society for Industrial and Applied Mathematics).

We have in the systems consulting office at 1208 W. Springfield bound volumes containing all the algorithms published in Communications of the ACM from 1960 to 1975 (when the algorithms department of CACM became Transactions on Mathematical Software).

CONVERSION OF SOME CYBER PACKAGES TO FTN5

In the June issue of *OFF-LINE*, we announced that a number of CYBER packages will be converted to FTN5, and gave a target date of

Monday, January 9, 1984

for the **final** conversion date. Prior to that date, FTN5 versions will be available via the FUTURE and GRAB/F commands for user testing. After that date, the old FTN versions will not be available except by special arrangement.

In the July issue, the first group of products to be converted to FTN5 was announced; this group was

IMSL	SLAM	FORSIM	GRG
FLEX	GMET	XMP	

In this issue, we announce the addition of the following products to this list

UOILIB	LINPACK	EISPACK	MINPACK	ITPACK
NETWORK	LSODE	AMOSLIB	GASP	FISHPAK

More products will be announced in the coming months.

IMSL NEWSLETTER

We have received the June 1983 issue of IMSL User News, IMSL's periodic newsletter telling of new IMSL products or information about current products. A copy is on display in the CSO Systems Consulting Office at 1208 W. Springfield. Additional copies are available from Stan Kerr, 179 DCL, 333-4715.

This issue of the newsletter announces a release date of July 1 for their new LP/PROTRAN product, a package similar in design to the PROTRAN package we have, with special linear programming abilities. If you are interested in obtaining LP/PROTRAN, please talk to Stan Kerr. At this point it seems unlikely we will obtain it, in view of the plethora of linear programming software already available here.

This issue also announces Edition 5 of TWODEPEP, IMSL's finite element program for solving elliptic, parabolic, and eigenvalue partial differential equations in general two-dimensional regions. We have not yet acquired TWODEPEP, due to its high lease price (about equal to that of the IMSL Library) and an apparent lack of local demand. Please talk to Stan Kerr if you are interested in it.

There are also two articles giving examples of the use of MATH/PROTRAN and STAT/PROTRAN.

COMBINED MATH/STAT PROTRAN NOW AVAILABLE

CSO has acquired a version of the PROTRAN package from IMSL which incorporates the abilities of both MATH/PROTRAN (which we announced some months ago) and STAT/PROTRAN. The new version can be accessed on the CYBER systems by the command

```
GRAB,PROTRAN/F.
```

or

FUTURE,PROTRAN.

The new version has been installed in FUTURE because of a few minor problems which we anticipate will be fixed when IMSL releases a new edition of the package this summer.

Manuals for STAT/PROTRAN are on order, and will sell for the same price as MATH/PROTRAN manuals ($15.50 with binder, $10 without). The MATH/PROTRAN binder is large, and will currently quite nicely accommodate both manuals.

With the advent of MATH/STAT PROTRAN, users will be able to write a PROTRAN program which calls on both the mathematical abilities (solution of linear equations, differential equations, integration, spline fitting, etc.) of MATH/PROTRAN and the statistical abilities of STAT/PROTRAN. The abilities of STAT/PROTRAN include the creation and manipulation of data files, simple statistics, analysis of variance, regression, correlation, and others. Moreover, because FORTRAN statements may be easily intermixed with PROTRAN statements in the same program, it is simple to go beyond the abilities of PROTRAN when that proves necessary.

If you have any questions about PROTRAN, please talk to Stan Kerr (179 DCL, phone 333-4715), or (for statistical questions) to Joan Alster (129D DCL, phone 333-2172).

MINICOMPUTERS, MICROCOMPUTERS, PERSONAL COMPUTERS

PROGRESS REPORT - MICROCOMPUTER SITE

An earlier edition of *OFF-LINE* announced our intention to establish a microcomputer site for use this Fall. Here's a progress report.

We encountered two suprises along the way. The first, a pleasant suprise, was the amount of interest the announcement generated. A number of people see this as a real opportunity. Consequently, we expect a fast startup in the use of the facilities. The second suprise was unpleasant. It is taking a lot longer than we had assumed to get delivery of the equipment.

Because of its popularity, we chose the IBM Personal Computer for this initial foray into microcomputers. A total of 50 units have been ordered of which 30 are being purchased by CSO. The remaining 20 are being purchased by the College of Business Administration but will be administered by CSO.

The minimum configuration ordered has 128K of memory, two disk drives, and a monochrome display. In addition, some units will be equipped with one or more of 192K of memory, a color graphics monitor, and a graphics printer. A single letter quality printer will also be available. All units will be equipped with the PC-DOS operating system, version 2.0.

A number of separate orders were written in the hopes that at least one source would be able to deliver in time for the start of the Fall semester. We still hope to have perhaps ten units in the minimum configuration available by the start of the semester or very shortly thereafter. The remainder should arrive during the Fall.

The initial installation (minimum of 20 microcomputers) will be at the RJE site at 70 Commerce West. We have had a number of other requests for clusters of microcomputers and offers of space to house them, but the final details have not yet been worked out. We hope to have a cluster of ten microcomputers on the North end of campus.

The LOTUS 1-2-3 software package has been ordered. By the time you read this, we expect to also have ordered a word processing package and spelling checker. This software is now being selected by a professor who will teach a course using it.

We need your help in selecting other software. If you would like to use our microcomputer facilities in your class, let us know what software you need. Similiary, if you are a researcher and need a package which you think would benefit a number of other researchers as well, tell us. We have budgeted money for software and want to spend it in the most productive way. When evaluating a proposal to buy a particular package we will consider the number of people who can be expected to use it and the contribution the package might make to general campus computing literacy.

If you would like to use our microcomputer facilities in a class, we would also like to know the mode of use and kind of equipment and environment you would need to make them most effective. Even if we can't accommodate you immediately, we expect the number of microcomputers to grow and we can factor in your needs when we next order more equipment. In any event, let us know if your class will use the equipment. We must schedule the use of sites to guarantee that equipment will be available when needed.

We are also willing to manage clusters of microcomputers which have been purchased by individuals or departments. We would expect some compensation, either in terms of some availability of equipment for general public use or money for the expenses we incur. If you are considering equipping a microcomputer cluster of your own but are put off by the problems of administering it, talk to us. We might be able to make a deal or at least offer some helpful suggestions.

If you have any questions about our microcomputer facilities or want to comment on any of the issues above, contact Jack Knott, 85 Comm West (333-6562).

MISCELLANEOUS

SERVICES OFFERED BY SURVEY RESEARCH LABORATORY

The University of Illinois Survey Research Laboratory (SRL), under the directorship of Richard B. Warnecke, maintains complete facilities for performing coding, on-line data entry, machine cleaning of data, analysis of data using SPSS, FORTRAN, SIR, and other statistical packages and program languages, and data-base development and management. Work is performed on a fee-for-service basis, and cost estimates can be provided on request.

In addition to providing these services, SRL offers one hour of advice per year at no charge on any data reduction, processing, or analysis problem.

SRL's Urbana office is located at 1005 West Nevada and is open year-round from 8 am to 5 pm, Monday through Friday. Appointments can be arranged by phoning Sue Lewis at 333-7674 or SRL's main office at 333-4279.

HELP WANTED

HALF-TIME RESEARCH ASSISTANT

The Computing Services Office (CSO) has an opening for a graduate student as a half-time research assistant to work in the Statistical Services area. The duties are to provide consulting assistance in the use of statistical software. The applicant must have some training in applied statistics, good communication skills, and a working knowledge of the SAS and SPSS statistical packages. If interested, please contact:

Beth Richardson
189 DCL
Telephone: 333-6276

OFF-LINE's Mailing List

If you wish to be placed on our mailing list for future issues of *OFF-LINE*, if you wish to be removed from the list, or if you wish to enter an address correction, please complete and return this page. **(Current subscribers are kept on the mailing list until a specific request for removal is received, or until a mailing is returned as undeliverable.)**

Check one:

- ☐ Place my name on mailing list
- ☐ Make the following corrections or changes
- ☐ Delete my name from mailing list

_____

First name -- Initial -- Last Name

Campus Address:

_____

Room - Building

_____

Department

Off-campus Address:

_____

Organization or Company (if applicable)

_____

Street Address

_____

City -- State -- Zip Code

If address correction, give old address and zip code below.

_____

_____

_____

8/83

SEND TO:

OFF-LINE
150 Digital Computer Laboratory
University of Illinois at Urbana-Champaign
1304 West Springfield Avenue
Urbana, Illinois 61801

Computing Services Office

Off-Line

University of Illinois at Urbana-Champaign VOL. 11, NO. 9 September 1983

EDITOR: Lynn Bilger
PHONE: (217) 333-6236
120 Digital Computer Lab
1304 W. Springfield Ave.
Urbana, Illinois 61801

CSO DIRECTORY - STAFF AND SERVICES

Administrative

Director	George Badger	150	DCL	333-4103
Business Manager	Stanley Rankin	150	DCL	333-6530
Secretary	Joyce McCabe	150	DCL	333-1637

User Services and Hardware/Software Support

User Accounting		**1208**	**W Springfield**	**333-7752**
Distribution Office		**1208**	**W Springfield**	**333-6760**
Systems Consulting		**1208**	**W Springfield**	**333-6133**
Statistical Services Consulting		**85**	**Comm West**	**333-2170**
Text Processing Consulting		**118**	**DCL**	**333-7318**
Terminal Repair Service		**150**	**DCL**	**333-0969**
Dial-up Numbers	**CYBER 175 (NOSA)**	**300**	**baud**	**333-4000**
	CYBER 174 (NOSB)	**300**	**baud**	**333-4004**
	IBM 4341 L10 (VME)	**300**	**baud**	**333-4005**
	IBM 4341 M2 (VMD)	**300**	**baud**	**333-4006**
	Switch			**333-4008**
Asst Dir User Services	Robert Penka	173	DCL	333-4709
Asst Dir Systems and Operations	Sandra Moy	177	DCL	333-4703
Asst Dir Development	J. M. Randal	175	DCL	333-9772
Asst Dir Engineering and Hardware Selection	Cliff Carter	195	DCL	333-3723
Manager Accounting Services	Gary Bouck	1208	W Springfield	333-7752
Manager Statistical Services	Beth Richardson	189	DCL	333-6276
Microcomputer Laboratory	Jack Knott	86	Comm West	333-6562
Documentation	Lynn Bilger	120	DCL	333-6236
CYBER-IBM Operations	Myra Williams Ann Ryczer Rex Duzan	194a	DCL	333-6562
VAX Operations	Debbie Hudson	123	DCL	333-8150
Laboratory Support Project	Mike Gardner	18	DCL	333-7904
RJE Operations North	Sylvia Hansen Chuck Harnsberger	65	ME	333-6285

RJE Sites (see Reference Guide RF-0.3 for operating hours)

Agriculture	N-120	Turner Hall	333-8170
Chemistry	153	Noyes Lab	333-1728
Commerce West	70	Comm West	333-4500
CRH Snack Bar	120	Snack Bar	333-1851
DCL Routing Room	14	DCL	333-6203
Electrical Engineering	146	EEB	333-4936
Florida Ave Res Hall		FAR	333-2695
Illinois St Res Hall		ISR	333-0307
Mechanical Engineering	65	MEB	333-1430
Psychology	453	Psych Bldg.	333-7815
Social Science	202	Lincoln Hall	333-0309

OFF-LINE is the monthly newsletter of the Computing Services Office at the University of Illinois at Urbana-Champaign. Unless otherwise indicated, permission to reprint is freely granted, provided that the author, if named, and the Computing Services Office (CSO) are credited. Information in this issue is current as of August 26, 1983.

CSO operates a CDC CYBER 175 with 262K words of central memory and a CDC CYBER 174 with 196K words of central memory. The 175 and 174 run under the NOS Operating System and share 512K words of ECS. The 175 serves over 200 simultaneously active text and graphics terminals and the 174 serves over 100 simultaneously active terminals. CSO also operates an IBM 4341 Model M2 with 8 million bytes of memory running HASP-OS/MVT under VM and an IBM 4341 Model L10 with 4 million bytes of memory running VM. In addition, CSO operates a DEC VAX 11/780 with 4 million bytes of memory, running under the UNIX Operating System and driving a GSI CAT-8 phototypesetter.

Operating Hours (see HEARYE,SCHEDUL for exceptions):

	CYBERS 174/175	IBM
M-F	8 am - 6 am	8 am - 6 am
SAT	8 am - Midnight	8 am - 6 am
SUN	Noon - 6 am	Noon - 6 am

POLICY

SHORT COURSES - FALL SEMESTER 1983

CSO is offering the following noncredit short courses during the Fall semester 1983 to acquaint potential users with our computing systems, facilities and services.

Starting this Fall, CSO will be making a small charge for most short courses. This is due to two factors: (1) There is a need for equipment to support improvement in teaching methods; (2) The volume of short courses has risen to the point where it is a serious drain on consulting staff time, and some compensation in staffing must be made. The income is dedicated to support of the short course program.

Short Course Policy

Please note that:

- **REGISTRATION IS REQUIRED** for all courses except where noted. Registration is accomplished by filling out a copy of the registration form for each course and sending these forms, with payment for the total of indicated fees, to CSO in either campus or U.S. mail.

- Each registrant will be sent a confirmation of registration on which the place of meeting is noted. This slip must be taken to all meetings of the class and shown when requested.

- Refunds of fees will be made only for canceled classes, or upon receipt of an application for refund on or before the day **BEFORE** the second meeting of the class. (There will be no such refunds for classes that meet only once.) Application for a refund must be made in room 150 DCL during normal office hours.

- A copy of the current (updated) short course listing may be examined on-line via:

  TYPE, COURSES/AS/UN=COURSES.

- A brief list of course titles, fees and meeting times is available via:

  TYPE, TITLES/AS/UN=COURSES.

- CSO reserves the right to cancel courses or sections with insufficient enrollment. All fees paid for these classes will automatically be refunded.

- Updates (changes) since the printed short course listing was issued may be examined on-line via:

  TYPE, CHANGES/AS/UN=COURSES.

Questions, comments and suggestions should be addressed to the CSO short course coordinator: Ron Szoke, 333-8630 or TELL,SZOKE from a CYBER terminal.

The following manuals are required for some of the courses listed below. These manuals may be purchased at the Illini Union Bookstore (Reference Section), 715 South Wright Street, or are available

(with the exception of the RNF Documentation which must be purchased separately) as a package at the CSO Distribution Office, 1208 W. Springfield, for a cost of $11.00.

- Introduction to the CYBER Systems $2.00
- A Tutorial Guide to the ICE Text Editor $1.25
- ICE Reference Manual $3.25
- RNF Documentation $4.00
- An Index to Software on the CYBER $3.25
- CYBER Fortran Debugging $1.25

Short Course Listing

NOTE: Since some of the fall short courses (August and early September) already will have been given before publication of this issue, only courses after September 9 will be listed.

1. Orientation to CSO Facilities and Services

A brief, nontechnical presentation for prospective users on the following topics: the mission and organization of CSO; computing equipment; available software; locations of facilities and hours of operation; where to obtain documentation; short courses and video tapes; consulting and other services; how to set up a computer account. Prerequisites: none. Recommended manual: #5 (see above list). Fee: none. NO REGISTRATION IS REQUIRED.

5. Sept 20 (Alster)	3:00pm -	4:00pm	168 EE
6. Sept 22 (Szoke)	12noon -	1:00pm	115 DCL

2. Computing for Poets

A brief orientation to computers and computing for those in the humanities and fine arts. Intended especially for those with "computer anxiety" and those who are "anti-computer" or "anti-technology." The course assumes no background in computing, mathematics, or any other technical field. Fee: $15.

Oct 31, Nov 2,4 (Szoke)	12noon -	1:00pm

3. Writing with a Word Processor

How to use a microcomputer and word processing package to produce publication-ready manuscripts. Prerequisite: none, but Computing for Poets (above) or equivalent knowledge of basic computer terminology is recommended. Fee: $20.

Nov 7,8,9,10 (staff)	12noon -	1:00pm

4. Basic Concepts in Computer Information Processing

An introductory survey of computer information processing concepts and terminology, intended for those thinking of acquiring a small computer or word processing equipment. Some guidelines for buying a personal or microcomputer are suggested, but attendees will NOT be told

which one to buy, nor which one is "best." The needs of UI administrative, clerical and non-academic personnel have been central to planning of the course. Prerequisites: none. Fee: $20.

Sept 19,20,21,22 (Szoke & others)	3:00pm -	4:00pm

5. Introduction to the CYBER System

This course is intended for the first-time CYBER system user. The emphasis is on timesharing usage and the ICE text editor. Batch usage may also be discussed. Required manuals: #1 and #2 (see above list). Prerequisite: Orientation (above). Fee: $15.

2. Sept 12,13,14,15,16 (Edwards-lwe)	3:00pm -	4:00pm
3. Sept 26,27,28,29,30 (Primer)	8:00am -	9:00am
4. Oct 10,11,12,13,14 (Foertsch)	8:00am -	9:00am
5. Nov 7,8,9,10,11 (Scheid)	3:00pm -	4:00pm

6. Introduction to IBM Timesharing: CMS

This course introduces CMS (Conversational Monitor System), the IBM timesharing system. Special emphasis is placed on using CMS and XEDIT to run SAS and SAS/GRAPH programs, though the content is general enough to help any user begin working with the IBM timesharing system. Topics include: general CMS and virtual machine concepts, useful standard and locally-written CMS commands and utilities, sending files between the CYBERs and CMS, guidelines for utilizing the available CMS documentation, how to use the full-screen simulator (SIM), and how to submit OS batch jobs and receive their output under CMS. The course consists of two 2-hour lectures and a 1½-hour lab. Fee: $15.

2. Nov 8,10 (Kesner)	7:00pm -	9:00pm
Nov 19 -- Lab	8:30am -	10:00am

7. Introduction to IBM Timesharing: XEDIT

This course introduces the IBM XEDIT text editor which is used under CMS. The lecture covers useful commands for both "ASCII typewriter" and "full-screen" or "simulated full-screen" terminals. Useful locally-written XEDIT commands are also discussed. The course, consisting of a 2-hour lecture and a 1-hour lab, will be offered in conjunction with the introductory CMS short course. Fee: $10.

2. Nov 17 (Kesner)	7:00pm -	9:00pm
Nov 19 -- Lab	10:00am -	11:00am

8. Introduction to Statistical Package Computing

The purpose of this course is to equip the novice computer user or the newcomer to this campus with the basic knowledge necessary to utilize CSO's computing resources to perform statistical analyses. Topics covered: an overview of the computer system and the statistical packages available, file concepts, program and data entry using the ICE text editor, and a glimpse of the four major statistical packages available: SPSS, SAS, SOUPAC, and BMDP. (Note: this course is NOT intended to replace the regular introductory courses for each package.) Two 2-hour lectures followed by a 2-hour lab session covering basic file concepts and timesharing terminal usage. Prerequisite: Orientation to CSO (above). Fee: $15.

2. Sept 20,21 (Mills)	7:00pm -	9:00pm
Lab - Sept 22		

9. Introduction to RNF

A beginning level discussion of the RNF text formatter, used in the preparation of documents such as papers, theses and manuals. Topics will include tabbing, margins, spacing, paragraphing, and justification. Familiarity with the CYBER system and ICE is assumed. Required manual: #4 (see list above). Fee: $15. Two sections are offered.

1. Sept 19,21,23 (DeWan)	12noon -	1:00pm
2. Oct 10,12,14 (DeWan)	12noon -	1:00pm

10. Intermediate ICE (Illinois Central Editor)

A course on the use of the ICE text editor, covering techniques for solving a variety of editing problems: reading from and writing to local files, copy and transfer of text lines, extended modify, case conversion, tab fields, windows, loops, stripe, procedures, and others. Assumes: Introduction to CYBER or Introduction to Statistical Packages, or the equivalent. Required manual: #3, (see list above). Fee: $15.

Oct 31, Nov 2,4 (DeWan)	12noon -	1:00pm

11. Use of MANAGE

How an instructor or project director allocates computer resources to individual users. Assumes familiarity with simple use of the CYBER. Fee: none.

Sept 13 (Primer)	8:00am -	10:00am

12. Using Magnetic Tapes

Covers the use of magnetic tapes on the CYBER. Assumes familiarity with use of the CYBER. Registrants should obtain the free manual, "Magnetic Tape Guide for the CYBER," at 1208 West Springfield or 85 Commerce West before attending. Fee: $20.

Oct 10,12,14,17 (Wetzel)	12noon -	1:00pm

13. Data Communications and Networking

This class will discuss: connecting a terminal to a computer both locally and remotely; long distance communication media; EIA RS232 standard interface wiring; modems; multiplexing; and local area networks. The discussion will be general and not necessarily deal with any particular hardware. Assumes: Basic Concepts (above) or equivalent knowledge. Fee: $20.

Oct 10,11,12,13 (Krol & Lathrop)	3:00pm -	4:00pm

14. Easy Graphing

Elementary presentation of a high-level interactive plotting program for X-Y plots, bar and pie charts. Its English-like commands require no programming experience to generate plots. Assumes: Introduction to the CYBER or equivalent knowledge. Fee: $15.

Sept 26,28,30 (Pommert)	3:00pm -	4:00pm

15. Introduction to the DI-3000 Graphics Package

DI-3000 is a library of FORTRAN callable subroutines for computer graphics. It is an implementation of the 1979 ACM/SIGGRAPH CORE graphics system proposed standard. This course is for those with little or no experience using DI-3000. Covers capabilities, structure, and basic use of the package. No high level or X-Y plotting capabilities will be presented. Assumes a working knowledge of CYBER FORTRAN. Fee: $15.

Oct 5,6,7 (Tuchman) 12noon - 1:00pm

16. Advanced DI-3000 Computer Graphics

A course for users with a solid basic knowledge of DI-3000. Covers retained segments, text attributes, graphics input, and 3-D viewing in DI-3000. Assumes: Introduction to DI-3000 or equivalent knowledge. Fee: $15.

Oct 24,26,28 (Tuchman) 12noon - 1:00pm

17. Survey of Mathematical Software

This course provides a brief summary of the mathematical (non-statistical) software available at CSO. The lecture will cover subroutine libraries and software for linear systems, eigenanalysis, differential equations, data fitting, and other uses. Assumes a working knowledge of CYBER FORTRAN. Fee: none.

Sept 15 (Kerr) 12noon - 1:00pm

18. Using the UWRIM Database Management System

UWRIM (University of Washington Relational Information Management) is an easy to learn and easy to use database management package based on the relational algebra model of data organization. It can be used to solve a wide range of information handling problems, from engineering and scientific applications to everyday office management. This course shows how to store data in simple tables that can be sorted and accessed in various ways, as well as some applications of UWRIM. Prerequisite: Introduction to the CYBER System. Fee: $15.

Nov 7,9,11 (Edwards-Iwe) 3:00pm - 4:00pm

19. The TAGS Mailing List System

TAGS is a procedure for creating, printing, and storing mailing labels and lists on the CYBER computers. Labels can be quickly changed or modified in many ways. One of TAGS' most useful features is its ability to sort labels according to name, address, zip code, or any of three other items one can enter as "keys." TAGS is an application of UWRIM, a relational database management system that collects and manipulates tables of data. This course explains how TAGS can be used to generate and print labels in different formats. Prerequisite: Using UWRIM (above). Fee: $10.

Nov 15,17 (Edwards-Iwe) 3:00pm - 4:00pm

20. CYBER FORTRAN-77 (FTN5) for the FTN User

A systematic presentation of the principal differences between CYBER FORTRAN IV (FTN) and the more recent FORTRAN-77 standard, as implemented by the CYBER FTN5 compiler.

(This new compiler will become the default at some time in the future.) How to convert FTN programs to FTN5 with the aid of F45. Assumes a good working knowledge of FTN. Fee: $25.

Sept 12,13,14,15,16 (Hirchert)	8:00am -	9:00am

21. Using Subroutine Libraries

This course deals with how the user can create, use and modify his own library of special or frequently used subroutines. Fee: $5.

Oct 6 (Hirchert)	3:00pm -	4:00pm

22. CYBER Control Language (CCL)

CCL can be used in batch jobs or procedures to control the flow of execution, alter the control statements to be executed, issue messages to the terminal, and generate text files which can be used as input to programs within a batch job or CCL procedure. This course will cover most CCL statements, CCL procedures, and, as time permits, related commands and examples which illustrate the various aspects of CCL. Assumes: Introduction to CYBER or equivalent. Fee: $10.

Oct 25,27 (Pommert)	3:00pm -	4:00pm

23. Using the PROTRAN Package

This course describes a new package from IMSL, Inc., which can greatly ease the burden of mathematical and statistical problem-solving by permitting one to input a simple, English-like description of a problem. Assumes: Introduction to CYBER or equivalent. Two sessions will be offered. Fee: $20.

1. Sept 19,20,21,22 (Kerr)	3:00pm -	4:00pm
2. Oct 17,18,19,20 (Kerr)	3:00pm -	4:00pm

24. Introduction to SPSS (Statistical Package for the Social Sciences)

This is an introductory course on the SPSS statistical package on the CYBER system. Topics covered: data preparation, basic components of the SPSS language, data modification, running SPSS programs at terminals, and using SPSS to obtain basic statistics and contingency tables. Participants are expected to have prior experience with program and data entry using the ICE text editor. Copies of the course handout are available in 85 Commerce West for participants who wish to read the material prior to attending the class. The course consists of two 2½-hour lectures and one 1-hour lab. Prerequisite: Intro to Stat Package Computing or equivalent knowledge. Fee: $30.

Oct 11,13 (Richardson)	6:30pm -	9:00pm
Oct 15 -- Lab	9:00am -	10:00am

25. Introduction to SCSS (SPSS Conversational System)

SCSS is a comprehensive set of data management and statistical analysis facilities available only on the IBM computer. Using SCSS requires very little computer expertise because very little time is spent learning to program. SCSS is so named because communication between the user and the system resembles an actual conversation: it asks a series of questions to which the user responds with his/her choice of appropriate answers. When asked to display or analyze data,

SCSS does so on the spot. The command language is simple, consisting of English words, and if a command is forgotten or misspelled, the system prompts the user for the information it needs. The facilities in SCSS will be described and, in the lab session, easy sample exercises will enable participants to quickly become familiar with using the SCSS system. Fee: $10.

Oct 5 (Richardson)	7:00pm	-	8:00pm
Oct 8 -- Lab	9:00am	-	10:00am

26. Introduction to SPSS/GRAPHICS

SPSS Graphics is available as part of the SPSS package under both the IBM OS/MVT and VM/CMS systems. It consists of three procedures: PIECHART, BARCHART and LINECHART, which are fully integrated into the SPSS system and use the familiar SPSS syntax. Commands for generating pies, bars and curves will be described, and techniques for producing plots under both the IBM batch and timesharing systems will be illustrated. Recommended manual: *SPSS Update 7-9* (1981), Chapter 4. Prerequisite: Prior experience with SPSS. Fee: $15.

Oct 25 (Richardson)	7:00pm	-	9:00pm
Lab to be arranged (1 hour).			

27. Introduction to SPSS-X (Statistical Package for the Social Sciences)

This course is designed for experienced SPSS users who wish to make the transition to SPSS-X, which is available only on the IBM computer. SPSSX offers a wide range of enhancements, especially in the areas of file definition, data manipulation and file management. These new features of SPSSX will be illustrated in a series of applications. Topics covered: basic file definition in SPSSX, moving from SPSS to SPSSX, working with system files in SPSSX, merging files, and working with complex files. The course will not concentrate on syntax or statistical procedures; rather, it is designed to ease the transition from SPSS to SPSSX by pointing out differences and enhancements. Prerequisite: Prior experience with SPSS. Fee: $30.

Nov 7,9,14 (Richardson)	7:00pm	-	9:00pm

28. SPSS and SPSS-X Processing Using CMS

This course provides an introduction to running SPSS and SPSSX programs using the IBM CMS timesharing system. It is intended for current SPSS and SPSSX users who wish to make the transition to CMS. Topics covered: the system command for invoking SPSS or SPSSX and its options; the CMS commands needed to define raw data input/output files and system file input/output; submitting SPSS jobs to OS/MVT from CMS. Prerequisite: Introduction to IBM Timesharing (CMS and XEDIT) and prior experience with SPSS. Fee: $10.

Sept 13 (Richardson)	7:00pm	-	8:00pm
Sept 17 -- Lab	9:00am	-	10:00am

30. Introduction to SAS (Statistical Analysis System)

An introductory course on the basics of SAS processing on the IBM 4341. Also offers an overview of more advanced SAS processing concepts and techniques which are presented in the SAS Basics Video Series. (This video series can be seen at the Media Center in the Undergraduate Library.) Examples will illustrate using SAS under CMS (the IBM timesharing) system and submitting OS/SAS jobs from CMS. Topics include: Using the DATA and PROC steps, creating temporary and permanent SAS data sets under OS and CMS, reading and writing

external data files, using basic SAS procedures, programming in the DATA step, and an overview of CSO SAS features and services. Prerequisite: Introduction to Statistical Package Computing (above) or equivalent knowledge of local system usage. Recommended reference: *SAS Introductory Guide.* The course consists of four 2-hour sessions and is OFFERED ONLY ONCE THIS SEMESTER. Fee: $40.

Oct 18,20,25,27 (Kesner) 7:00pm - 9:00pm

31. Introduction to CMS/SAS

An introduction to SAS processing using the IBM CMS timesharing system. Topics include: using SAS interactively and non-interactively, creating and maintaining SAS data libraries under CMS, reading and writing external CMS files, using SAS/FSP (the SAS Full-Screen Product) to interactively edit SAS data sets, highlights of other SAS/FSP features, and submitting batch OS/MVT SAS jobs from CMS. All SAS users are encouraged to make the transition from CYBER timesharing to CMS to perform their SAS processing because of the marked advantages CMS offers for such work. Prerequisites: Introduction to IBM Timesharing: CMS and XEDIT and Introduction to SAS or equivalent knowledge. NOTE: This short course is offered only ONCE this term-- SAS users should make every effort to attend if they need this information. Fee: $20.

Sept 12,13 (Kesner) 7:00pm - 9:00pm

32. Introduction to CMS/SAS/GRAPH

An introduction to using SAS/GRAPH under the local IBM CMS timesharing system and CMS/SAS. Topics include: how to use SAS/GRAPH with various graphics devices, how to produce hard-copy Zeta plots from SAS/GRAPH output, global features of SAS/GRAPH, using map data sets, calculating dimensions and proportions for Zeta plots, and highlights of new features in SAS/GRAPH. Prerequisites: Introduction to IBM Timesharing: CMS and XEDIT, Introduction to SAS, and Introduction to CMS/SAS; or equivalent knowledge. NOTE: This course is offered only ONCE this term-- SAS users should make every effort to attend if they need this information. Fee: $20.

Nov 1,3 (Kesner) 7:00pm - 9:00pm

33. Introduction to SOUPAC (Statistically Oriented User's Package)

This is an introductory course on the use of the SOUPAC statistical package on the CYBER system. Topics covered: data preparation, structure of the SOUPAC language, data modification, running SOUPAC programs at terminals, and using SOUPAC for basic statistics, regression, and analysis of variance. The course, consisting of a three-hour lecture and a two-hour lab (to be arranged), will be offered once this semester. Prerequisite: Introduction to Statistical Package Computing or equivalent knowledge. Fee: $25.

Oct 4 (Mills) 6:30pm - 9:30pm

34. Introduction to BMDP (Biomedical Statistical Programs)

BMDP, a widely-used package of statistical programs developed by UCLA's Department of Biomathematics, is oriented toward the biological, medical, nutritional, agricultural and veterinary sciences. This is an introductory course on the use of BMDP on the CYBER system. Topics covered: data preparation, elements of the BMDP language, running BMDP programs at terminals, data editing, creation and use of system files, and examples of using BMDP for

descriptive statistics, regression and analysis of variance. The course, consisting of two 2-hour lectures and a lab session (to be arranged), will be offered once in the Fall semester. Prerequisite: Intro to Stat Package Computing or equivalent knowledge. Fee: $30.

Nov 15,16 (Szoke) 7:00pm - 9:00pm

35. Using SPSS MANOVA (Multivariate Analysis of Variance)

This course provides an introduction to the use of SPSS MANOVA for univariate and multivariate analyses of variance in the general linear model context. Topics include: analysis of variance, analysis of covariance, regression analysis, use of special contrasts, partitioning sums of squares, and choice of model for special designs. As time permits, SPSS MANOVA will be compared and contrasted with the SAS GLM procedure. Some familiarity with SPSS and analysis of variance is assumed. Recommended reference: *SPSS Update 7-9* (McGraw-Hill, 1981), pp. 1-79. Fee: $15.

Oct 17,19,21 (Alster) 3:00pm - 5:00pm

36. Using BITNET

A brief presentation on how to use the BITNET communications facility, which links over 30 IBM/CMS installations (mostly universities, including UIUC) across the country, and which allows the transmission of files as well as messages. Fee: $5.

Sept 29 (Foertsch) 12noon - 1:00pm

Computing Services Office -- UIUC
Short Course Registration Form, Fall 1983

Note: You must have the current short course listing at hand in order to register. USE A SEPARATE SHEET FOR EACH COURSE. This form may be copied freely as needed. Fill in all blanks down to the "****" and send completed forms to:

CSO Short Courses
150 DCL
1304 West Springfield
Urbana, IL 61801

University ID (Social Security) number (9 digits) _____

Name: (Please print clearly)
last _____, first _____

Telephone: Office _____ Residence _____

Send receipt to: (address)

_____

_____

____ CAMPUS MAIL or ZIP _____

Course number	Section number	Course title	Course fee enclosed
_____	_____	_____	$_____

Paid by:

_____ Personal or organizational check made payable to University of Illinois.

_____ University account number (11 digits): _____
Note: "Real money" accounts only; Research Board, SARA, and class accounts are NOT acceptable.

Account title:_____

- -

This portion will be returned. You must take it to class with you.

Computing Services Office -- UIUC
Short Course Registration Receipt and Admission Slip, Fall 1983

Course	Section	IDN	Last name	Initial
_____	_____	_____	_____,	_____

***

Meets in room: _____

Note: _____

Validation by registrar:

Date:_____/_____/_____

CHANGES IN PORT SWITCH (GANDALF PACX) ACCESS

In order to better manage the 1200 baud dial-up service and allow easier use of the facilities of PACX, CSO will make some changes in the initial dialog with the switch at 8:00am on October 10, 1983. These changes are:

- The selection of machine will be by alphanumeric string (class name) from the following list. Any entry by class number (e.g., 175) will no longer be accepted. Any other alphanumeric strings which are not in the following list will cease working.

- The class name must be appropriate to the terminal being used to access the switch. The convention to be used is as follows: hardwired public terminals are to use the standard format machine name (e.g., NOSA for the 175); private rental access terminals are to use the standard name preceded by a "R" (e.g., RNOSA); and, finally, 1200 baud dial-up service will use the standard format name prefixed by a "D" (e.g., DNOSA).

- The valid class names are:

NOSA	-	CYBER 175 NOS from direct connect switch terminals
NOSB	-	CYBER 174 NOS from direct connect switch terminals
DNOSA	-	CYBER 175 NOS from 1200 baud dial-up lines
DNOSB	-	CYBER 174 NOS from 1200 baud dial-up lines
RNOSA	-	CYBER 175 NOS from privately rented switch terminal
VMD	-	IBM 4341 M2 from direct connect switch terminals
VME	-	IBM 4341 L10 from direct connect switch terminals
DVMD	-	IBM 4341 M2 from 1200 baud dial-up lines
DVME	-	IBM 4341 L10 from 1200 baud dial-up lines
UXC	-	VAX 780 Unix from direct connect switch terminals

FREE STUDENT ACCOUNTS

Beginning with the 1983 Fall Semester, students will be able to apply for a **Free Student Account** number at the following locations:

CSO South Operations Room 70 Commerce West	or	CSO Accounting Office 1208 W. Springfield, Urbana

The Free Student Account is available to students currently enrolled at the University of Illinois. Each student is required to present a valid University I.D. to obtain the student account.

Current restrictions on obtaining a Free Student Account are as follows:

- Persons having an active SARA account (graduate students in the process of a Masters or Doctoral Thesis) are not eligible.

- Students currently enrolled in 100 level CS courses are not eligible.

ELIMINATION OF SOME CARD HANDLING EQUIPMENT

Over the next nine months we will be eliminating some of our card handling equipment. This machinery is getting quite old and is becoming more difficult and expensive to maintain with the passage of time. Since card usage has steadily declined over the past few years we will eliminate much of the equipment rather than replace it with more modern gear.

The company which manufactured our card readers has notified us that, starting next January, parts will no longer be available. Because of this we will retire all but two of our card readers, leaving the ones at the RJE sites of Commerce West and Mechanical Engineering. We presently use the card readers to load the RJE software in the PDP-11 minicomputers which form the heart of the RJE stations. We have ordered equipment which will allow us to load the software from an electronic image which will be stored within the RJE equipment. We will retire the card readers when this equipment has been installed.

In recent years nearly all of our keypunch use has been by students in large classes which could not be accommodated on terminals because of lack of equipment. With the installation of the second large cluster of student terminals, the last such large class will switch from cards to terminals this fall. Because of the expense of maintaining them, all but six of the remaining keypunches will be withdrawn from service. Three keypunches will be left at each of the Commerce West and Mechanical Engineering RJE sites. We will remove the other keypunches as soon as we have made arrangements for their disposition.

Finally, as we announced in the May edition of *OFF-LINE*, we will retire the card punch by May, 1984. After talking with people who raised objections to this announcement, we think most of services provided by the card punch can be provided in some other way. We will supply microcomputers and software which will conveniently transfer data between the CYBER or IBM systems and a floppy disk. If you have an overriding need for punched output from either the CYBER or the IBM system we can arrange to have the cards produced at some other computer center, transferring the data there by tape. If your work will be adversely affected by removal of the card punch please call Bob Penka (333-4709) to discuss your needs.

CYBER

VIDEOTAPES

CSO has produced a series of eight videotapes which introduce the novice to the CYBER systems. A viewing guide containing the major pictorials used in the series is available and can be used to facilitate note taking.

The title and a brief synopsis of each of the videotapes is given below. Running time is 10-15 minutes for each videotape.

Introduction to Computing at CSO

A brief look at the steps required to solve a problem using a computer and some of the hardware used.

Using a Terminal

A description of the physical operation of a terminal and some of the keys that have a special meaning to the CYBER.

Introduction to CYBER Timesharing

A tutorial on the logging on and off of the CYBER.

File Usage - Local File and Indirect Access to Permanent Files

An introduction to CYBER files and the commands used to manipulate them.

Introduction to ICE Text Editing

A tutorial on entering and modifying files with ICE.

Running a FORTRAN Program - Concepts

A discussion of the concepts of compilation loading and execution.

Running a FORTRAN Program - The PROGRAM statement

A discussion of the PROGRAM statement and its relationship to files accessed by the program.

Running a FORTRAN Program - Control Statement

A discussion of the control statements used to compile, load, and execute a FORTRAN program.

Anyone can view these videotapes by going to the Media Center desk in the Undergraduate Library in person to make a reservation for use of the video equipment. Ask for a copy of the viewing guide when you check out the videotape for viewing.

A copy (Betamax format) of these videotapes is available for loan from CSO to any instructor wishing to use them in class. They have been effectively used in this environment several times recently, with the instructor stopping the playback equipment whenever he/she wished to elaborate further or questions arose from the class.

To borrow a videotape for classroom use and obtain copies of the viewing guide for classroom distribution, call Ron Szoke (333-8630). If you do not already have access to the required videotape equipment, Betamax viewing equipment can be borrowed from the Office of Instructional Resources (333-3690).

IBM

CSO SAS (STATISTICAL ANALYSIS SYSTEM) SERVICES

The SAS Program Products Offered by CSO

Over the last five years, SAS has grown from a library of statistical routines to a large system of sub-libraries of routines for data management, statistical data analysis, report writing, color graphics, and even interactive systems development for office, business, research, and educational environments. As an integrated system of procedures useful for accomplishing a wide range of programming and analytical applications, SAS requires very little educational overhead from the user to serve his or her needs. After learning one set of language conventions, the user can accomplish tasks in any of the areas mentioned. Today the name "Statistical Analysis System" is something of a misnomer -- SAS has become much more.

With such rapid expansion in the software tools available in SAS, perhaps many local users are unaware of the full range of SAS features and their implementation on the CSO IBM system. This article presents a survey of the SAS services currently available to CSO users and the short-term future directions of SAS services. If you have any questions or suggestions regarding SAS or the information presented,,please discuss them with Greg Kesner, the CSO SAS Coordinator.

SAS contains a core system of procedures or PROCs for data management, systems programming, and statistical analysis called the SAS Base that are managed by a supervisory program called the SAS Supervisor. A computer services installation like CSO may license one or more additional sub-libraries of PROCs for certain specific applications.

SAS can operate under a number of IBM operating systems. CSO runs two of these operating systems on its IBM computers: OS/MVT and CMS.

OS/MVT is the batch operating system that local SAS users have been using for years to run their SAS programs. Running SAS in batch requires that all your SAS programming statements be stored in a file (locally this is usually on the CYBER) and submitted to the OS/MVT operating system. The job then runs on the IBM and the output listing is either sent back to the CYBER FETCH queue (a holding area for output files) or it is printed at public computer sites like DCL, Lincoln Hall, or Commerce West.

SAS can be run under CMS (IBM timesharing) in several modes. The user can interactively enter the SAS statements for a SAS DATA or PROC step at a timesharing terminal and then execute just that step. After examining the output, the user may enter statements for the next SAS job step and then have them executed. This process can be repeated with variations until all the programming tasks are accomplished. This interactive mode of SAS processing offers great flexibility in the way a data or program analyst accomplishes his work. Alternatively, the researcher may store all the SAS programming statements in a CMS file using the CMS interactive editor (XEDIT), then run the SAS job at the timesharing terminal, and then examine the results that may be stored in another CMS file or displayed on the terminal screen. This is referred to as non-interactive SAS processing.

Another SAS processing option under CMS is for the user to submit the SAS program as a batch job to be run under OS/MVT. The output listing can be routed back to the user's CMS disk storage or to a line printer. Whatever the processing mode selected, CMS is a much more flexible system than OS/MVT for working with SAS.

For details on acquiring a CMS logon for access to the IBM timesharing system, please see the "Introducing IBM Timesharing -- CMS" article in this issue.

Not all sub-libraries of SAS procedures are available for all operating systems. And even if they are available, a computer services installation may choose to not have all of them available under all the operating systems. Locally only the SAS Base product is offered under OS/MVT. Under CMS we offer the SAS Base product (CMS/SAS), SAS/GRAPH, and SAS/FSP. SAS/GRAPH is a collection of SAS PROCs for high-resolution color graphics on a wide range of graphics devices. SAS/FSP is a sub-library of PROCs that enable the interactive editing of SAS data sets in a full-screen or simulated full-screen environment. ("Full-screen" refers to the primary type of timesharing terminal used with CMS. Most terminals around campus are called "ASCII typewriter" terminals because the user can only work on one "active" line at a time. On a full-screen or software simulated full-screen terminal, the user can edit or use any line on the entire screen. CSO has a software tool called SIM3278 that simulates an ASCII typewriter terminal as a full-screen terminal. This enables campus users of most non-full-screen terminals to use all the CMS and SAS full-screen software products.)

With the SAS data set editing capabilities of SAS/FSP, users can also design entire screens for interactive data entry or display. For example, a researcher might design a screen as a facsimile of a lab report form and have the lab assistants enter the data directly into a SAS data set for subsequent analysis. The screen can be designed in such a way as to have SAS check for the appropriateness of the data entered in the fields. This can help expedite or even eliminate the steps of raw data coding and data cleaning in data processing. SAS/FSP also enables SAS users to store business and form letters in SAS data sets for processing with SAS data sets containing names and addresses from mailing lists. And in the 82.3 version of CMS/SAS/FSP which is expected to arrive in a few months, SAS users will find a full-screen facility for spread-sheet processing of financial data.

CSO will also be providing SAS/ETS with the 82.3 version of CMS/SAS. SAS/ETS is a collection of procedures for analyzing econometric and time-series data, systems modeling, and financial report writing.

In considering all these features, it is important to remember that all the procedures in all the sub-libraries are integrated into one system. This means that a user can interactively enter data into a SAS data set using SAS/FSP, analyze the data using procedures in the SAS Base or SAS/ETS, and then use SAS/GRAPH to plot results or other procedure output. The possibilities seem almost limitless, especially in the IBM timesharing environment of CMS. Coupling the full-screen interactive features of CMS with the MACRO Language Facility in CMS/SAS 82.3, users will be able to design their own interactive SAS systems for a vast array of applications in office management, data management and analysis, education, and information reporting.

With all this available under CMS, you can understand why we are encouraging all current SAS users to begin using SAS under CMS! We also urge new SAS users to learn SAS and perform their SAS processing under CMS. Our plan for future SAS services is to continue to offer only the SAS Base under OS/MVT and to offer the complete set of SAS sub-libraries under CMS. This includes the newest sub-library of SAS procedures for operations research called SAS/OR which should be released late this year. Truly, now is an excellent time for SAS users to make the transition to IBM timesharing.

SAS User Education Services Offered by CSO

With such a variety of software tools available in SAS, CSO considers quality SAS user education to be a major objective. In reaching this goal, CSO offers a variety of options:

1. Self-paced SAS Instruction

   CSO makes available the full library of SAS Video Training Courses produced by SAS Institute. These are:

   - SAS Basics 100-Series Video Training Course
   - SAS Color Graphics 100-Series Video Training Course

   There is no charge for using these courses on an individual basis. Both video training series are available in the Media Center of the Undergraduate Library. You may check them out and view them there. We also sell the training workbooks for both series at the CSO Distribution Office (1208 W. Springfield, Urbana).

2. CSO SAS Short-courses and Seminars

   Each academic semester, CSO offers a number of short-courses and seminars on many of the SAS features offered on our IBM system. At present, these are introductory courses for the SAS Base product, CMS/SAS, CMS/SAS/FSP and CMS/SAS/GRAPH. The course offerings will most likely expand as more SAS features become available on the system. Nearly all are focused on using SAS in the IBM timesharing environment. In addition to the short-courses, we also offer periodic seminars to keep users current on developments in SAS services. Up to this point these seminars have also focused on SAS usage under CMS. Announcements for short-courses are published at the beginning of each term.

3. The Local SAS Users Mailing List

   The SAS users list is a means of communicating news to SAS users. Periodically SAS users will receive mailings announcing seminars, courses, or changes and enhancements to SAS. If you are not already on the mailing list, see the following article "SAS Users -- Please Tell Us Who You Are" for how to be included on the list.

4. SAS Documentation Available

   CSO sells most of the major documentation produced by SAS Institute at the CSO Distribution Office (1208 W. Springfield). We also have copies of the documentation available for your examination and use at both the CSO South Consulting Office (85 Commerce West) and in the Closed Reserves section of the Undergraduate Library. This documentation includes the reference manuals and the SAS Technical Reports on special topics (The technical reports are not currently stocked for sale by CSO; you may order them directly from SAS Institute.) Also at the South Consulting Office and the Undergraduate Library is "SAS Communications", the quarterly user magazine published by SAS Institute.

CSO SAS Consulting

Users with questions regarding CSO SAS services, their own SAS programs, or SAS programming applications to be run on the CSO system may contact the consultant on duty in the CSO South Consulting Office (85 Commerce West). The office hours are from 9:00-5:00pm Monday through Friday. Questions on research statistical design should be referred to the Mathematical and Statistical Consulting Committee (MSCC) of the Mathematics Department Statistics Division (221 Altgeld Hall; 333-2167).

SAS has evolved into a powerful system of programming languages and information processing pro cedures useful in an extensive range of applications. CSO is dedicated to providing you with quality SAS services in a computing environment which facilitates your research and educational applications. We welcome your suggestions on maintaining this service and hope you find using SAS both productive and enjoyable!

SAS USERS -- PLEASE TELL US WHO YOU ARE!

We are continuing to compile a list of local SAS users which we will use to notify SAS users of changes or enhancements to SAS, SAS training seminars, and new information we receive from SAS Institute. If you have not already contacted us and would like to be added to the list, please specify:

University ID#,
Office Telephone Number,
Name,
Staff Position/Title
Campus address (please include department)

in a message to Greg Kesner via the CYBER Message Facility by entering:

TELL,UN=KESNER

or by mailing the information to:

Greg Kesner
150 DCL (CSO)
Campus

INTRODUCING IBM TIMESHARING -- CMS

Since January of this year, IBM timesharing has been available to the general user community. CMS (Conversational Monitor System) is the acronym for this operating system that brings to users, requiring or desiring IBM processing, many enhanced benefits currently experienced by CYBER timesharing users. Prior to this time, for example, local SAS users were only able to use SAS from the CYBERs via batch processing under the OS/MVT operating system on the IBM. Now they are experiencing enhanced productivity with an expanding set of SAS products under CMS. This semester, SPSS users will be able to use the most recent releases of SPSS-X, SPSS, SPSS/GRAPHICS, and SCSS (the SPSS Conversational System) in IBM timesharing.

Distinctive Features of CMS

CMS is a timesharing system that runs under the supervision of the main operating system of the IBM 4341 called the "Virtual Machine System Product Control Program" (referred to as "VM/SP CP" or often simply called "CP"). The batch system under which IBM users have run their applications programs for many years is called OS/MVT and runs concurrently with CMS under the CP operating system.

CMS enables users to work interactively with software packages and language processors on the IBM much like the CYBER timesharing system. However, the IBM timesharing system differs from the CYBER system in some very significant ways.

Virtual Memory

The Virtual Machine (VM) operating system on the IBM manages program execution in a way allowing execution of programs requiring more central memory than is physically available in the computer. The concept behind this system management is called "virtual memory". This is in contrast to "real memory" which refers to the amount of memory that actually exists in the computer hardware for program execution. The virtual memory operating system makes a computer of finite memory resources operate as if it had much greater capacity. Some users have already taken advantage of this when their programs have exceeded the upper limit for central memory on the CYBERs.

The Virtual Machine

Another conceptual distinction of IBM timesharing is the "virtual machine". A virtual machine is essentially the CP operating system supplying you with your own computer. Actually the VM operating system simulates the features of a computer as if it was your personal machine. Your computing machine is "virtually" there, but not physically -- hence the name "virtual machine". When you obtain a CMS logon to the IBM system, an entry is made in the CP Directory of users. Whenever you log on to the IBM, CP recognizes you from the directory and links you to your file storage disk space which is called your mini-disk. (A mini-disk is essentially your section of storage space on a physical magnetic disk which CP simulates as your own private disk for storing your CMS files.) Later in the logon process, you enter a "PROFILE" command which instructs CP to execute a series of commands that completes the configuring of your virtual machine. This simulation process of generating your virtual machine includes establishing your virtual reader, virtual punch, and virtual printer, as well as setting up links to various system disks that contain software tools you need to perform your work. The timesharing terminal you use is considered your virtual console. During this process, CP also starts CMS "running" on your virtual machine. This configuration is completed in a matter of seconds after which you can begin working with you virtual machine.

CMS Processing Environments

Another conceptual distinction of IBM timesharing is that of "levels of processing environments". A processing environment is a context created for you by a program running on the computer which facilitates use of a limited set of computer processing features to accomplish some work. The program which creates the environment may be either supplied by the computer operating system or written by you. A useful analogy in the CYBER system is using the ICE text editor to create or edit a file. Here the ICE editor program establishes the processing environment in which you can work with a file of information. To this point, the concept of processing environments is similar on the two systems. But an integral part of IBM timesharing is the additional concept of nested levels of processing environments in and out of which a user can move. In the CYBER analogy, this would be somewhat like entering ICE to edit a file, and then being able to exit ICE for awhile to enter another environment like an on-line help facility, or a subset of the timesharing system, or other processing environments. On the IBM the procedure could be something like the follwing. First a user might enter interactive SAS to create a SAS data set. Then she might enter the CMS subset operating system to examine some files she will need a little later in her interactive SAS session. While in the CMS SUBSET, she could also use the CMS editor (XEDIT) to create a SAS program file to be included in her SAS session. While within XEDIT, she might need to look up the details of an XEDIT command and therefore enters the CMS Help Facility. No matter how deeply into the levels of processing environments she goes, she can

simply enter the return command to get back to the point she left in the level she entered from. This is all a bit difficult to describe in words and a little bit easier to relate using diagrams (see p.17 of the IBM document, "CMS User's Guide" for such a diagram). But you will find this facility for nested levels of processing environments very powerful and productive to use as you get more familiar with it.

Full-screen Processing

Another distinctive feature of CMS is the capability to work with full screens of information rather than a single line at a time. This is called "full-screen" processing versus "typewriter" processing. Special full-screen timesharing terminals are marketed to fully utilize the power of full-screen processing. Later this fall, some of these full-screen terminals for color graphics will become available in the Turner Hall computer site. Most of the public timesharing terminals on campus are called "ASCII typewriter" terminals because the screen scrolls much like a typewriter allowing you to only work with one line at a time. However, CSO has installed a software interface (SIM3278) on the IBM that simulates a typewriter terminal as a full-screen terminal allowing use of the full-screen features of CMS.

In full-screen processing, the cursor can be moved anywhere on the screen by means of cursor control keys. Data can be entered or overwritten anywhere on the screen (except designated protected fields) in any sequence. The user works with an entire screen of information and then presses the "enter" key to send the modified screen of information to the IBM for processing. Or the user can press another key to see the next screen or a previous screen of information. The CMS system has a number of software tools that take advantage of the conveniences and power of full-screen processing.

Programmed Function Keys

Another feature of full-screen or simulated full-screen terminals is program function keys (PF keys) which allow certain keys on the terminal keyset to be programmed with certain commands. Instead of having to enter an entire command repeatedly, the PF key programmed with that function is simply pressed and the command is executed. The PF key functions can be changed at any time affording great flexibility in the various CMS processing environments. Users will notice their computing productivity increase dramatically as a few key presses replace a series of command entries. There is even a PF key function to retrieve the last few commands entered!

All of this serves to introduce you to some of the major concepts and features of IBM timesharing; but how do you get started?

Getting Started with CMS

You need a CMS "logon" to begin working with CMS. You can apply for your logon at the CSO Accounting Office (1208 W. Springfield, Urbana). Presently the CMS accounting system does not support SARA accounts -- this support is in development and we hope to have it ready soon. Until then, only PS numbers funded by Research Board or hard-money contract accounts can be used to set up a CMS account. (We will announce when SARA accounts are supported under CMS.) Your logon will be ready for you to use within twenty-four hours of the time you submit your application with a valid PS number. From that point you will need documentation to learn more about how to use CMS.

CMS Documentation

We have produced a number of documents introducing features of the IBM timesharing system and plan to write more in the months to come. You can obtain a set of these locally written documents from either the CSO South Consulting Office (85 Commerce West) or the CSO North Consulting Office (1208 W. Springfield). A set will also be available in the Closed Reserves section of the Undergraduate Library. One of the documents in the set is a list of helpful IBM documents for CMS users. These reference manuals are available for your examination and use at both the CSO consulting offices and the Closed Reserves Section of the Undergraduate Library. They are also on sale at the CSO Distribution Office (1208 W. Springfield). CMS also has an excellent on-line Help Facility, the usage of which is documented in Part 4 of the "CMS User's Guide". A locally written software tool called WRITEUP is available to obtain a hard-copy listing of any Help Facility subject. A one-page document in the set of local CMS documentation describes how to use WRITEUP under CMS. (For on-line help on WRITEUP, enter: HELP CSO WRITEUP when logged on to CMS.)

What is Available Under CMS

The following software processors are available with CMS:

SAS	Statistical Analysis System
SAS/GRAPH	SAS Graphics sub-library
SCSS	SPSS Conversational System
SPSS-9	SPSS Release 9
SPSS-X	SPSS Extended
SPSS/GRAPHICS	SPSS Graphics under SPSS-9
VS/FORTRAN	FORTRAN 77

The following software processors are in development and are expected to be available for general use by January, 1984:

PASCAL
PLC
PL/I (Optimizing Complier)
WATFIV

If you have questions or comments regarding the IBM timesharing system or services, please contact:

Campus Mail:
Greg Kesner
CMS User Services Coordinator
150 DCL (CSO)
Campus
Telephone: 333-4668

On the CYBER:

TELL,UN=KESNER

On the IBM/CMS:

NOTE KESNER

SENDING FILES TO CMS FROM THE CYBER

One or more files can be sent to CMS (IBM timesharing) from the CYBER using the CYBER PRINT command. (See the *CSO Local Utilities Manual* or Reference Guide RF-7.16 for more details on the PRINT command.) The PRINT command sends the file as a listing to be "printed" to the virtual reader of your CMS virtual machine. You must then read the file from your virtual reader into a CMS file and then edit the CMS file to remove the IBM burst-page lines added to the beginning and end of the file. Also, the logical record length (LRECL) of the file must be reduced from the 132-byte (character) LRECL of a listing file to a length of 80 bytes which is the required LRECL of a program file to be use with a processor like SAS.

The following procedure will accomplish this. An example is discussed which involves sending one CYBER file named SASCODE to the virtual reader of CMS userid KESNER. In this example, the lower case text is what you must provide.

I. ON THE CYBER:

1. Print the file SASCODE to CMS using the following command:

   PRINT,filename/J=userid/FORMS=CMS

   where ***filename*** is the CYBER file name and ***userid*** is the CMS userid. For the example, the following command would be used:

   ```
   PRINT,SASCODE/J=KESNER/FORMS=CMS
   ```

   Notes:

   a. Notice the ***J=*** option specifies the CMS userid of the virtual reader to which the file should be sent. (The term "userid" is analogous to "logon name".)

   b. If the file contains lower case text, be sure to use the "/AS" option on the PRINT command.
      (For example: PRINT,SASCODE/J=KESNER/FORMS=CMS/AS)

II. ON THE IBM (CMS)

1. Enter the command: ***QUERY READER ALL***
   This determines if the print file is on your virtual reader. The output displayed will look like the following when the file is on the reader:

   ```
   ORIGINID FILE  CLASS  RECORDS  CPY  HOLD  DATE   TIME      NAME    TYPE    DIST
   MVT      3380  1 PRT  00000063 001  NONE  08/12  08:40:39  KESNER  MVT594  MVT
   R; T=0.01/0.02 08:41:16
   ```

2. Enter the command: ***READ fn ft fm***
   where: ***fn*** is the name of the CMS file in which you wish to store the contents of the print file on the virtual reader; ***ft*** is the filetype of the CMS file; and ***fm*** is the filemode of the CMS file. (For a discussion of CMS file identifiers, see p. 43 of "CMS User's Guide". For more details on the READ command, see the "CMS Command and Macro Reference" or enter HELP READ when logged onto CMS.)

In the example below, the contents of the SASCODE CYBER file will be stored in the CMS file identified by a filename of SASCODE, a filetype of SAS, and a filemode of A:

```
READ SASCODE SAS A
```

The response by CMS displayed at the terminal will be:

```
DMSRDC738I RECORD LENGTH IS '132' BYTES.
R; T=0.08/0.14 08:42:22
```

Note:

a. The print file on the reader will have line-printer carriage control because it was "PRINTed" from the CYBER. The READ command strips off the carriage control column from a print file as it reads it off the virtual reader. (If you want to preserve this carriage control column when moving the file off the reader, use the DEPRINT command instead of READ. Ordinarily, however, you will not want to keep this extra carriage control column added to a file printed from the CYBER.)

3. Now the file from the CYBER is one of your permanent CMS files. You will next need to edit the CMS file using XEDIT to remove extraneous lines added during the printing process. (For information on specific XEDIT commands, see the "System Product Editor Command and Macro Reference" or enter 'HELP XEDIT command-name' when you are logged onto CMS. For example, for help on the DELETE command, enter HELP XEDIT DELETE .)

   a. In general enter: ***XEDIT fn ft fm***
   (For the example, XEDIT SASCODE SAS A would be used.)

   b. Use the DELETE command in XEDIT to delete the IBM burst-page lines from the beginning and end of the file. (Use the LOCATE command to find the beginning and end of the text which is the actual contents of the file you sent from the CYBER.)

   c. Enter: ***SET LRECL 80***
   This causes the file to be truncated after column 80 when the FILE command is issued.

   d. Enter: ***FILE***
   This command saves the changes made to the file and exits from XEDIT.

The file is now ready to be used by a CMS processor like SAS.

SIM3278 FOR IBM TIMESHARING

What is SIM3278?

SIM3278 is a program that allows ASCII video display terminals to function as simulated IBM 3270-type terminals (3277/3278). SIM3278 is usually referred to as a 3270 emulator or as an ASCII-to-3270 protocol converter. The SIM3278 program will currently function on any IBM VM/370 computer or its equivalent.

As most users are aware, ASCII terminals are limited to functioning only in a line-by-line mode. The ASCII terminal writes one line of text at the bottom of the screen and then scrolls the screen upwards

As most users are aware, ASCII terminals are limited to functioning only in a line-by-line mode. The ASCII terminal writes one line of text at the bottom of the screen and then scrolls the screen upwards before another line is written. This line-by-line mode makes the display and editing of text on the ASCII screen time-consuming.

3270-type terminals, on the other hand, display a full screen of text and wait for the user to clear the screen (by hitting a key) before displaying the next screen. Editing a file on a 3270 screen is equally convenient. The user can study a full screen of text and by using the cursor keys move to any displayed line and change the text by overtyping. Similarly, the user benefits from a great number of full screen software tools that have been written for 3270-type terminals. Essentially, the SIM3278 program provides the user of an ASCII terminal with all the advantages of a "smart" 3270-type terminal.

How Does SIM3278 Work?

SIM3278 is essentially a protocol converter between the VM Control Program (CP) and the ASCII terminal. The SIM3278 program resides on the computer where it waits for input from both the ASCII terminal and CP. When a user connects to SIM3278 from a terminal, CP initializes a logical 3270 screen that it associates with that terminal. CP can read and write data to this logical 3270 screen thinking the user is really on a 3270-type terminal. SIM3278 can detect any changes made to the logical 3270 screen and when this happens the data in that screen is converted to addresses and written to the real ASCII terminal screen. The user can make any desired changes to the real ASCII screen using the cursor keys and overtyping. Now by hitting the carriage return key all changes are sent to SIM3278. SIM3278 writes these changes to the logical 3270 screen which tells CP that the terminal screen has been updated and therefore completing the simulation process.

How Does a Person Use SIM3278?

There are several ways to have a terminal connected to the IBM. Usually it is either directly via the SWITCH or telephone dial-up through the SWITCH. Once communication has been established with the IBM, you are prompted from the system by a period (.). Ordinarily, you begin your session by entering the LOGON command. However, to use SIM3278 you enter two commands prior to entering the LOGON command.

The following sequence details this process:

1. System Prompts: . (period)

2. User Enters: DIAL SIM <CR>

3. System Response: DIALED TO SIM 099
   ENTER TERMINAL ID NUMBER IF KNOWN ELSE TYPE "?":

4. User Enters: 34

   NOTE: It is important to select the correct terminal type due to the different ways in which terminals handle cursor addressing. (For example, in the example the user selected terminal number "34", which is for a "MIME in ACT4 Mode". This information can be obtained by entering a question mark at the terminal number prompt.) The user must select the correct terminal id number to prevent "garbage" being written to the terminal screen. (If you do enter the wrong number, WAIT UNTIL THE GARBAGE OUTPUT STOPS; then hit the <RETURN>-key; and then enter: LOGOFF. This will drop your terminal from the simulator and you can begin the process again.)

5. System Response: After you have entered the terminal number, The full screen VM/370 LOGO appears.

6. User Enters: <CR>

7. System Response: The screen clears and a 'CP READ' appears in the bottom right hand corner of the screen.

8. User Enters: LOGON <userid>

(Follow the standard logon procedure from this point.)

The user is now logged onto the system in the full screen mode.

Where Can You Get More Documentation on SIM3278?

A complete guide for SIM3278 is contained in the file: SIMGUIDE CSO which resides on one of the system mini-disks. This guide includes appendices detailing the terminal types supported by SIM3278 and their features. To determine which disk, enter:

LISTFILE SIMGUIDE CSO *

The system will respond with a message like:

SIMGUIDE CSO $

where the $ will be the mode letter of the disk.

To get a hard-copy listing of the guide, enter:

RPRINT SIMGUIDE CSO $

substituting the proper mode letter of the disk on which the file resides for the $. (For example, if the file resides on the disk with mode F, the reply from the LISTFILE command would be similar to: SIMGUIDE CSO F2. You would then enter: RPRINT SIMGUIDE CSO F to obtain a listing.)

Special note: The SIMGUIDE CSO file has carriage control characters in column one. Be sure to specify CC and EJ when prompted by RPRINT.

Finally, most public terminals around the campus should have a label on them identifying what type of terminal they are, whether the terminal type is supported by SIM3278, and the SIM3278 terminal identification number. Also, terminals will have a special label near the terminal settings switches identifying the proper settings for the terminal to be used with SIM3278. Ordinarily these settings should not be altered, but if the terminal you wish to use is not working properly with SIM3278, you should make certain these settings are correct.

MATHEMATICAL SERVICES

CONVERSION OF SOME CYBER PACKAGES TO FTN5

In the June issue of *OFF-LINE*, we announced that a number of CYBER packages will be converted to FTN5, and gave a target date of

Monday, January 9, 1984

for the final conversion date. Prior to that date, FTN5 versions will be available via the FUTURE and GRAB/F commands for user testing. After that date, the old FTN versions will not be available except by special arrangement.

In the July and August issues, the first groups of products to be converted were announced. In this issue, we announce the availability of the MSL library in an FTN5 version. The full group of products now available in FTN5 versions via GRAB/F and FUTURE is

IMSL	SLAM	FORSIM	GRG	FLEX	GMET
XMP	UOILIB	LINPACK	EISPACK	MINPACK	ITPACK
NETWORK	LSODE	AMOSLIB	GASP	MSL	

More products will be announced in the coming months.

You are strongly encouraged to begin converting your FTN programs to FTN5 and to use the FTN5 versions of these products whenever possible. The FTN compiler, though reliable, is no longer supported by CDC; FTN5 is their "official" FORTRAN compiler now, and represents the new standard for the FORTRAN language. In deference to this, CSO would like to encourage the eventual "fading away" of the old compiler. At some point in the (not too near) future, we plan to alter the system so that the FTN control statement will access the new compiler, and so that FTN4 will access the old one.

Manuals for FTN5 can be purchased at the CSO Distribution Office, 1208 W. Springfield. A utility, F45, is available for doing automatic conversions of some types of FTN statements to equivalent FTN5 forms. The manual for this also can be purchased at the Distribution Office; a copy of the manual is available for inspection at the CSO Systems Consulting Office at 1208 W. Springfield.

FEATURE ARTICLES

FORTRAN FUTURES

This article is part of a series on the work of the FORTRAN Standards Committee X3J3 in producing the next revision of the FORTRAN standard. The reader is reminded that the features described in this article are not a part of any FORTRAN compiler currently available, but rather are proposed requirements for FORTRAN processors produced in the late 1980's through the mid 1990's. Although every effort has been made to accurately describe the current position of X3J3 on these matters, the development of a revision to the FORTRAN standard is an evolutionary process, and these proposals may be subject to refinement, revision, or even retraction. Comments on these proposals may be given to Kurt Hirchert of the CSO Systems Consulting staff, who is a member of X3J3.

The 87th meeting of X3J3 was held August 8-12, 1983 in Los Alamos, New Mexico. Activity at that

meeting included the following:

- The largest task undertaken at this meeting was the review and editing of a draft Fortran technical information bulletin (or FIB) presenting the current status of the committee's work on the next revision of the Fortran standard. It is expected that this FIB will reach a much wider audience than the channels normally used to publicize the committee's actions.

- Additional work was done on interpretations of the 1978 standard. It is expected that the collected interpretations will also soon be published as a FIB.

- The BUNDLE subprogram was renamed a MODULE.

- A facility was adopted which allows one to redefine assignment for data structures (now being called derived data types).

- Since the use the period as the structure qualification character leads to possible ambiguities relative to operators such as .EQ., the percent sign (%) has, at least temporarily, taken its place. Should an acceptable resolution to the ambiguities problem be found, the use of period may be restored.

- The * and -* notations for referencing array sections were dropped, since the same sections could also be expressed with the more general triplet notation.

- Because of problems with the previously passed facility for user-defined elemental functions, this facility was retracted. It is unclear whether it is likely to be reinstated if a corrected proposal is brought forth.

- The array facilities were regularized to allow zero sized arrays (i.e., arrays in which there are no legal subscript values for one or more of the subscript positions).

- A change was made in the syntax used to specify character variables, functions, etc. in order to make it more like the syntax proposed for other data types such as REAL.

- A facility was added allowing one to determine the number of values actually read or written by an input/output statement.

- The committee formally switched from working with X3J3/S6, a document describing only the new features in the next Fortran standard, to X3J3/S7, a document describing the entire language.

- The committee heard presentations by employees of the Los Alamos National Laboratory on requirements for supporting very large Fortran programs.

- Event handling facilities were discussed further. Included in these discussions was the presentation of an alternative approach to exception handling.

- In informal discussion, the committee decided that the concept of "data type" should not include array properties such as rank or size.

- Informally, the committee agreed to change from talking about data FORMs to data TYPEs.

- A proposal for providing aliases for variables, parts of variables or arrays, etc. was reviewed with no formal action taken.

- Possible solutions to a number of small problems relating to input/output or program source form were discussed.

DOCUMENTATION

NEW AND REVISED DOCUMENTATION

CSO now has an *Addendum to an Index to Software on the CYBER Systems* available, free, at the CSO Distribution Office, 1208 W. Springfield. This addendum contains listings of new software packages which have been installed on the CYBER systems since the date of last publication of the manual (December 1981). In addition, there are a few corrected/revised pages from that edition. A completely revised manual, incorporating this addendum will be printed in 1984.

Revised Reference Guides

Following is a list of revised Reference Guides now available at the RJE sites. We recommend that you pick up copies of the revised guides.

RF-1.2	CYBER Rates	Revised 06/29/83
RF-11.1	ID Card	Revised 07/20/83
RF-11.2	4341 Rates/Charge Estimator for MVT Under VM	06/29/83
RF-7.31	PLOT	07/20/83

MISCELLANEOUS

STATISTICAL HELP

The Mathematical and Statistical Consulting Committee (MSCC) provides assistance to members of the University Community on statistical and mathematical problems in their research work. The 1982-83 director of MSCC is John I. Marden who is assisted by graduate statistics students Kenton Juhlin, Bruce Richardson, and Tom Grzesiak.

The service is offered free for problems that can be settled within several hours. To use the service drop in at 221 Altgeld Hall or call 333-2167 during the following times:

Mon & Wed	10-12 and 1-5
Tues & Thurs	1-5

At other times call above number to make an appointment.

Problems that require a considerable amount of consulting time and effort are also welcome. In such cases, the consultee may be asked to contribute financially into an account that has been set up recently to help defray the cost of running the MSCC. Faculty members anticipating the need of substantial

statistical help (for themselves or for their students) are urged to try to obtain funding for this purpose, for instance by including a budget item for statistical consulting in a research proposal.

SALES - EXCHANGES - HELP WANTED

APPLICATIONS OF COMPUTER SCIENCE TO DECISION SCIENCE

Anyone at the University of Illinois who has developed programs or knows of available programs designed to do decision-making algorithms should contact Professor Stuart Nagel, 361 Lincoln Hall, 359-8541. He is interested in programs that do benefit-cost analysis, decision theory, optimum level analysis, allocation theory, and time optimizing models, especially if they are likely to have public policy applications. He is also interested in variations on such programs for microcomputers that can be used interactively by one or more decision-makers simultaneously. Perhaps a group of campus people who share this type of interest could be formed.

INDEX

1982-83 ARTICLES OF INTEREST TO READERS

In the yearly index issue we usually reprint various articles that we feel would be of interest (or importance) to our readers. However, since the last year has been so full of articles discussing new software, important changes to existing software, rate changes, etc., we would have to publish a "small book" to reprint all of the recommended articles. We have compiled a list of articles we believe are important; however, we strongly recommend that you peruse the entire index listing. If anyone needs a particular back issue, contact Lynn Bilger, 120 DCL (333-6236).

- Important rate changes were discussed in the following articles: "IBM Rates to Change" (September 1982), "Service Unit Price Reduced" (October 1982), "New Rates for Plotting" (June 1983), and "Rate Changes" (August 1983). A related article was entitled "Overhead Charges" (June 1983).

- Persons using 1200-baud dial-up should have read the article "Change and Expansion in 1200-baud Dial-up Service" (May 1983).

- Anyone using tapes at the U of I should have read the articles "Temporary Tape Storage" (December 1982), "Two IBM Tape Drives Upgraded to 6250/1600 BPI" (January-February 1983), and "Tape Services at CSO" (June 1983).

- With the addition of more computers, CSO had to develop a new naming scheme for the various machines. We strongly urge users to read the article "Machine Names" (April 1983) and to begin using these new names.

- Articles of particular importance to CYBER users included: "New Product Set Installed on System" (January-February 1983), and three articles entitled "Conversion of Some CYBER

Packages to FTN5" (June 1983, July 1983, August 1983).

- CYBER users may also be interested in the articles "New Subroutines for Full ASCII Input and Output to Terminals" (April 1983), the PROM burning and erasing service announced in "New and Updated Services" (January-February 1983), "Software from National Bureau of Standards" (June 1983), and "FORTRAN Programs and Files" (April 1983).

- Articles about new and revised CYBER software packages included "UWRIM Installed" (July 1983) and "SIR/DBMS 2.1 Available on CYBER 175" (January-February 1983), respectively. Updates to CYBER utilities included: "New Version of P Command" and "MESSAGE Facility Enhancement" (January-February 1983).

- File transfer articles included "File Transfer Service" (October 1982) which dealt with floppy disk and DEC tape transfer to and from the CYBER, and "New and Updated Services" (January-February 1983) part of which dealt with LSI-11 file transfer.

- DI-3000 Plotting articles appearing in the January-February 1983 issue included: "DI-3000 Graphics Software", "GRAFMAKER", "The DI-3000 Metafile Facility", and "PVI Contouring System" (also in March 1983).

- CSO is developing a campus computing network and articles about the network and related topics which appeared in the January-February 1983 issue included: "SYTEK Campus Network" (an update appeared in June 1983), and "Dedicated Lines for Private Facilities." In addition, CSO is developing a microcomputer (personal computer) laboratory site as discussed in "Progress Report - Microcomputer Site" (August 1983).

- Statistical software updates on the CYBER included: "SPSS-CDC Release 8.3 Available" (October 1982) and "BMDP 1981 Release Available" (November 1982).

- Mathematical software updates on the CYBER included: "SLAM II Version 1.0 Release 5.0" (December 1982), "ITPACK Version 2C" (January-February 1983), "New Version of GRG" (January-February 1983), and "Updates to UOILIB" (April 1983; also see May 1983).

- Articles about new mathematical software included: "The LINDO Linear Programming Package" (November 1982; also see LINDO article in January-February 1983), "The SIMSCRIPT Simulation Package" (November 1982; also see SIMSCRIPT article in March 1983), "MATLAB -- a Matrix Desk Calculator" (March 1983), "The CAYLEY Group Theory Language" (March 1983), and "The PROTRAN Problem-Solving Language" (January-February 1983) and related articles (March, April, May, June and August 1983).

- Articles of particular interest to IBM users included: "Upgrade of IBM Equipment" (September 1982), and "SAS Videotapes Available" (January-February 1983).

- Articles about software upgrades on the IBM included: "SAS Release 79.6 Installed" (October 1982), "SPSSX on the IBM" (April 1983), and "OS/SAS82.2: New Release Available on IBM" (May 1983).

- Articles about new software on the IBM included: "CMS/SAS ad CMS/SAS/GRAPH Available" (January-February 1983), and "SAS Sample Library Available" (June 1983).

1982-83 INDEX

POLICY

CYBER

IBM

STATISTICAL SERVICES

MATHEMATICAL SERVICES

PLOTTING SERVICES

NETWORK SERVICES

MINICOMPUTERS, MICROCOMPUTERS, PERSONAL COMPUTERS

FEATURE ARTICLES

DOCUMENTATION

MISCELLANEOUS

OFF-LINE's Mailing List

If you wish to be placed on our mailing list for future issues of *OFF-LINE*, if you wish to be removed from the list, or if you wish to enter an address correction, please complete and return this page. **(Current subscribers are kept on the mailing list until a specific request for removal is received, or until a mailing is returned as undeliverable.)**

Check one:

- ☐ Place my name on mailing list
- ☐ Make the following corrections or changes
- ☐ Delete my name from mailing list

_____

First name -- Initial -- Last Name

Campus Address:

_____

Room - Building

_____

Department

Off-campus Address:

_____

Organization or Company (if applicable)

_____

Street Address

_____

City -- State -- Zip Code

If address correction, give old address and zip code below.

_____

_____

_____

9/83

SEND TO:

OFF-LINE
150 Digital Computer Laboratory
University of Illinois at Urbana-Champaign
1304 West Springfield Avenue
Urbana, Illinois 61801

Computing Services Office

Off-Line

University of Illinois at Urbana-Champaign VOL. 11, NO. 10 October 1983

EDITOR: Lynn Bilger
PHONE: (217) 333-6236
120 Digital Computer Lab
1304 W. Springfield Ave.
Urbana, Illinois 61801

Page *Contents*

CSO DIRECTORY - STAFF AND SERVICES

Administrative

Director	George Badger	150	DCL	333-4103
Business Manager	Stanley Rankin	150	DCL	333-6530
Secretary	Joyce McCabe	150	DCL	333-1637

User Services and Hardware/Software Support

User Accounting		**1208**	**W Springfield**	**333-7752**
Distribution Office		**1208**	**W Springfield**	**333-6760**
Systems Consulting		**1208**	**W Springfield**	**333-6133**
Statistical Services Consulting		**85**	**Comm West**	**333-2170**
Text Processing Consulting		**118**	**DCL**	**333-7318**
Terminal Repair Service		**150**	**DCL**	333-0969
Dial-up Numbers	**CYBER 175 (NOSA)**	**300**	**baud**	**333-4000**
	CYBER 174 (NOSB)	**300**	**baud**	**333-4004**
	IBM 4341 L10 (VME)	**300**	**baud**	**333-4005**
	IBM 4341 M2 (VMD)	**300**	**baud**	**333-4006**
	Switch			**333-4008**
Asst Dir User Services	Robert Penka	173	DCL	333-4709
Asst Dir Systems and Operations	Sandra Moy	177	DCL	333-4703
Asst Dir Development	J. M. Randal	175	DCL	333-9772
Asst Dir Engineering and Hardware Selection	Cliff Carter	195	DCL	333-3723
Manager Accounting Services	Gary Bouck	1208	W Springfield	333-7752
Manager Statistical Services	Beth Richardson	189	DCL	333-6276
Microcomputer Laboratory	Jack Knott	86	Comm West	333-6562
Documentation	Lynn Bilger	120	DCL	333-6236
CYBER-IBM Operations	Myra Williams Ann Ryczer Rex Duzan	194a	DCL	333-6562
VAX Operations	Debbie Hudson	123	DCL	333-8150
Laboratory Support Project	Mike Gardner	18	DCL	333-7904
RJE Operations North	Sylvia Hansen Chuck Harnsberger	65	ME	333-6285

RJE Sites (see Reference Guide RF-0.3 for operating hours)

Agriculture	N-120	Turner Hall	333-8170
Chemistry	153	Noyes Lab	333-1728
Commerce West	70	Comm West	333-4500
CRH Snack Bar	120	Snack Bar	333-1851
DCL Routing Room	14	DCL	333-7685
Electrical Engineering	146	EEB	333-4936
Florida Ave Res Hall		FAR	333-2695
Illinois St Res Hall		ISR	333-0307
Mechanical Engineering	65	MEB	333-1430
Psychology	453	Psych Bldg.	333-7815
Social Science	202	Lincoln Hall	333-0309

OFF-LINE is the monthly newsletter of the Computing Services Office at the University of Illinois at Urbana-Champaign. Information in this issue is current as of September 22, 1983.

CSO operates a CDC CYBER 175 with 262K words of central memory and a CDC CYBER 174 with 196K words of central memory. The 175 and 174 run under the NOS Operating System and share 512K words of ECS. The 175 serves over 200 simultaneously active text and graphics terminals and the 174 serves over 100 simultaneously active terminals. CSO also operates an IBM 4341 Model M2 with 8 million bytes of memory running HASP-OS/MVT under VM and an IBM 4341 Model L10 with 4 million bytes of memory running VM. In addition, CSO operates a DEC VAX 11/780 with 4 million bytes of memory, running under the UNIX Operating System and driving a GSI CAT-8 phototypesetter.

Operating Hours (see HEARYE,SCHEDUL for exceptions):

	CYBERS 174/175	IBM
M-F	8 am - 6 am	8 am - 6 am
SAT	8 am - Midnight	8 am - 6 am
SUN	Noon - 6 am	Noon - 6 am

IBM

SPSS AVAILABLE IN IBM TIMESHARING

The SPSS statistical package, now available for use in IBM timesharing, offers convenient access to the most up-to-date software available from SPSS Inc. Previously, to use the SPSS software on the IBM you had to do your editing on the CYBER, send your job across tieline to be run in batch mode on the IBM, and then wait for the output to be sent back across tieline and stored in the fetch queue. Now, you can edit your program and data files using the XEDIT editor on IBM/CMS, execute your job in timesharing mode on the IBM, and store your output in an IBM/CMS disk file or route it to the printer. Running SPSS programs under IBM/CMS offers you all the benefits you're accustomed to in CYBER timesharing plus the added advantage of convenient access to both an expanded set of SPSS products and the most recent releases of software available from SPSS Inc.

Running SPSS Programs in a CMS Environment

To run SPSS under CMS you must create an SPSS program file with a filetype of SPSS. NOTE: The program file is the file that contains SPSS program statements such as DATA LIST and CONDESCRIPTIVE. You use the XEDIT editor on CMS to create the program file. Simply type

XEDIT filename SPSS

to create a program file with the filename of your choice and a filetype of SPSS. Since SPSS program statements have a specification field that begins in column 16, it's convenient to use a tab control character when typing the statements in XEDIT. After you initialize XEDIT, issue the command **TABS 1 16** to establish the tab column settings. Then, when typing SPSS program statements in input mode, type the back slash just before typing the specification field; this causes the specification field to be tabbed beginning in column 16. The SPSS processor is invoked and the program is run by using the command:

SPSS filename

where ***filename*** is the name of the SPSS program file. SPSS writes the program results to a display file that has the same filename as the program file, but a filetype of LISTING. Any error messages, generated while running the program, are listed at the terminal as well as being written to the display file. You can view the display file at your terminal by using the command:

TYPE filename LISTING

You can get a copy of the file on paper by using the command:

RPRINT filename LISTING

Several options are available on the SPSS command to redirect the display file. To use any of these options, specify them after a left parenthesis following the program filename, as in:

SPSS filename (options

If you use several options on the SPSS command they are separated by spaces.

The available options are:

ALLTERM	All output is listed at the terminal.
NOTERM	Error messages are not listed at the terminal.
PRINT	Output from SPSS is written to your virtual printer.
SPACE nK	This is the total amount of workspace and transpace available to SPSS. The default is SPACE 80K.

Using a FILEDEF

If this is your first attempt to run SPSS programs in a CMS environment, chances are you've been running SPSS programs either on the CDC CYBER computer or in batch mode on the IBM computer using OS/MVT. The following section explains the control language necessary for running SPSS programs, and shows the corresponding control language for the CYBER, OS/MVT, and CMS systems. Throughout this section the term "filename" stands for the SPSS program file, "fn" stands for the data file to be read or written, "ft" stands for filetype, and "fm" stands for filemode.

Raw Data Input

It's customary for the data to be in a separate file from the SPSS program statements. Typically, you will have many program files that all access the same data file. You may make many changes to the program (e.g., change CONDESCRIPTIVE to FREQUENCIES); however, once you get the data cleaned, it's usually considered sacred and you won't want to include the data in an XEDIT editing session. The control language for running an SPSS program that has raw data input is as follows:

CYBER:
```
GRAB,SPSS
SPSS,I=filename,D=fn,...
```

OS/MVT:
```
// EXEC SPSS
//FT08F001 DD ...
```

CMS:
```
SPSS filename (BCDIN fn
...note the filetype of fn must be SPSS...
        -or-
FILEDEF 8 DISK fn ft fm
SPSS filename
...note the filetype of fn need not be SPSS...
```

Raw Data Output

You would generate raw data ouput from an SPSS program if, for example, you used the WRITE CASES procedure, requested a correlation matrix written out from the PEARSON CORR procedure, or requested factor scores written out from the FACTOR procedure. The control language for running an SPSS program that has raw data output is as follows:

CYBER: GRAB,SPSS
SPSS,I=filename,A=fn,...

OS/MVT: // EXEC SPSS
//FT09F001 DD ...

CMS: SPSS filename (BCDOUT fn
...note the calculated output will be stored...
...with a file specification "fn SPSS A"....
-or-
FILEDEF 9 DISK fn ft fm
SPSS filename
...note the filetype of fn need not be SPSS...

Use of RAW OUTPUT UNIT Statement

You would use the RAW OUTPUT UNIT program statement if you had several procedures in your SPSS program that generated raw data output. The control language for running an SPSS program that uses the RAW OUTPUT UNIT statement is as follows:

CYBER: You can use a filename on the RAW OUTPUT UNIT statement as in: RAW OUTPUT UNIT fn

OS/MVT If you used RAW OUTPUT UNIT 15 then you would specify
// EXEC SPSS
//FT15F001 DD ...

CMS: If you used RAW OUPUT UNIT 15 then you would specify
FILEDEF 15 DISK fn ft fm
SPSS filename

Raw Data Output with Large Record Lengths

Calculated raw data files from SPSS, such as correlation matrices or factor score matrices, are limited (by default) to 80 characters. If you use the WRITE CASES procedure to create a raw data file longer than 80 characters, you *must* specify LRECL and BLOCK on the FILEDEF statement as in:

FILEDEF 9 DISK fn ft fm (LRECL n BLOCK n

where ***n*** is the record length. There is no advantage to specifying a block size larger than the record length because CMS disk files are pre-blocked. In spite of this, ***BLOCK n*** must be present on the FILEDEF command.

CYBER: GRAB,SPSS
SPSS,I=filename,A=fn,RL=n

OS/MVT: // EXEC SPSS
//FT09F001 DD DCB=(LRECL=n,BLKSIZE=n,...

CMS: FILEDEF 9 DISK fn ft fm (LRECL n BLOCK n
SPSS filename

System File Input

An SPSS system file is a specially formatted file containing both data and a dictionary. SPSS expects a system file as input when you use the GET FILE program statement. The control language for running an SPSS program that has system file input is as follows:

```
CYBER:      GRAB,SPSS
            SPSS,I=filename,G=fn,...

OS/MVT:     // EXEC SPSS
            //FT03F001 DD ...

CMS:        FILEDEF 3 DISK fn ft fm
            SPSS filename
```

System File Output

SPSS outputs an SPSS system file when you use the SAVE FILE program statement. The control language for running an SPSS program that has system file output is as follows:

```
CYBER:      GRAB,SPSS
            SPSS,I=filename,S=fn, ...

OS/MVT:     // EXEC SPSS
            //FT04F001 DD ...

CMS:        FILEDEF 4 DISK fn ft fm
            SPSS filename
```

Merge Files

The MERGE FILES procedure creates a unified SPSS system file by merging all, or a subset, of the variables from several SPSS system files. The control language for running an SPSS program that merges two files is as follows:

```
CYBER:      GRAB,SPSS
            SPSS,I=filename,G1=fn1,G2=fn2,...

OS/MVT:     // EXEC SPSS
            //FT03F001 DD ...
            //FT03F002 DD ...

CMS:        FILEDEF FT03F001 DISK fn1 ft fm
            FILEDEF FT03F002 DISK fn2 ft fm
            SPSS filename
```

A final note about FILEDEF is that after executing an SPSS program the file definition is no longer in effect and you need to reissue the FILEDEF command if you want to rerun the program. To prevent this from happening use the "PERM" option on the FILEDEF command, as in:

FILEDEF 8 DISK fn ft fm (PERM

Use of the "PERM" option causes the file definition to be retained for the duration of your terminal session or until you explicitly change it with a new FILEDEF command. The command:

Q FILEDEF

provides a display of all file definitions that are in effect.

Communicating with the CYBER and with OS/MVT

You may have existing data sets on OS disk or CYBER disk that you want to use in an SPSS-CMS program,or you may want to submit an SPSS job to OS/MVT for execution in batch mode. The following sections explain how to accomplish this.

Use an OS Created Dataset in an SPSS-CMS Program

You may have raw data files or SPSS system files on PUBLIC, PDISK1, or MERLIN that you want to use as input to an SPSS-CMS program. You can access one of the OS disks by using the "LINKMVT volume" command, where volume stands for PUBLIC, PDISK1, or MERLIN. When you're finished using the OS disk dataset you can release it by using the "LINKMVT volume OFF" command. When specifying an OS disk dataset the FILEDEF command is written as:

FILEDEF ddname * DSN dataset name

where ***ddname*** is a "3" for system file input and an "8" for raw data input. For example, if the raw data file USER.P9999.RAWDATA exists on PUBLIC you can use it in an SPSS-CMS program as follows.

```
LINKMVT PUBLIC
FILEDEF 8 * DSN USER.P9999.RAWDATA
SPSS filename
LINKMVT PUBLIC OFF
```

where "filename" stands for the SPSS program file. NOTE: Since CMS does not support data containing spanned records, you cannot use an OS dataset in SPSS-CMS that has a record format of VBS.

Use a CYBER Created Dataset in an SPSS-CMS Program

Raw data files existing on CYBER disk may be sent to CMS for processing by an SPSS-CMS program; however, SPSS system files created on the CYBER are not useable by SPSS-CMS programs. To send a file from the CYBER to CMS issue the following command while logged onto the CYBER computer:

PRINT,fn/J=userid/FORMS=CMS

where ***fn*** is the name of the file to be sent and ***userid*** is your CMS userid. The file will be sent from the CYBER to your IBM virtual reader. You can query your reader to find out if the file has arrived by using the command:

Q RDR ALL

If the file is on your reader, MVT will appear under the label ORIGINID and your userid will appear under the label NAME. Use the READ command to read the data records from your virtual reader and to create a CMS disk file containing the data records. The format of the READ command is:

READ fn ft fm

where ***fn ft fm*** stand, respectively, for the filename, file type, and filemode you want to assign to the file being read. Next you need to XEDIT the file to accomplish several things. You need to use the DEL command in XEDIT to eliminate the burst page from the top lines and bottom lines of the file, and you need to use the SET LRECL 80 command in XEDIT to fix the record length of the file. Finally, use the FILE command in XEDIT to save the corrected file. You now have a copy of a CYBER file on your CMS disk.

Submit a Job to OS/MVT from CMS

There may be situations in which you want to use CMS to create your program file using the XEDIT editor, but you want to submit your file to OS/MVT for execution in batch mode. To accomplish this you need to append to the top of your SPSS program file the ID lines and JCL lines necessary for running the job in batch mode. These must include the lines:

//userid JOB and /*ID FORMS=CMS

where ***userid*** is your CMS userid. The file would appear as follows:

```
//userid JOB
/*ID PS=xxxx
/*ID CODE=xxxx
/*ID REGION=350K,TIME=(x)
/*ID FORMS=CMS
// EXEC SPSS
   ...the SPSS program statements go here...
```

Then you would send the file to OS/MVT for execution in batch mode by typing the command:

SUBMIT fn ft fm

where ***fn ft fm*** stand, respectively, for the filename, filetype, and filemode of the file to be submitted. To query the status of your OS/MVT job you can use the command:

$DJ userid

When the job is finished executing, the program output will be sent to your virtual reader. You can query your reader to find out if the file has arrived by using the command:

Q RDR ALL

If the output is on your reader, MVT will appear under the heading ORIGINID and your userid will appear under the heading NAME. Use the DEPRINT command to read records from your virtual reader and to store them in a CMS disk file. Format of the DEPRINT command is:

DEPRINT fn ft fm

where ***fn ft fm*** stand, respectively, for the filename, filetype and filemode you want to assign to the file being read. The difference between the DEPRINT command and the READ command, explained earlier, is that READ strips off carriage control, while DEPRINT preserves the carriage control of the file.

Since the submit file contains your PS number and code word you won't want other users to have access to this file. Storing the file with a filemode number of 0 will accomplish this. When other users access your disk in read status, files with a filemode number of 0 cannot be accessed. To create a file with a filemode number of 0 type:

XEDIT fn ft A0

where ***fn*** and ***ft*** stand for the filename and filetype of the file to be created. To determine the filemode of an existing file type:

LISTFILE fn ft

To change the filemode of an existing file use the "RENAME" command. For example, if you want to change the file "PROG1 SPSS A1" so that it has a filemode of A0 type:

```
RENAME PROG1 SPSS A1 = = A0
```

MATHEMATICAL SERVICES

SIMSCRIPT NEWSLETTER -- SIMSNIPS

CSO has received the July 1983 issue of SIMSNIPS, a newsletter published by CACI, Inc., for users of the SIMSCRIPT simulation language. This newsletter is devoted to describing ongoing activity with SIMSCRIPT, conferences, books, and other things of interest to those doing simulation.

If you wish to see SIMSNIPS, please talk to Stan Kerr (179 DCL, 333-4715, or send a message on the CYBER systems by the command TELL,UN=MATHLIB).

SLAM USERS GROUP

We have received a letter from a person interested in forming a national users group for the SLAM simulation package. As he puts it, the group "is intended to provide a forum for exchange of ideas on applications of SLAM and for discussion of solutions to problems encountered when using the approach and language." He asks that interested persons send him a letter at the following address:

Lt Col Thomas D. Clark, Jr.
Air Force Institute of Technology
School of Engineering; Dept of Operational Sciences (ENS)
Wright-Patterson AFB, Ohio 45433

Mr. Clark will be at the 1983 Winter Simulation Conference from December 12-14 of this year at the Marriot Crystal Gateway Hotel in Arlington Virginia, and plans a Users Group meeting on Tuesday evening, December 13.

CONVERSION OF SOME CYBER PACKAGES TO FTN5

In the June issue of *OFF-LINE*, we announced that many CYBER packages are being converted to FTN5, and gave a target date of

Monday, January 9, 1984

for the final conversion date. Before that date, FTN5 versions will be available via the FUTURE and GRAB/F commands for user testing. After that date, the old FTN versions will not be available except by special arrangement.

In the July, August and September issues, we announced the conversion of various packages. The full group of products now available in FTN5 versions via GRAB/F and FUTURE is

IMSL	SLAM	FORSIM	GRG	FLEX	GMET
XMP	UOILIB	LINPACK	EISPACK	MINPACK	ITPACK
NETWORK	LSODE	AMOSLIB	GASP	MSL	

More products will be announced in the coming months.

You are strongly encouraged to begin converting your FTN programs to FTN5 and to use the FTN5 versions of these products whenever possible. The FTN compiler, though reliable, is no longer supported by CDC; FTN5 is their "official" FORTRAN compiler now, and represents the new standard for the FORTRAN language. In deference to this, CSO would like to encourage the eventual "fading away" of the old compiler. At some point in the (not too near) future, we plan to alter the system so that the FTN control statement will access the new compiler and the FTN4 control statement will access the old one.

Manuals for FTN5 can be purchased at the CSO Distribution Office, 1208 W. Springfield. A utility, F45, is available for doing automatic conversions of some types of FTN statements to equivalent FTN5 forms. The manual for this also can be purchased at the Distribution Office; a copy of the manual is available for inspection at the CSO Systems Consulting Office at 1208 W. Springfield.

THE PROTRAN PROBLEM-SOLVING LANGUAGE

CSO would like to encourage users to acquaint themselves with the PROTRAN problem-solving language from IMSL. Designed as a family of similar languages directed toward different problem areas, the PROTRAN family now includes MATH/PROTRAN and STAT/PROTRAN (both of which CSO has acquired), and will include LP/PROTRAN, a language for linear programming (CSO has no plans to acquire LP/PROTRAN).

This Fall, CSO offered two sections of a short course on PROTRAN. The first section was held in September; however, the second section meets from October 17 to October 20 from 3pm to 4pm. To register, either fill out a short course registration form (if one is handy), or come to 150 Digital Computer Laboratory (the CSO office) to sign up. The fee is $20.

PROTRAN is available on the CYBER systems by entering the command

GRAB,PROTRAN.

This accesses the MATH/PROTRAN version. A combined version with MATH and STAT options can be accessed by the command

GRAB,PROTRAN/F.

or

FUTURE,PROTRAN.

(Sometime this Fall, the combined version will become the default.)

The GRAB command places several files used for PROTRAN into your local file space. Running PROTRAN involves first translating it with the PROTRAN command, much as you use the FTN or FTN5 command to compile a FORTRAN program.

For example, to evaluate the integral of EXP(X**2) from X=0 to X=1, the following PROTRAN program might be used:

```
      PROGRAM MAIN(OUTPUT,TAPE6=OUTPUT)
$     INTEGRAL  EXP(X**2); FOR(X=0,1); IS XINT
$     PRINT XINT
$     END
```

If you place this in, say, file XYZ, then the following command causes PROTRAN to translate this program into FORTRAN (FTN5 FORTRAN, to be exact), and then compile it into an LGO file:

```
PROTRAN,XYZ.
```

Then, the command

```
LGO.
```

causes the program to run (just as for FORTRAN). The integral is evaluated and the value stored in the FORTRAN variable XINT; this is then printed by the $PRINT command in the program. You will see the output displayed on your terminal.

Documentation on PROTRAN is available in several forms:

1. An IMSL manual can be purchased at the Distribution Office, 1208 W. Springfield, for $15.50 (includes a binder). This price applies to each of the manuals for MATH and STAT/PROTRAN; the total cost for a combined MATH/STAT PROTRAN manual without binder is $30.

   Updates to the manual will cost extra; an update is expected within the next several months. At the time of this printing, we have no price schedule for these manual updates.

2. A reference guide (RF-4.32) summarizes how to access PROTRAN, and includes a one-page summary of all the math procedures in MATH/PROTRAN.

3. An on-line writeup summarizes PROTRAN and includes examples for using the various math procedures. This is not intended to replace the full IMSL manual, only to supplement it. An experienced person may find this on-line writeup adequate for their purposes.

NETWORK SERVICES

SELF-SERVICE PLOTTER

A recent development project has centered around the use of a Printronix dot-matrix printer capable of printing 600 lines per minute with a resolution of 72 dots per inch vertically and 60 dots per inch horizontally. The printer is directly attached to a Digital Equipment Falcon microcomputer that is connected via the campus Sytek network to the PDP 11/40 that front-ends the CYBER 175. The printer is located in the Mechanical Engineering RJE site, room 65 ME.

Software has been developed on both the CYBER and the Falcon that takes a plot file normally intended for the Zeta plotter and converts it to be printed on the Printronix printer. This process involves taking the vectors normally intended for the pen plotter and changing them into raster points that may address the dot printing capabilities of the printer.

The benefits of this implementation are as follows:

- The average plot requires approximately 5 minutes on the pen plotter; an average plot on the Printronix will require about 30 seconds.

- The device is initially being used in a self-service mode, implying a shorter turnaround time for retrieving a plot.

Although the resolution of the printer is not as good as the pen plotter, it should be more than adequate for class assignments and for proofing plots before using the pen plotter for final productions.

Usage

To submit a plot to the self-service printer/plotter, follow these steps:

- Generate a Zeta plot file, using GCS, DI-3000, or any other plotting package.

- Issue the command:

  PLOT,file/DEST=SSP

If all has gone well, you will receive the message:

```
FILE(S) NOW READY FOR PRINTING
GO TO THE PRINTER TERMINAL AND FOLLOW
THE POSTED INSTRUCTIONS
```

- You may now logoff, or continue working as you wish. When you are ready to retrieve your plot, go to the terminal next to the printer and follow the posted directions.

Restrictions

The following restrictions apply:

1. You may have only **one** set of plots or prints waiting at a time. Thus, if you issue the following set of commands

   ```
   PLOT,FILE1/DEST=SSP.
   PLOT,FILE2/DEST=SSP.
   ```

   only the second file (FILE2) will be available to be printed. The first file (FILE1) will be gone.

2. There is a limit to the complexity of the plot that you may submit to the self-service printer. A plot that requires more than 15,000 vectors will be considered too large for plotting, and will have to be directed to the Zeta pen plotter. Error messages for this and other situations will be displayed on your terminal.

3. Plots may be submitted only from a timesharing session at this time.

Future Plans

We are considering many things. Your comments on the following list of items, along with any additional comments or suggestions, are welcome.

1. We are considering installing more of these printers around the campus. One will probably be installed at Commerce West by the start of the second semester.

2. We recognize that there is a need to make some of the printers queued devices. This would allow you to submit a plot, then come to the site to pick up the plot without having to specifically ask for your plot and wait for it to be printed. Your preference for a self-service plotter versus a queued plotter would be helpful in our future decisions.

3. At this point, we realize that there can be improved quality in the rasterization process to speed the printing of the plot and to improve the quality of the plot. We will also work on reducing the CPU time used to rasterize a plot and to reduce the number of DAYFILE messages that are generated.

DOCUMENTATION

NEW AND REVISED REFERENCE GUIDES

Following is a list of new and revised reference guides now available at the various RJE sites:

RF-0.3	Job Entry Sites (RJE)	Revised 09/15/83
RF-7.16	PRINT	Revised 07/20/83
RF-7.18	PUNCHC	Revised 07/20/83
RF-20.2	SIM3278 for IBM Timesharing	New 09/15/83
RF-20.3	Sending Files to CMS From the CYBER	New 09/15/83

OFF-LINE's Mailing List

If you wish to be placed on our mailing list for future issues of *OFF-LINE*, if you wish to be removed from the list, or if you wish to enter an address correction, please complete and return this page. **(Current subscribers are kept on the mailing list until a specific request for removal is received, or until a mailing is returned as undeliverable.)**

Check one:

- ☐ Place my name on mailing list
- ☐ Make the following corrections or changes
- ☐ Delete my name from mailing list

_____

First name -- Initial -- Last Name

Campus Address:

_____

Room - Building

_____

Department

Off-campus Address:

_____

Organization or Company (if applicable)

_____

Street Address

_____

City -- State -- Zip Code

If address correction, give old address and zip code below.

_____

_____

_____

10/83

SEND TO:

OFF-LINE
150 Digital Computer Laboratory
University of Illinois at Urbana-Champaign
1304 West Springfield Avenue
Urbana, Illinois 61801

Computing
Services
Office

Off-Line

University of Illinois at Urbana-Champaign VOL. 11, NO. 11 November 1983

EDITOR: Lynn Bilger
PHONE: (217) 333-6236
120 Digital Computer Lab
1304 W. Springfield Ave.
Urbana, Illinois 61801

CSO DIRECTORY - STAFF AND SERVICES

Administrative

Director	George Badger	150	DCL	333-4103
Business Manager	Stanley Rankin	150	DCL	333-6530
Secretary	Joyce McCabe	150	DCL	333-1637

User Services and Hardware/Software Support

User Accounting		**1208**	**W Springfield**	**333-7752**
Distribution Office		**1208**	**W Springfield**	**333-6760**
Systems Consulting		**1208**	**W Springfield**	**333-6133**
Statistical Services Consulting		**85**	**Comm West**	**333-2170**
Text Processing Consulting		**118**	**DCL**	**333-7318**
Terminal Repair Service		**150**	**DCL**	**333-0969**
Dial-up Numbers	**CYBER 175 (NOSA)**	**300**	**baud**	**333-4000**
	CYBER 174 (NOSB)	**300**	**baud**	**333-4004**
	IBM 4341 L10 (VME)	**300**	**baud**	**333-4005**
	IBM 4341 M2 (VMD)	**300**	**baud**	**333-4006**
	Switch			**333-4008**
Asst Dir User Services	Robert Penka	173	DCL	333-4709
Asst Dir Systems and Operations	Sandra Moy	177	DCL	333-4703
Asst Dir Development	J. M. Randal	175	DCL	333-9772
Asst Dir Engineering and Hardware Selection	Cliff Carter	195	DCL	333-3723
Manager Accounting Services	Gary Bouck	1208	W Springfield	333-7752
Manager Statistical Services	Beth Richardson	189	DCL	333-6276
Microcomputer Laboratory	Jack Knott	86	Comm West	333-6562
Documentation	Lynn Bilger	120	DCL	333-6236
CYBER-IBM Operations	Myra Williams Ann Ryczer Rex Duzan	194a	DCL	333-6562
VAX Operations	Debbie Hudson	123	DCL	333-8150
Laboratory Support Project	Mike Gardner	18	DCL	333-7904
Site Operations North	Sylvia Hansen Chuck Harnsberger	65	ME	333-6285

CSO Sites (see Reference Guide RF-0.3 for operating hours)

Agriculture	N-120	Turner Hall	333-8170
Chemistry	153	Noyes Lab	333-1728
Commerce West	70	Comm West	333-4500
CRH Snack Bar	120	Snack Bar	333-1851
DCL Routing Room	14	DCL	333-6203
Electrical Engineering	146	EEB	333-4936
Florida Ave Res Hall		FAR	333-2695
Illinois St Res Hall		ISR	333-0307
Mechanical Engineering	65	MEB	333-1430
Psychology	453	Psych Bldg.	333-7815
Social Science	202	Lincoln Hall	333-0309

OFF-LINE is the monthly newsletter of the Computing Services Office at the University of Illinois at Urbana-Champaign. Unless otherwise indicated, permission to reprint is freely granted, provided that the author, if named, and the Computing Services Office (CSO) are credited. Information in this issue is current as of October 21, 1983.

CSO operates a CDC CYBER 175 with 262K words of central memory and a CDC CYBER 174 with 196K words of central memory. The 175 and 174 run under the NOS Operating System and share 512K words of ECS. The 175 serves over 200 simultaneously active text and graphics terminals and the 174 serves over 100 simultaneously active terminals. CSO also operates an IBM 4341 Model M2 with 8 million bytes of memory running HASP-OS/MVT under VM and an IBM 4341 Model L10 with 4 million bytes of memory running VM. In addition, CSO operates a DEC VAX 11/780 with 4 million bytes of memory, running under the UNIX Operating System and driving a GSI CAT-8 phototypesetter.

Operating Hours (see HEARYE,SCHEDUL for exceptions):

	CYBERS 174/175	IBM
M-F	8 am - 6 am	8 am - 6 am
SAT	8 am - Midnight	8 am - 6 am
SUN	Noon - 6 am	Noon - 6 am

POLICY

TAPE SERVICE AT CSO

Magnetic tapes to be used on the CYBER or IBM must be checked-in to the machine room and assigned a label. This function is performed in Room 123 DCL. There is a desk immediately to your left upon entering Room 123; on the desk is a switch box and a logbook. When you turn the switch box "on" an operator will be prompted to assist you. While you're waiting for a person to respond, complete a new line entry in the logbook for each of your tapes. Tapes can also be checked-out of the machine room or cleaned in Room 123.

You may request tape cleaning in one of three ways;

1. Call 333-8640 and state your request.

2. Send a CYBER message to UN=20333.

3. Come to Room 123 DCL and fill out a tape cleaning request form. All users who wish to have more than three tapes cleaned ***must*** fill out this form.

Room 123 is open between the hours of 8:30 AM and 4:30 PM, Monday through Friday. We believe these hours will accommodate almost everyone's tape handling needs. These hours enable us to staff this service with full-time CSO employees, providing an opportunity to expand the functions of the service. If you have any questions or comments about this service, please stop by Room 123 DCL (333-8640) and talk with Greg German.

MACHINE NAMES

(Reprinted from the June 1983 issue of *OFF-LINE*)

With the acquisition of a second CYBER and a second IBM, it became obvious that a logical naming scheme was necessary to differentiate the various machines in our documentation and for purposes of access via various local and national networks. The staff members of CSO have had intensive discussions about the choice of names for the various machines, and have adopted the following naming conventions as a result of these discussions:

Name	*Machine Referenced*	*Operating System*
NOSA	CDC CYBER 175	NOS
NOSB	CDC CYBER 174	NOS
UXC	VAX 11/780	UNIX
VMD	IBM 4341 Model M2	VM
VME	IBM 4341 Model L10	VM

The attempt has been to reference each machine by its operating system followed by an alphabetical letter which could be used as a short-hand form of address in some instances (e.g., if only the CYBERs were being discussed, one could talk about A and B rather than always using the terms CYBER 175 and CYBER 174, respectively).

For persons across the country who access our machines via a network system, the naming scheme should work equally as well. The only difference would be that they would need to prefix the name of the machine they wish to use with the letters UIUC (e.g., to access the IBM 4341 Model M2, they would use UIUCVMD).

In conjunction with the adoption of this naming scheme, CSO is encouraging persons using the Gandalf switch to immediately begin using the above names to access the machine they want. For example, when using a terminal on the switch, type **NOSA** instead of **175** to connect to the CYBER 175 or **NOSB** instead of **174** to connect to the CYBER 174. The old switch names will remain in effect for a period of time to allow all users to become familiar with the new names.

> ***NOTE: Since the publication of the above article in June, several additions have been made. Persons using a dial-up to connect to the computers must preface the machine names with the letter D. Persons accessing the computers via a purchased port, must preface the machine names with the letter R. Also, the old machine designators (i.e., 174, 175) are no longer valid; only the names given above are accepted by the Gandalf switch.***

PROTECT YOUR COMPUTER FUNDS

Thefts of computer funds, if they are to occur at all, usually happen late in the semester. Now is a good time to consider what you can do to protect your computer funds.

You should give your signon the same consideration that you would a credit card. Some points to consider are as follows:

- Change your password ***frequently***. Use some creativity when choosing a password. Avoid obvious possibilities such as your initials, spouse's name, etc.

- ***Do not*** keep signon information in publicly accessible files.

- When signing onto a terminal, ***do not*** enter your ID and password on the same line. Enter your ID by itself and wait for the computer to prompt you for your password. If your terminal is operating in full duplex mode (the default for the CYBERs and the Gandalf switch) your password won't print when you enter it after the PASSWORD prompt.

- Be extremely careful with any card decks which contain your signon information.

- Observe the signon message, printed at your terminal or placed in your dayfile, which gives the time and date of the last recorded use of your signon. This message can alert you quickly to unauthorized use of your signon.

- If you are a Project Manager, you should be alert to misuse of your project. You should periodically change the codeword, which is requested every time you use the MANAGE program to add or delete user numbers from your project or to change limits/allocations for user numbers currently in your project. We know of several instances in which a thief gained access to a Project Manager's signon and proceeded to use MANAGE to add new bogus user numbers to the project for his own use.

Your signon is your responsibility and only you can take care of it. CSO will refund stolen computer funds only if CSO is at fault. We will, however, make every effort to help you identify the offender, using the logs maintained by the system.

In the past, cases involving theft of computer funds have been turned over to campus disciplinary authorities who have always required that restitution be made. One case, which involved large sums of money, was also turned over to the State Attorney's office for criminal prosecution.

CYBER

SHAZAM VERSION 4.5 INSTALLED

We have recently installed SHAZAM version 4.5 on the CYBER computer system. SHAZAM is an econometrics package developed by K. J. White, Department of Economics, University of British Columbia, Vancouver, Canada. SHAZAM is relatively easy and inexpensive to use, and it provides a great deal of flexibility in the format of the data.

It has primary capabilities in regression analysis, simultaneous equation estimation, and principal components.

Secondary capabilities are in factor analysis, analysis of variance, sorting and plotting variables. SHAZAM provides extensive data manipulation capabilities; it can generate various random probability distributions and a Divisia Price Index.

The Regression analysis section of SHAZAM features some of the most current econometric models and algorithms. Regression analysis is a statistical technique that attempts to "explain" or "predict" in the form of a single empirical equation, movements in one variable, called the dependent variable (or "regressand"), as a function of the movements in a set of variables, called the independent variables (or "regressors"). The Regression model, and it's various extensions, is perhaps the most widely used technique in applied econometrics and statistics. Computer programs implementing and integrating these various extensions are not readily available. However, SHAZAM offers, in addition to the standard Ordinary Least Squares technique, some alternative functional forms for the regression model. For example, linear restriction and hypothesis testing can be easily done in SHAZAM using the RESTRICT and TEST subproblem commands, respectively.

In fact, very few programs have the capabilities in these areas that SHAZAM has. Other Regression models include: First and Second Order Maximum-Likelihood; Least Squares Estimation of Higher-order Autoregressive and Moving-average models; First Order Autoregressive models with a lagged dependent variable; Testing of residuals for normality; Exact and Higher Order Durbin-Watson tests; Forecasting; Ridge Regression; Box-Cox and Box-Tidwell Regressions; Box-Cox Autoregressive models; Autoregressive models with missing observations; Distributed Lags; Nonlinear Regression; Probit, Logit, and Tobit regressions; Gram-Schmidt regression; Two-Stage Least Squares; Generalized Least Squares; Pooled Cross Section - Time Series; Regression on principal components; Iterative Three Stage Least Squares; Multivariate Regression; Seemingly Unrelated Regressions; and Regressions using Householder Transformations.

SHAZAM can be run interactively on the terminal as well as in the batch subsystem. To access and run SHAZAM from time-sharing, enter:

GRAB, SHAZAM.
SHAZAM, lfn1, lfn2.

where

lfn1 the name of your local file containing your problem specification and data

lfn2 the name of the local file to which your results are written.

The commands in a SHAZAM run can be classified into seven types as follows:

1. Global Commands [optional]
2. SOLOMON Command [required]
3. HERCULES Housekeeping Section [optional]
4. ATLAS Variable Generation Section [optional]
5. ZEUS Data Section [required]
6. ACHILLES Subproblem Commands [required]
7. MERCURY Finish Command [required]

Usually, Global commands will not be needed. The other types of SHAZAM commands will probably be used in most runs. The order of the SHAZAM run should appear as above. All SHAZAM commands must be typed beginning in column 1. SHAZAM command names can be abbreviated to the first four characters if the name is longer than four characters. (For example: the SOLOMON command could be abbreviated to SOLO.) Most jobs will use only a few of the available commands.

Command-Type/Name	***Description***
GLOBAL	
PAR	Sets Workspace Dynamically
TERMINAL/BATCH	Terminal/Batch Mode Command
SPACE	Space allocation Command
SIZE	Set maximum number of variables

SOLOMON	Begins the Main Problem

HERCULES

IO	Input-Output Options Command
PUNCH	Punched Output Options Command
SMPL	Sample Selection Command
NAME	Variable Names Command

ATLAS

GENR	Generate Variables Command
IF	Conditional GENR Command
SKIPIF	Skip Observations Command
DO	Do Loop Command
READ	Read Statement
USER	User Written Subroutine Command

ZEUS

DATA	DATA Command (If your data is not in a file it should go here)

ACHILLES

1. Regression Subproblem Commands:
   OLS, HH, GS, RPC, AUTO, 2SLS,
   DL, BOX, GLS, POOL, PROBIT, LOGIT,TOBIT

2. Other Subproblem Commands:
   PC, ANOVA, GRAPH, HS, VARS, FC, SORT, INDEX

SYSTEM	System Definition Command
NL	Nonlinear Definition Command
EQ	Nonlinear Equations Command
OPTIONS	Subproblem Options Command
LAMBDA	Lambda Restrictions Command
LAG	Lag Restrictions Command
RESTRICT	Linear Restrictions Command
TEST	Hypothesis Test Command
ADD	Addition Calculation Command
FM	Format Command

MERCURY	End of Main Problem Command

You can stack as many Main Problems as desired by repeating the sequence from SOLOMON to MERCURY as in the example shown below (the lines beginning with an asterisk (*) are comments):

```
* GLOBAL DECLARATION SECTION
PAR=50
* BEGIN MAIN PROGRAM
SOLOMON 4 9 OLS EXAMPLE
* ZEUS SECTION
DATA(F4.0,3(1X,F3.0))
1948 100 100 100
1949 106 104 99
1950 107 106 110
1951 120 111 126
1952 110 111 113
1953 116 115 103
1954 123 120 102
1955 133 124 103
1956 137 126 98
* ACHILLES SECTION
OLS 2 3 4 / RE CR CV
MERCURY
SOLO 5 50 TEST SHOWING CHANGING SAMPLE PERIODS
SMPL 3 25
DATA
  (insert data here, using free format option)
OLS 5 1 3 / RE SS
OLS 5 2 4 / RE SS
SMPL 26 48 (50)
SS
MERC
```

While the above list may appear to be complicated, it shows the wide variety of options available in SHAZAM. In fact, while most users will only use a few of the commands shown, the list does indicate the order of any SHAZAM Commands that are used. Users will find that SHAZAM actually requires little information for most types of problems. Users who do not require any special SHAZAM options will probably find everything they need to know in Chapter II of the on-line document. This document can be obtained and printed on the PPS printer by entering:

WRITEUP,SHAZAM.
PRINT,SHZDOC/AS/CC/EJ/FORMS=PPS.

All questions on SHAZAM use should be referred to Esther Edwards-Iwe, (131-B DCL, or send a message on the CYBER systems by the command TELL,UN=3SDYNT3).

TAGS - MAILING LABEL PROGRAM

CSO has installed TAGS, a CCL-based procedure for creating, printing, and storing mailing labels and lists on the CYBER computer. Labels can be quickly changed, or modified in many ways. TAGS can be used interactively at your terminal or with batch control statements.

One of TAGS' most useful features is its ability to sort your labels. The sort "keys" include name, address, zip code, or any three other codes you type in as "keys." This means if you specify a key for last names, your labels will be sorted by last name, but printed with first names first.

TAGS automatically formats your labels with one line for name, two for ADDRESS, one for CITYSTATE, and one for ZIP CODE. Labels can be printed on standard paper or on sticky-back forms. Each of the first four lines of a label can contain 28 characters, including any blank spaces. The fifth line, CODE, is limited to ten characters. TAGS recognizes the first line as NAME, the next two as ADDRESS1 and ADDRESS2, the fourth as CITYST, and the last as CODE.

TAGS is an application of UWRIM, a relational data base management system that collects and manipulates tables of data. Although TAGS simplifies UWRIM for mailing label use, it is compatible with all UWRIM commands.

TAGS can be accessed on the CYBERs by the command

GRAB, TAGS.

This makes available to you the TAGS library and all the necessary products needed to use this program.

An on-line document describing TAGS further can be obtained and printed on the PPS printer by entering

```
WRITEUP,TAGS.
PRINT,TAGDOC/AS/CC/EJ/FORMS=PPS.
```

IBM

SCSS AVAILABLE ON THE IBM COMPUTER

The SCSS system is a comprehensive set of data management and statistical analysis facilities available on the IBM 4341-M2 (VMD) computer. Since SCSS is a product of SPSS Inc., you can use many of the features available in the SPSS batch system. However, using SCSS is different from using SPSS because you don't have to spend time writing a program.

SCSS, the SPSS Conversational System, is so named because communication between you and the system resembles an actual conversation. SCSS guides you through its facilities by asking a specified series of questions; you respond with your choice of appropriate answers. The conversation is in English, with no programming skills needed. If you're unsure how to answer a question, you can request help by typing a question mark. If you make a mistake, SCSS tells you immediately and repeats the question. You don't have to go back to the beginning and start again like you do with SPSS. Being able to correct mistakes instantly means that you can get your statistical results in your first session rather than collecting stacks of error messages. When you ask SCSS to give you some information about the data or a table of statistical

results, it does so immediately. You proceed naturally through your data investigation using your research skills without needing to master computer skills.

The statistical procedures available in SCSS are:

UNIVARIATE	Basic statistics, frequency tables, histograms.
CROSSTABS	Generates 2-way to n-way joint frequency distribution tables and related statistics.
T-TEST	Tests the significance of differences in the means of a variable measured in two independent subgroups or in paired samples.
BREAKDOWN	Basic statistics with control groups.
SCATTERPLOT	Produces bivariate scatter diagrams and simple linear regression statistics.
CORRELATIONS	Calculates Pearson product-moment correlations between variables.
PARTIALS	Calculates nth order partial correlation coefficients.
REGRESSION	Develops a model in which the dependent variable is expressed as a linear combination of one or more independent variables.
FACTOR	A multivariate procedure used primarily to reduce the information contained in a set of variables to a smaller set of composite variables called factors.

If you're interested in data management SCSS can be used to:

- create and transform variables
- weight, select, and sample data
- choose missing data treatment
- store matrices

To run the SCSS system signon to VMD and type:

SCSS

Three sample data files, named SURVEY, AUTO and ECON, are available to users who wish to practice using SCSS. The file SURVEY contains information on the American population which was gathered in 1977 by the National Opinion Research Center. The file AUTO contains gasoline mileage information for various kinds of automobiles. The file ECON consists of U.S. economic data from the years 1947 to 1970. To access any one of these files type:

SCSS filename

For example, type SCSS SURVEY to access the SURVEY file.

A handout of seven SCSS practice exercises is available in the CSO South Consulting Office, 85 Commerce West. The exercises use the SURVEY file and offer you quick hands-on experience with using the SCSS system. Exercise 1 helps you to signon to SCSS and to investigate the UNIVARIATE procedure. Exercise 2 demonstrates use of the CROSSTABS procedure. Exercise 3 gives you practice with revising variables. Exercise 4 demonstrates use of the BREAKDOWN procedure. Exercise 5 gives you practice with computing new variables. Exercise 6 demonstrates entering data at a terminal. Exercise 7 helps you to investigate the REGRESSION procedure.

The SCSS system is documented in *SCSS, A User's Guide to the SCSS Conversational System*, McGraw-Hill, 1980. Another manual, *SCSS Short Guide*, McGraw-Hill, 1982, is also available. The short guide presents information in the user's guide in a condensed form to enable you to become quickly familiar with the system. Both of these manuals may be purchased at the CSO Distribution Office, 1208 W. Springfield.

MATHEMATICAL SERVICES

IBM ACRITH SUBROUTINE PACKAGE

CSO has received information from IBM concerning a new subroutine package intended for extremely high-accuracy numerical calculations. Called ACRITH, this package permits, for example, the calculation of a dot product of arbitrary length to machine accuracy. (Normally, there is a rounding error associated with each term in the dot product; this error can accumulate to serious levels across a dot product of thousands of terms.) This accuracy is accomplished by the use of a superwide accumulator which permits arbitrary sums and products to be accumulated with zero rounding error.

The package includes routines for doing basic arithmetic operations, vector and matrix manipulations, plus routines for the solution of linear equations and eigenvalue problems. Though considerably slower than conventional algorithms for linear equations and eigenanalysis, these routines have much superior accuracy, making them appropriate for highly ill-conditioned problems.

As IBM's price for this package is rather high, CSO would like to see a positive response from users before we purchase it. If you think such a package can be of significant use in your work, please talk to Stan Kerr (179 DCL, phone 333-4715, or TELL,UN=MATHLIB on the CYBER systems).

PROTRAN INTERACTIVE HELP PROGRAM, PROHELP

CSO has created an interactive help program for the PROTRAN package, using help text supplied by IMSL. This facility is still under development and design, but a first, experimental version is now ready for use; we would very much like user feedback on it.

To access the help facility, enter the CYBER command

GET,PROHELP/UN=MATHLIB

Having accessed it, you can run the help program by entering

PROHELP

The interactive program will be loaded and will ask you what you want. Currently, it can provide general information about PROTRAN, or information about any particular PROTRAN procedure. If you enter the name of a PROTRAN procedure when it requests it, it can provide a detailed description of the procedure, a short syntax summary, or one or more example PROTRAN programs which use that procedure.

We encourage users to try this facility. Suggestions and other feedback can be sent to us on the CYBER either by TELL,CSO or TELL,UN=MATHLIB.

NEW VERSION OF ACSL BEING INSTALLED

CSO has received Version 8D of the ACSL simulation package, and is now installing it. Although a longer article in the next issue of *OFF-LINE* will give more details about Version 8D, certain points are worth mentioning now, especially for instructors planning to use ACSL in the Spring semester. This new version will be placed in FUTURE on the CYBERs; it should be available by the time this issue of *OFF-LINE* appears.

The new ACSL is being installed with the FTN5 FORTRAN compiler, to make it consistent with the other products that we are moving to FTN5, and in keeping with our intention of eventually supporting only FTN5-compiled products. Also, the new ACSL is being installed using different options for its internal table management; this will result in some changes to the way ACSL is used.

We plan to make the new ACSL the default for the Spring semester, at the same time that various other products are converted to FTN5, namely January 9, 1984.

If you are an instructor planning to use ACSL in the Spring, please take note of this, and watch for the next issue of *OFF-LINE* for more details on the new ACSL. Questions about the new ACSL can be taken to Stan Kerr (179 DCL, phone 333-4715, or TELL,UN=MATHLIB on the CYBERs).

CONVERSION OF SOME CYBER PACKAGES TO FTN5

In the June issue of *OFF-LINE*, we announced that many CYBER packages are being converted to FTN5, and gave a target date of

Monday, January 9, 1984

for the final conversion date. Before that date, FTN5 versions will be available via the FUTURE and GRAB/F commands for user testing. After that date, the old FTN versions will not be available except by special arrangement.

In the July, August and September issues, we announced the conversion of various packages. The full group of products now available in FTN5 versions via GRAB/F and FUTURE is

IMSL	SLAM	FORSIM	GRG	FLEX	GMET
XMP	UOILIB	LINPACK	EISPACK	MINPACK	ITPACK
NETWORK	LSODE	AMOSLIB	GASP	MSL	

More products will be announced in the coming months.

You are strongly encouraged to begin converting your FTN programs to FTN5 and to use the FTN5 versions of these products whenever possible. The FTN compiler, though reliable, is no longer supported by CDC; FTN5 is their "official" FORTRAN compiler now, and represents the new standard for the FORTRAN language. In deference to this, CSO would like to encourage the eventual "fading away" of the old compiler. At some point in the (not too near) future, we plan to alter the system so that the FTN control statement will access the new compiler and the FTN4 control statement will access the old one.

Manuals for FTN5 can be purchased at the CSO Distribution Office, 1208 W. Springfield. A utility, F45, is available for doing automatic conversions of some types of FTN statements to equivalent FTN5 forms. The manual for this also can be purchased at the Distribution Office; a copy of the manual is available for inspection at the CSO Systems Consulting Office at 1208 W. Springfield.

NETWORK SERVICES

APPLE II AND IBM PC COMMUNICATION SOFTWARE

CSO has software for communicating with other computers available for both the Apple II and the IBM PC.

The Apple package was developed by Neil Romy and the IBM PC package was developed by C. C. Cheng. The IBM PC package is tailored specifically for use with the CYBER systems.

Copies of both packages are available free. Simply take a diskette to Jack Knott, 70 Commerce West, to get your copy of the package of your choice.

On-line documentation is available on the CYBER for both packages, and may be obtained be entering:

```
WRITEUP,APPLE                    for the Apple II writeup
PRINT,APDOC/AS/CC/EJ
```

or

```
WRITEUP,IBMPC                    for the IBM PC writeup
PRINT,IBMDOC/AS/CC/EJ
```

These writeups may be viewed at your terminal by simply using the TYPE command instead of the PRINT command in the above statements.

MISCELLANEOUS

LIBRARY USER SURVEY

The microcomputer revolution coupled with a phenomenal upsurge in information available electronically present real challenges as well as opportunities for library service. In an attempt to meet that challenge, the University Library will conduct an electronic survey via CYBER and PLATO to identify the information needs of those individuals on campus who access information and library resources electronically rather than through traditional means. Please watch for a banner announcing the Library's questionnaire and if possible take a few minutes to respond. Your interest and cooperation will be greatly appreciated.

Questions and requests for the results of the research should be directed to Pat McCandless, Public Services, 246A Library, 3-0317.

OFF-LINE's Mailing List

If you wish to be placed on our mailing list for future issues of *OFF-LINE*, if you wish to be removed from the list, or if you wish to enter an address correction, please complete and return this page. **(Current subscribers are kept on the mailing list until a specific request for removal is received, or until a mailing is returned as undeliverable.)**

Check one:
- ☐ Place my name on mailing list
- ☐ Make the following corrections or changes
- ☐ Delete my name from mailing list

_____

First name -- Initial -- Last Name

Campus Address:

_____

Room - Building

_____

Department

Off-campus Address:

_____

Organization or Company (if applicable)

_____

Street Address

_____

City -- State -- Zip Code

If address correction, give old address and zip code below.

_____

_____

_____

11/83

SEND TO:

OFF-LINE
150 Digital Computer Laboratory
University of Illinois at Urbana-Champaign
1304 West Springfield Avenue
Urbana, Illinois 61801

Computing Services Office

Off-Line

University of Illinois at Urbana-Champaign VOL. 11, NO. 12 December 1983

EDITOR: Lynn Bilger
PHONE: (217) 333-6236
120 Digital Computer Lab
1304 W. Springfield Ave.
Urbana, Illinois 61801

CSO DIRECTORY - STAFF AND SERVICES

Administrative

Director	George Badger	150	DCL	333-4103
Business Manager	Stanley Rankin	150	DCL	333-6530
Secretary	Joyce McCabe	150	DCL	333-1637

User Services and Hardware/Software Support

User Accounting		**1208**	**W Springfield**	**333-7752**
Distribution Office		**1208**	**W Springfield**	**333-6760**
Systems Consulting		**1208**	**W Springfield**	**333-6133**
Statistical Services Consulting		**85**	**Comm West**	**333-2170**
Text Processing Consulting		**118**	**DCL**	**333-7318**
Terminal Repair Service		**150**	**DCL**	**333-0969**
Dial-up Numbers	**CYBER 175 (NOSA)**	**300**	**baud**	**333-4000**
	CYBER 174 (NOSB)	**300**	**baud**	**333-4004**
	IBM 4341 L10 (VME)	**300**	**baud**	**333-4005**
	IBM 4341 M2 (VMD)	**300**	**baud**	**333-4006**
	Switch			**333-4008**
Asst Dir User Services	Robert Penka	173	DCL	333-4709
Asst Dir Systems and Operations	Sandra Moy	177	DCL	333-4703
Asst Dir Development	J. M. Randal	175	DCL	333-9772
Asst Dir Engineering and Hardware Selection	Cliff Carter	195	DCL	333-3723
Manager Accounting Services	Gary Bouck	1208	W Springfield	333-7752
Manager Statistical Services	Beth Richardson	189	DCL	333-6276
Microcomputer Laboratory	Jack Knott	86	Comm West	333-6562
Documentation	Lynn Bilger	120	DCL	333-6236
CYBER-IBM Operations	Myra Williams Ann Ryczer Rex Duzan	194a	DCL	333-6562
VAX Operations	Debbie Hudson	123	DCL	333-8150
Laboratory Support Project	Mike Gardner	18	DCL	333-7904
Site Operations North	Sylvia Hansen Chuck Harnsberger	65	ME	333-6285

CSO Sites (see Reference Guide RF-0.3 for operating hours)

Agriculture	N-120	Turner Hall	333-8170
Chemistry	153	Noyes Lab	333-1728
Commerce West	70	Comm West	333-4500
CRH Snack Bar	120	Snack Bar	333-1851
DCL Routing Room	14	DCL	333-6203
Electrical Engineering	146	EEB	333-4936
Florida Ave Res Hall		FAR	333-2695
Illinois St Res Hall		ISR	333-0307
Mechanical Engineering	65	MEB	333-1430
Psychology	453	Psych Bldg.	333-7815
Social Science	202	Lincoln Hall	333-0309

OFF-LINE is the monthly newsletter of the Computing Services Office at the University of Illinois at Urbana-Champaign. Unless otherwise indicated, permission to reprint is freely granted, provided that the author, if named, and the Computing Services Office (CSO) are credited. Information in this issue is current as of October 21, 1983.

CSO operates a CDC CYBER 175 with 262K words of central memory and a CDC CYBER 174 with 196K words of central memory. The 175 and 174 run under the NOS Operating System and share 512K words of ECS. The 175 serves over 200 simultaneously active text and graphics terminals and the 174 serves over 100 simultaneously active terminals. CSO also operates an IBM 4341 Model M2 with 8 million bytes of memory running HASP-OS/MVT under VM and an IBM 4341 Model L10 with 4 million bytes of memory running VM. In addition, CSO operates a DEC VAX 11/780 with 4 million bytes of memory, running under the UNIX Operating System and driving a GSI CAT-8 phototypesetter.

Operating Hours (see HEARYE,SCHEDUL for exceptions):

	CYBERS 174/175	IBM
M-F	8 am - 6 am	8 am - 6 am
SAT	8 am - Midnight	8 am - 6 am
SUN	Noon - 6 am	Noon - 6 am

CYBER

NEW VERSION OF DEBLOCK AND TBLOCK PROCEDURES

New versions of the DEBLOCK and TBLOCK procedures on the CYBER for reading and writing tape files of "fixed length" records have been installed in FUTURE. These versions can be accessed by the commands

GRAB, DEBLOCK/F.
GRAB, TBLOCK/F.

They will become the default versions in January 1984, not long after various products are converted to FTN5 (see elsewhere in this issue).

There is little visible difference in the use of the new versions. A parameter NFILES has been added allowing one to easily process a number of files at once on unlabeled (not labeled) tape. This is coded as NFILES=x, where x is the number of files to process (and x=1 if NFILES is omitted).

Some internal file handling has been made cleaner; for instance, if only part of a tape file is read with DEBLOCK, by using the LIMIT parameter, then DEBLOCK carefully positions the file to its end after the operation is complete, rather than leaving the tape positioned where it was. This makes it easy (with unlabeled tape, at least) to list the first several records of a series of tape files with one call to DEBLOCK.

In addition, some dayfile messages which the programs used to produce no longer appear.

IBM

ENHANCED DATA MANAGEMENT IN SPSSX

The new release of the SPSS batch system, SPSSX, is available for use with both OS/MVT and CMS on the IBM 4341-M2(VMD) computer. Among its many new features SPSSX offers enhanced data management capabilities. Some of these enhancements are as follows:

- In one SPSSX job you can now input many raw data files or system files, manipulate them, and output many calculated files or system files. This is accomplished by means of the command, FILE HANDLE, which is explained in the next article.

- Formerly, there was no data checking in the IBM version of SPSS, but now you can read a dirty data file by using the FILE TYPE GROUPED command. A dirty data file is a file that contains multi-record cases in which there are duplicate records, missing records, wild records, and out of order records. The FILE TYPE GROUPED facility provides a handy way of detecting these kinds of data exception conditions. FILE TYPE GROUPED will successfully read the data, warn you about bad data, and take appropriate action.

- Input files previously were required to be rectangular in nature, but by using the command FILE TYPE NESTED you can read a hierarchical file into SPSSX. An example of a hierarchical file is an accident file in which some records have data that apply to the accident such as time of day and weather conditions, some records have data that apply to the vehicle such as type of vehicle and state in which it's registered, and some records have data that apply to the victim such as sex and extent of injury. FILE TYPE NESTED builds one case for each victim with information from the accident and vehicle records spread to the victim records.

- Formerly, files to be merged were required to be parallel in nature, but using MATCH FILES in SPSSX you can merge non-parallel files. For example, suppose the two files to be merged consist of the Fortune 500 data for the top fifty industrials in 1980 and 1981. These are non-parallel files because some companies that appear in the top 50 in 1980 drop out of the rankings in 1981, and some companies that were not in the top 50 in 1980 move into the rankings in 1981. Using MATCH FILES you can take the 1980 data and add to it the data for 1981. MATCH FILES will add the data to the appropriate company or create a new case for each company that does not appear in the top fifty both years.

  You can also use MATCH FILES to do a table look-up match. For example, you can merge a case file in which records are data about cities with a table look-up file in which records are data about states. The result file from MATCH FILES will have cases as cities with data from the state records added to the city records.

- If you wish you can even take control of the input phase yourself with the INPUT PROGRAM facility. For example, using INPUT PROGRAM you can read in a case, break up the variables into sets, and output one case for every set of variables. In that way a single case of input is mapped into several cases of output, and you will have stored many more cases on the active file than there are on the input file.

The features described above are documented in the *SPSSX User's Guide*, McGraw-Hill, 1983. This manual is available for purchase at the CSO Distribution Office, 1208 W. Springfield. If you were unable to attend the CSO short course on SPSSX offered recently, you may obtain the handout for the course entitled "Getting Started with SPSSX" in the CSO South Consulting Office, 85 Commerce West.

SPSSX AVAILABLE IN IBM TIMESHARING

The SPSSX statistical package is available for use in IBM timesharing. You can edit your program and data files using the XEDIT editor, execute your job in timesharing mode, and store your output in a CMS disk file or route it to the printer. During your timesharing session, you can use data files stored on CMS disk, tape, OS disk, or use raw data files that were previously routed from the CYBER to CMS for processing. You can even submit an SPSSX job to OS/MVT for execution in batch mode and have the output routed back to CMS. Running SPSSX programs in IBM timesharing offers you all the benefits you're accustomed to in CYBER timesharing, plus, you have the advantage of convenient access to the most recent release of software available from SPSS Inc.

Running SPSSX Programs in a CMS Environment

To use the SPSSX statistical package in a CMS environment, you must create an SPSSX program file with a filetype of SPSSX. Note that the program file is the file that contains SPSSX commands such as DATA LIST and CONDESCRIPTIVE. You can create the program file using XEDIT by typing:

XEDIT filename SPSSX

This creates a file with the filename of your choice and a filetype of SPSSX. Then using appropriate XEDIT commands you can begin entering lines into this file. You can run the program by typing:

SPSSX filename

where ***filename*** stands for the name of the program file. SPSSX will write program results to a display file with the same filename as the program file and a filetype of LISTING. The display file is written to the disk upon which the program file resides. To view the display file at your terminal, enter:

TYPE filename LISTING

You can get a copy of the display file on paper by typing:

RPRINT filename LISTING

For example, if the program file FREQ SPSSX is on the A disk, then typing SPSSX FREQ causes SPSSX to run the program and store the output in the file FREQ LISTING A. SPSSX expects the program file to have a filetype of SPSSX; if it does not, you can specify the filetype on the SPSSX command as in: SPSSX FREQ1 PROG. You can also specify the filemode as in: SPSSX FREQ1 PROG B. By default SPSSX writes error messages, warnings, and notes to your terminal as well as writing them to the display file. You can control the printing of error messages and redirect the display file with optional specifications on the SPSSX command. To use any of the options, specify them after a left parenthesis following the file specifications as in:

SPSSX filename (options

The available options are:

ALLTERM	Write the display file to the terminal rather than to the LISTING file.
NOTERM	Do not write error messages, warnings and notes to the terminal.
PRINT	Write the display file to the virtual printer.

All options can be abbreviated to the first three characters. If you type **SPSSX ?** you will receive some brief help text.

Using a FILE HANDLE Command

SPSSX can read as input either raw data files or SPSSX system files, and it can produce as output either SPSSX system files or calculated files written by procedures such as PEARSON CORR and FACTOR. You must inform SPSSX about these files in one of two ways. You can either use the FILE HANDLE command in your SPSSX program file, or you can use the FILEDEF command prior to executing SPSSX.

The FILE HANDLE command is used to specify the disk data set to be used and to provide a "handle" for use in other SPSSX commands. When the FILE HANDLE command is used it takes the place of the FILEDEF command. For example, if you want to read the raw data file ILLCORN DATA A then the first two lines of your program file would be:

```
FILE HANDLE CORN/NAME='ILLCORN DATA A'
DATA LIST FILE=CORN/...
```

where CORN is the arbitrary "handle" assigned to ILLCORN DATA A. The only subcommand on the FILE HANDLE is NAME with which you supply the file specification (filename filetype filemode) enclosed in apostrophes. The FILE HANDLE statement in general form is written as:

FILE HANDLE handle/NAME='fn ft fm'

where ***handle*** is one to eight characters in length consisting of the alphabetic characters A-Z and the numbers 0-9, and must begin with an alphabetic character. As another example, suppose you want to save a correlation matrix into file CORMAT CALC A. The program file would contain the statements.

```
FILE HANDLE CORREL/NAME='CORMAT CALC A'
   .
   .
   .
PROCEDURE OUTPUT OUTFILE=CORREL
PEARSON CORR AGE TO OCCUP
OPTIONS 4
```

A program which reads in raw data and saves an SPSSX system file might look like this.

```
FILE HANDLE DATAIN/NAME='DEPT30 DATA A'
FILE HANDLE SFOUT/NAME='DEPT30SF SPSSXFIL A'
DATA LIST FILE=DATAIN/ ...
   .
   .
   .
SAVE OUTFILE=SFOUT
```

Using a FILEDEF Command

Not all files that SPSSX reads and writes can be handled with the FILE HANDLE command, some must be defined by using a FILEDEF command prior to the SPSSX command. The FILEDEF command is necessary when the data resides on an OS disk, when the record length of the data file must be specified, and when the data resides on tape. The general form of the FILEDEF command is:

FILEDEF ddname specifications

When using a FILEDEF with SPSSX the ***ddname*** corresponds to the handle used on SPSSX commands. For a description of the FILEDEF command please see the October *OFF-LINE* article entitled, "SPSS Available in IBM Timesharing".

Accessing the INFO File

Information about running the SPSSX system in a CMS environment is not included in the SPSSX manual. This information is made available in machine readable form through use of the INFO command. The following is a program file INFO SPSSX A which obtains all available documentation since the first release of SPSSX and stores that information in the file XDOC LISTING A.

```
FILE HANDLE XDOC/NAME='XDOC LISTING A'
INFO OUTFILE=XDOC ALL SINCE 1
```

To run this program and print the documentation obtained, you would type the following commands:

```
SPSSX INFO
RPRINT XDOC LISTING
```

IDA AVAILABLE IN IBM TIMESHARING

The IDA interactive statistical package for small matrices (500 x 19) is now available on the IBM CMS timesharing system. IDA, which is the acronym for Interactive Data Analysis, does regression related statistics, including Box-Jenkins and scatterplots where residuals are immediately available for further analysis. IDA has the advantage of being truly interactive; commands are carried out as soon as they are issued. This means that the course of an analysis can be decided while in progress -- adjusting the procedure to the results found.

This CMS version of IDA is almost identical to the CYBER version, which has been available at this installation for some time and is scheduled to continue. Reference Guides for both the CYBER and CMS IDA are in preparation.

Although the document *A User's Guide to IDA* by Ling and Roberts (available at the Distribution Office, 1208 W. Springfield) applies to both CYBER and CMS, there are some facts relative to the IBM CMS system of which the user should be aware. Obviously, the user must obtain an IBM CMS logon to run CMS IDA. Also, CMS file handling conventions apply. Of note is the fact that the user's own files may be accessed directly by name without special commands or IDA-imposed naming conventions (simply use the filename filetype filemode of the CMS file). Files from other CMS areas may require LINK or ACCESS commands with corresponding RELEASE or DETACH commands when done. IBM OS/MVT files (stored on the IBM 4341 under batch mode) or tape files require FILEDEF.

The following technical information was released with the distribution of IDA, and is contained in the IDA NEWS file:

> If you give a command such as ENTE or FSAV which requires access to a file, IDA will prompt you for the name of the file with FILE NAME? You should enter the name of the file just as you would for any other processor running under VM/CMS. That is, you should enter a name, type and mode. There is no default for the name. The type defaults to "IDA", and the mode defaults to "*" for input files. For output files, the mode defaults to "A". "Binary" files, i.e. FSAV and HOLD files, must have a mode of digit of "4".
>
> IDA will refuse to write into an existing file. That is, the file specification given when you are creating a HOLD, FSAV, FOUT, or BOUT file must not reference an existing file.

> In order to reference esoteric files, you may enter a FILEDEF command before invoking IDA and then respond to the FILE NAME? prompt with an integer between 21 and 89. This mechanism is useful for reading files on OS disk volumes or for referencing files with DCB attributes which are incompatible with IDA.

For the beginning IDA user working with his own disk files, the one important fact from the technical notes is that you do not instruct IDA to write on existing files. If you want file output (required for printed output from a CRT terminal or saving data from IDA) either ERASE any old file by that name before invoking IDA or use a new output file name for each new run of IDA.

To run IDA using interactive data entry, follow these steps:

- Enter the command IDA
- Specify ENTE on the > prompt.
- Specify TERMINAL for mode input on the ? prompt.
- Give integers for sample size and number of variables when prompted.
- Enter data one row at a time with spaces between values (rows are subjects) when prompted.
- Name variables when prompted.
- On next > prompt, enter a command such as REGR or YVSX for regression or scatterplot.
- Follow prompts for variable names or numbers, etc.
- On next > prompt, you may issue any independent command.
- To exit IDA, type QUIT on a > prompt.

Note that regression statistics like COEF for coefficients refer back to the most recent REGR in this session. The command COMM gets you a command directory, and the command HELP gets you into an interactive help procedure.

To run IDA with existing data, specify FILE instead of TERMINAL for mode on the ? prompt. Give the full CMS file name when prompted, i.e., filename filetype filemode (filetype defaults to IDA, filemode defaults to *). IDA can read existing files it has written (on CMS) with FSAV or raw files (typed in with the editor or sent from CYBER, etc). When IDA asks if it is an FSAV file, answer YES or NO (NO means a raw data file). If raw data is given, a format is required or the data must be appropriate for freefield.

The distribution tape for CMS IDA also contained 78 sample data files and a directory for their use (IDAFILES MEMO). These have been loaded with the programs. Some examples of the use of these datasets have been documented in *A User's Guide to IDA* and also in *Conversational Statistics with IDA* by Ling and Roberts. The following example shows the use of one of these FSAV files in CMS IDA.

Example: We will use the file BOSTON IDA B4, where IDA and B4 are understood, from the IDAFILES MEMO listing. (The number "4" must be used; however, the "B" prefix will vary from user to user.) Note that all entries in bold face in the example are what the user would enter, and that abbreviations are used for some responses.

(Logon to class VMD)

```
IDA
DMSACC723I  C (120)  R/O
IDA  (CMS RELEASE 01-JAN-81)
> ENTE
WANT EXPLANATION ? N
MODE OF INPUT: FROM "FILE" OR "TERMINAL" ? F
* FILE NAME ? BOSTON
WAS YOUR DATA FILE CREATED BY THE IDA COMMAND FSAV ? Y
DATA NOW IN FIRST 40 ROWS, FIRST 2 COL(S)
COMPUTING MEANS & STD. DEV.(S) ...
USE IDA COMMAND "NAME" TO PRINT OUT NAMES
> REGR
WANT EXPLANATION ? N
ENTER NAME OR COL NO. OF DEPENDENT VARIABLE : 2
HOW MANY INDEPENDENT VARIABLES ? 1
INDEP. VAR. 1 : 1
UPDATING CORR. MATRIX ...
COMPUTING REGRESSION ...
```

MATHEMATICAL SERVICES

CONVERSION OF SOME CYBER PACKAGES TO FTN5 EFFECTIVE JANUARY 9, 1984

In the June issue of *OFF-LINE*, we announced that many CYBER packages are being converted to FTN5, and gave a target date of

Monday, January 9, 1984

for the final conversion date. Before that date, FTN5 versions will be available via the FUTURE and GRAB/F commands for user testing. ***NOTE: After January 9, 1984 the old FTN versions will not be available except by special arrangement. Please convert your programs now!***

In previous issues we announced the conversion of various packages. The full group of products now available in FTN5 versions via GRAB/F and FUTURE is

IMSL	SLAM	FORSIM	GRG	FLEX	GMET
XMP	UOILIB	LINPACK	EISPACK	MINPACK	ITPACK
NETWORK	LSODE	AMOSLIB	GASP	MSL	ACSL

You should have converted your FTN programs to FTN5 by now, and should be using the FTN5

versions of these products. The FTN compiler, though reliable, is no longer supported by CDC; FTN5 is their "official" FORTRAN compiler now, and represents the new standard for the FORTRAN language. In deference to this, CSO would like to encourage the eventual "fading away" of the old compiler. At some point in the (not too near) future, we plan to alter the system so that the FTN control statement will access the new compiler and the FTN4 control statement will access the old one.

Manuals for FTN5 can be purchased at the CSO Distribution Office, 1208 W. Springfield. A utility, F45, is available for doing automatic conversions of some types of FTN statements to equivalent FTN5 forms. The manual for this also can be purchased at the Distribution Office; a copy of the manual is available for inspection at the CSO Systems Consulting Office.

IMSL EDITION 9.1

An update to the IMSL Edition 9 Subroutine Library has been received and installed on both the IBM and CDC systems. This brings the Library to the Edition 9.1 level.

The Edition 9.1 Library has been installed in FUTURE on the CYBER systems, and can be accessed by the command

GRAB,IMSL/F.

This version has been installed using FTN5, and replaces the FTN5 version of IMSL which was placed in FUTURE a few months ago. On January 9, 1984 (or thereabouts), when FTN5 versions of a number of packages in FUTURE become the "default", this FTN5 version will become the default IMSL accessed through GRAB; the old FTN version will be available only by special arrangement.

On the IBM, the IMSL Library currently exists only on the MVT batch system (which is available only on the VMD machine, if you are a CMS user). The Edition 9.1 Library has been placed in dataset SYS9.IMSL, and can be accessed by using LIBFILE='SYS9.IMSL' in the invocation of any standard FORTRAN procedure, e.g.

```
// EXEC FORTLDGO,LIBFILE='SYS9.IMSL'
```

The Edition 9.1 Library will become the default on the IBM at the end of January, 1984.

A list of the Library routines affected can be seen in the Systems Consulting Office, 1208 W. Springfield.

IMSL NEWSLETTER

We have received the September 1983 issue of IMSL User News, IMSL's periodic newsletter, telling of new IMSL products or information about current products. A copy is on display in the CSO Systems Consulting Office at 1208 W. Springfield. Additional copies are available from Stan Kerr, 179 DCL, 333-4715.

This issue contains a short article describing the reuse of PROTRAN programs, i.e., rerunning the same program with new data. (This is very easy on our CYBER systems -- after calling PROTRAN the first time, you can simply LGO the file which results, and later re-run this file without calling PROTRAN again. On some systems, it's not so easy as this!)

The newsletter also announces three new technical reports from IMSL:

#8205 "Addition of Points to Gauss-Laguerre Quadrature Formulas"
#8301 "Integer Programming with LP/PROTRAN"
#8302 "STAT/PROTRAN and FORTRAN: Some Examples"

Another example of the use of IMSL's finite-element package TWODEPEP is given in "Solving Diffusion and Heat Conduction Problems with TWODEPEP", and a new release of the EISPACK eigenanalysis package is announced -- EISPACK 3. This new release is a "portable" version which can be used on "any" computer; the previous version of EISPACK was released separately for each of several computer types.

VERSION 2.0 of SLAM II

Version 2.0 of the SLAM II simulation package has been received and installed in FUTURE on the CYBER systems. (The version we are currently running is 1.0, release 5.0.) Version 2.0 is being installed using the FTN5 compiler, and replaces the FTN5 version of SLAM announced in a previous *OFF-LINE*. SLAM II 2.0 can be accessed with the command

GRAB,SLAM/F.

This version will become the default on January 9, 1984, when various products are converted to FTN5 (see elsewhere in this issue). A document describing the changes in this version can be viewed at the Systems Consulting Office,1208 W. Springfield.

Unfortunately, with this version, SLAM has become too large to run during the hours 2PM-4PM, when CYBER memory limits are down. Your CENTRAL MEMORY limit must be at least 52000 (decimal) to run this version. (We did some experiments with "segmentation", but they were only a partial success. For your information, file SLAMSEG/UN=MATHLIB contains a set of loader segmentation directives which can be used with this version of SLAM, but we warn you that our own trials with these were not very successful.)

With this version of SLAM, we have created a CCL procedure called SLAMRUN, intended to simplify the process of running SLAM when FORTRAN code is supplied by the user. Below we will describe the methods used to run SLAM without this procedure, then show how SLAMRUN can make a SLAM run more convenient.

If no FORTRAN routines or user-supplied main program are to be used -- i.e., if you are doing a "pure" network problem with SLAM -- then the following control statement is used (as with the current version)

SLAM,xxx,yyy.

where ***xxx*** should be the name of a local file containing SLAM data, and ***yyy*** should be the name of a local file in which the SLAM report will be placed. If ***xxx*** is omitted (SLAM,,yyy), then data input is read from file INPUT. If ***yyy*** is omitted (SLAM,xxx), then the report appears on file OUTPUT (which is your screen in timesharing, and the batch output in a batch job). If both ***xxx*** and ***yyy*** are left out (SLAM.), then data is read from INPUT, and the report is placed on OUTPUT.

If you are supplying FORTRAN routines to be used in the SLAM run, in file ***rrr***, say, then the control statements used to run SLAM might be like this:

```
FTN5,REW,I=rrr,L=0,EL.
COPYL,SLAM,LGO,NEWSLAM,,AR.
NEWSLAM,xxx,yyy.
```

where ***xxx*** and ***yyy*** are, as before, the data input file and the report file output from SLAM. The COPYL control statement is used to merge your FORTRAN routines into the SLAM binaries and produce an updated binary NEWSLAM, which is then run.

NOTE: If you are supplying your own main program to drive the SLAM run, it must be called MAIN.

The above could be accomplished by the SLAMRUN procedure in either of two ways:

1. You can compile the routines yourself, then call SLAMRUN --

```
FTN5,REW,I=rrr,L=0,EL.
SLAMRUN,xxx,yyy,B=LGO.
```

2. Or you can request SLAMRUN to compile the routines for you --

```
SLAMRUN,xxx,yyy,PROG=rrr.
```

If the PROG parameter is used, SLAMRUN compiles the indicated file with FTN5. (It uses OPT=0, which results in fast compilation but inefficient execution, so for a "production" run on a large problem, method 1 above is better.)

A side effect of using SLAMRUN is that it builds an "absolute" binary for the program, which can be loaded very fast and cheaply into memory. If you have done one run with SLAMRUN and wish to do another run and change ONLY the data input, you can do so by entering

```
SLAMABS,xxx,yyy.
```

since SLAMABS is the name of the "absolute" binary file which is created.

SLAMRUN does not have to be used only in the case when you have FORTRAN routines to be used in the run. If you enter

```
SLAMRUN,xxx,yyy.
```

this behaves just like ***SLAM,xxx,yyy***, with the added benefit of creating SLAMABS, which you can then use for cheaper and faster loading on subsequent runs.

Random Number Generator Routine

A note concerning the random number generator used in SLAM -- all random events in SLAM are created using a random number generator routine called DRAND. As received, SLAM had a certain version of DRAND, but, as we have done with previous versions of SLAM, this was replaced by a version using the IMSL uniform random number algorithm. In this version, however, we have retained

the old version of the generator, under the name XRAND. If you wish to do a SLAM run using the XRAND generator, simply add DRAND=XRAND on the call to SLAMRUN, e.g.,

SLAMRUN, xxx, yyy, DRAND=XRAND.

Those who are interested in the effects of a particular random number generator on the outcome of a simulation may wish to use this feature. Also, we have a complete description of the operation of DRAND, for those who may wish to design their own replacement for DRAND.

VERSION 2G.5A OF SPICE CIRCUIT ANALYSIS PROGRAM

Version 2G.5A of the SPICE circuit analysis program from Berkeley, which was received earlier this year, has been installed on the CYBER systems in FUTURE. It can be accessed by the command

GRAB, SPICE/F.

This version is used in the same way as the old (current) version, but requires more memory, because the "dynamic memory" features of the new version have not yet been installed. To use this version, your CENTRAL MEMORY limit must be at least 48000.

Version 2G.5A will become the default version of SPICE in mid-January, before the beginning of the Spring semester.

Because this version uses a fixed internal array to store information about a circuit, it may fail on some large circuits. Please see Stan Kerr (179 DCL, phone 333-4715) if you are interested in making a larger version of SPICE.

This version came with a new manual, which has been reformatted for our use and placed under WRITEUP on the CYBERs. Until this version of SPICE becomes the default, the new manual can be obtained and printed by the following commands:

WRITEUP, SPICE/F.
PRINT, SPCDOC/AS/CC/EJ/FORMS=PPS.

The ***FORMS=PPS*** option is recommended, as the 60-page manual is easier to handle on the pre-punched 8.5 by 11 inch page printer forms.

TRANSACTIONS ON MATHEMATICAL SOFTWARE

We now have on tape all algorithms published in Transactions on Mathematical Software up to September 1983 (from algorithm 493 published in the first issue of March 1975, up to algorithm 604).

All TOMS algorithms are kept on a public tape; file TOMS/UN=MATHLIB contains a directory of this tape, plus directions for reading particular files on it. To print a copy of this directory, enter the following CYBER commands.

GET, TOMS/UN=MATHLIB.
PRINT, TOMS.

The "Math Note" series described in the March 1981 issue of *OFF-LINE* includes a note summarizing all algorithms published in TOMS to date. To print this note (it is note number 6), enter the following:

GET, MNOTES/UN=MATHLIB.
MNOTES, 6.
PRINT, OUT/AS/CC/EJ.

(To print instructions on the use of MNOTES, substitute a 0 (zero) for the 6 above.)

The algorithms published in the June 1983 and September 1983 issues are as follows :

In the June 1983 issue:

596: A Program for a Locally Parameterized Continuation Process, by Werner C. Rheinboldt and John V. Burkardt

597: Sequence of Modified Bessel Functions of the First Kind, by W. J. Cody

598: An Algorithm to Compute Solvents of the Matrix Equation AX**2 + BX + C = 0 , by George J. Davis

599: Sampling from Gamma and Poisson Distributions, by J. H. Ahrens, K. D. Kohrt, and U. Dieter

600: Translation of Algorithm 507. Procedures for Quintic Natural Spline Interpolation, by John G. Herriot and Christian H. Reinsch

562: Remark on algorithm 562. Shortest Path Lengths., by U. Pape

In the September 1983 issue:

601: A Sparse Matrix Package -- Part II: Special Cases, by J. M. McNamee

602: HURRY: An Acceleration Algorithm for Scalar Sequences and Series, by Theodore Fessler, William F. Ford, and David A. Smith

603: COLROW and ARCECO: FORTRAN Packages for Solving Certain Almost Block Diagonal Linear Systems by Modified Alternate Row and Column Elimination, by J. C. Diaz, G. Fairweather, and P. Keast

604: A FORTRAN Program for the Calculation of an Extremal Polynomial, by Frederick W. Sauer

The following articles which appeared in the June and September issues may be of interest to some of our users:

In the June 1983 issue

Modeling Languages Versus Matrix Generators for Linear Programming, by Robert Fourer

The Efficient Solution of Linear Complementarity Problems for Tridiagonal Minkowski Matrices, by C. W. Cryer

A Locally Parameterized Continuation Process, by Werner C. Rheinboldt and John V. Burkardt

In the September 1983 issue:

A Quantitative Evaluation of the Feasibility of, and Suitable Hardware Architecture for, an Adaptive, Parallel Finite-Element System, by Pamela Zave and George E. Cole, Jr.

The Numerical Solution of Separably Stiff Systems by Precise Partitioning, by David S. Watkins and Ralph W. HansonSmith

The Multifrontal Solution of Indefinite Sparse Symmetric Linear Equations, by I. S. Duff and J. K. Reid

If you are looking for software, Transactions on Mathematical Software is one of the journals to check, along with "Numerische Mathematik", "BIT", the "Computer Journal", and various publications of ACM (the Association for Computing Machinery) and SIAM (the Society for Industrial and Applied Mathematics).

We have, in the Systems Consulting Office at 1208 W. Springfield, bound volumes containing all the algorithms published in Communications of the ACM from 1960 to 1975 (when the algorithms department of CACM became Transactions on Mathematical Software).

ACSL VERSION 8D INSTALLED IN FUTURE

A new version of the ASCL continuous simulation language has been installed in FUTURE on the CYBER systems, and can be accessed by the command

GRAB,ACSL/F.

We plan to make this version the default version of ACSL (available through GRAB,ACSL) at the beginning of the Spring semester.

With this version of ACSL, we are introducing a few changes in the way ACSL is used on the CYBERs. (***Instructors, please take note.***) These changes will be described first below, after which we will describe other features of the new version.

1. ACSL 8D is being installed using the FTN5 FORTRAN compiler, rather than FTN. This means several things: the package itself, and all the subroutines used in it, have been compiled with FTN5; the user's ACSL model, after being translated to FORTRAN, is compiled with FTN5 rather than FTN, and therefore, any FORTRAN statements which are used within an ACSL model must conform to FTN5 standards.

2. The new version has been installed using memory management options which we have not used before with ACSL. The immediate effect of this on usage is the appearance in the ACSLCOM and ACSLGO procedures of a new parameter, TS, representing "table space". The default value of the TS parameter is sufficient for most small models. When the new ACSL exhausts its allotted table space, it will print a message to that effect; then, ACSLCOM or ACSLGO (whichever one failed) must be re-run with a larger TS value, e.g.

```
ACSLGO,TS=5000.
```

(The default TS for ACSLGO is 4000.)

Following is a list of the features in the new version of ACSL. Only features which are new are mentioned below; corrections to bugs are not listed. The complete newsletter from Mitchell and Gauthier describing this version is on view in the Systems Consulting Office at 1208 W. Springfield, or can be obtained from Stan Kerr (179 DCL, phone 333-4715). This newsletter also contains a new description of the ANALYZ linear analysis facility in ACSL and of the discrete event capabilities of ACSL.

1. ADD "ALL" TO PREPAR, OUTPUT, AND DISPLY
   The subcommand "ALL" has been added to DISPLY, PREPAR, and OUTPUT, allowing one to include all model variables in the command. When used with DISPLY (DISPLY "ALL"), a standard ACSL DEBUG dump is displayed.

2. FIRST DERIVATIVE EVALUATION (VARIABLE STEP ALGORITHMS)
   There is now an option to evaluate the derivatives at the beginning of each step with the state on the solution trajectory. This uses the following system variable:

FDEITG (default .FALSE.)	First Derivative Evaluation During Integration

   This only affects the variable step size algorithms, IALG=1 or 2. Normally the state is predicted or extrapolated over the step and then a corrector loop established with derivatives evaluated every time. Unfortunately the first or predicted state is the furthest from the solution trajectory and this is where the first flag (ZZFRFL) is made .TRUE. . With the new modification, when FDEITG is SET to .TRUE. , an extra derivative evaluation is made with ZZFRFL equal to TRUE with the states having the value at the end of the last step or actually on the solution trajectory. The penalty for this is the overhead on an extra derivative evaluation.

3. SPECIFY BEGIN CHARACTER OF SEQUENCE ON PLOTS
   When a series of plots is done, each curve is marked by a symbol. The symbol can be selected, or it is chosen by ACSL from a sequence starting at "A" that is incremented for each ordinate variable. The new version uses any specified character as the start of a new sequence, e.g.

   ```
   PLOT X,"CHAR"="X", Y, Z
   ```

   will plot X with X's, Y with Y's and Z with Z's since the specification of the first character establishes the sequence. To go through the centered symbols, use "CHAR"="%".

   The character can be specified prior to the first ordinate, which is useful with "ALL", e.g.,

   ```
   PLOT "CHAR"="1", "ALL"
   ```

   would plot nominally three curves per page (NPPPPL) with each curve using the characters 1, 2, 3 in sequence.

4. VALIDATE JACOBIAN
   Code has been added in order to validate the calculation of the Jacobian, which is used in the stiff integration algorithm (IALG=2) and also in the options on the ANALYZ command. This action is controlled by two system variables as follows:

CJVITG (default .FALSE.)	Check Jacobian validity
TJNITG (default 0.1)	Threshold for Jacobian non-linearity

   If this CJVITG flag is true then the derivatives are evaluated by perturbing plus, then minus, and then plus again so that two derivatives are obtained at nominally the same point. If the derivative evaluation is truly a function, then the results will be identical. If concealed state

variables are present these will be flagged by the error report saying the derivative evaluation routine is not repeatable. Concealed state variables are of the form:

```
X=X+Y
```

These types of calculations normally need initializing and if present cause large errors in the elements of the Jacobian matrix since the elements themselves are obtained by small differences due to perturbations on the order of the allowed errors.

As well as a repeatability check, a non-linearity check is included whereby the straight line between the two perturbed points is compared with the center. If F(x+dx), F(x-dx) and F(x) are the derivative values respectively then an error is reported if

```
ABS(0.5*(F(x+dx) + F(x-dx)) - F(x)) > ABS(F(x+dx) - F(x-dx)) * TJNITG
```

5. TIMES OF EXTREMA ON RANGE COMMAND
An option "TIMES" has been added to the RANGE command which causes the times associated with the extremal values to be listed, e.g.,

```
RANGE "TIMES","ALL"
```

(This assumes that the independent variable T is the FIRST variable on the PREPAR list; this is normal and recommended practice.)

6. MODIFICATIONS TO THE VARIABLE STEP INTEGRATION ALGORITHMS
Some modifications have been made to the variable step integration algorithms (IALG=1 and 2). These allow the algorithm to change step sizes and orders faster and, in particular, help the recovery after a discontinuity in the system is encountered. The typical improvement in execution time is about 15%. (These improvements are due to suggestions made by Professor William Gear of the University of Illinois.) For a detailed description of the changes, see the newsletter referred to earlier.

7. EXTENSIONS TO ANALYZ COMMAND
The options on the ANALYZ run-time command have been extended in order to accomplish the following: control variables and observable variables can now be defined; ANALYZ commands are cumulative and must be specifically deleted via a "CLEAR"; the A, B, C and D matrices can be written out to a file for further processing; the format of the Jacobian printout has been modified to help visualize large systems; the eigenvalue finder from EISPACK has been incorporated into the ACSL Library, removing the dependence on subroutine EIGRF of the IMSL Library. These changes are described in more detail in the newsletter mentioned earlier.

8. ADDITION OF STATE EVENT AND TIME EVENT FINDER
A capability has been added to synchronize the integration stepping method to externally defined events, either occurring at a specified time or when a specified function crosses zero. The intent is (eventually) to added a SCHEDULE statement to the ACSL model definition syntax. Mitchell and Gauthier, the vendors of ACSL, have been considering the most reasonable design of this feature for some time; this version of ACSL implements their latest ideas as ACSL macros.

There are two macros,

```
SKEDSE(realexpression, block, flag)
```

which is a "state event finder", and

```
SKEDTE(realexpression, block, flag)
```

which is a "time event finder". The state event finder causes the named DISCRETE block to be entered when realexpression crosses zero; the flag is set .TRUE. when the "event" has occurred. The time event finder causes the named DISCRETE block to be entered when the independent variable (usually T, or time) attains a value equal to realexpression; again, the flag is set .TRUE. when the time event has occurred. The newsletter from Mitchell and Gauthier describing this feature can be consulted for further details.

9. RESTART VARIABLE STEP INTEGRATION ALGORITHMS

A subroutine RSTART has been included in this version, allowing the user to explicitly request restarting of the variable step integration algorithm. This may be particularly useful in handling discontinuities, when a user-controlled restart may be more efficient than the automatic techniques used in ACSL. The routine is called as follows

CALL RSTART(name, hnew)

where

name is the name or integer number of a DERIVATIVE block

hnew is the step size to use for the first step after restart. If ***hnew*** is 0, the old step will be retained (but may be reduced by the error control algorithm).

RSTART must be called from a DISCRETE block that is normally used to handle state and/or time events which would usually introduce discontinuities.

10. CHANGE FORMAT OF 'VARIABLE' STATEMENT

The VARIABLE statement can be used to change the name of the initial condition variable but the format given in the manual doesn't work. This is

```
VARIABLE  name=realconstant , icname=realconstant
```

In fixing this it became obvious that this is a clumsy way to express the information since the realconstant value had to be specified twice. The new forms are:

```
VARIABLE X            $ "X is independent variable"
VARIABLE X = 1.0      $ "X will start at 1.0"
VARIABLE X, XIC=1.0   $ "ICname is XIC"
```

where the third form is new. The default name for the initial condition value for the independent variable is ZZTICG which can be changed at a pinch, but choosing your own name is preferable.

DOCUMENTAION

NEW AND REVISED DOCUMENTATION

Following is a list of new and revised CSO documentation. Some of the Reference Guides and Technical Notes announced below may still be in the printing process at the time of this publication; however, if this is the case, they should be available to users shortly.

Reference Guides

RF-0.9	TELENET on the CYBER 175	Revised 11/10/83
RF-5.7	DI-3000 Library	New 10/21/83
RF-5.8	GRAFMAKER	New 10/21/83
RF-5.9	PVI Contouring System	New 10/21/83
RF-5.10	The DI-3000 Metafile Facility	New 10/21/83
RF-11.10	Submitting an IBM OS/MVT Job on the CYBER	New 11/21/83

Technical Notes

TN-107	TAGS - CYBER Mailing Labels Program at the University of Illinois	New 10/09/83

On-line Documentation

Several on-line documents have been added over the last few months that may be of interest to our users.

IBM PC NETWORK: CYBER INTERFACE. An interface program that allows the IBM PC user to communicate with the CYBER systems through a communication line. The writeup may be obtained and printed by entering:

```
WRITEUP,IBMPC.
PRINT,IBMDOC/AS/CC/EJ.
```

APPLE: CYBER INTERFACE. An interface program that allows the APPLE user to communicate with the CYBER systems through a communication line. The writeup may be obtained and printed by entering:

```
WRITEUP,APPLE.
PRINT,APDOC/AS/CC/EJ.
```

SHAZAM. An econometrics package that is relatively easy and inexpensive to use. Has capabilities for regression analysis, simultaneous equation estimation, principal components, factor analysis, analysis of variance, sorting and plotting variables, among others. The writeup can be obtained and printed by entering:

```
WRITEUP,SHAZAM.
PRINT,SHZDOC/AS/CC/EJ/FORMS=PPS.
```

TAGS. A CCL-based procedure for creating, printing and storing mailing labels and lists on the CYBER. The writeup can be obtained and printed by entering:

```
WRITEUP,TAGS.
PRINT,TAGDOC/AS/CC/EJ.
```

HELP WANTED - SALES

STUDENT PROGRAMMER WANTED

Student Programmer, Apple Basic, prefer 15-20 hours per week; should be available between semesters and summer. Please contact: Gary Rossman at 333-8380.

HALF-TIME PROGRAMMER

Desired: Half-time programmer. Knowledge of PRIME 750 and PRIMOS computer system desirable but not required. Will be using above system along with CSO's CYBER 175 and IBM 4341s, a DEC VAX 11/750, IBM PCs, and an APPLE II. The work on the PRIME system will involve the use of high level cartographic and relational data base management systems in conjunction with all types of natural resources data and information.

Please contact: Bob Sinclair, Illinois State Water Survey, 51 Water Resources Building; Telephone: 333-4952.

QUARTER-TIME ASSISTANTSHIP

Quarter-time Assistantship available. We need a student with programming experience. Job functions include: programming, coordinating with CSO, some data entry.

Please contact: Betty Wagoner, Small Animal Clinic; Telephone: 333-1988.

FOR SALE

For sale: Digital DecWriter III Teleprinter, LA 120 AA; used very little; $1700. Call: 333-0044

OFF-LINE's Mailing List

If you wish to be placed on our mailing list for future issues of *OFF-LINE*, if you wish to be removed from the list, or if you wish to enter an address correction, please complete and return this page. **(Current subscribers are kept on the mailing list until a specific request for removal is received, or until a mailing is returned as undeliverable.)**

Check one:

- ☐ Place my name on mailing list
- ☐ Make the following corrections or changes
- ☐ Delete my name from mailing list

_____

First name -- Initial -- Last Name

Campus Address:

_____

Room - Building

_____

Department

Off-campus Address:

_____

Organization or Company (if applicable)

_____

Street Address

_____

City -- State -- Zip Code

If address correction, give old address and zip code below.

_____

_____

_____

12/83

SEND TO:

OFF-LINE
150 Digital Computer Laboratory
University of Illinois at Urbana-Champaign
1304 West Springfield Avenue
Urbana, Illinois 61801