University Computer Center Newsletter

University of Minnesota
Twin Cities

September 1983
Volume 17, Number 9

All Pascal Users Take Note!

We will rotate the versions of Pascal-6000 on Wednesday, September 14, 1983. The FUTURE version of Pascal will become the current, default Pascal compiler accessed by the PASCAL control statement. The current version of Pascal will be accessible only as a PAST product. The current version of Pascal is not upward-compatible with the new version. Some changes to user programs may be necessary.

The most obvious incompatibility is that the control statement you use to call Pascal will have different parameters. If you try to invoke Pascal in the old way, with the old order-dependent parameters, you will get error messages.

The indexed writeup PASCALF describes the differences between the current and future versions of Pascal-6000. See WRITEUP(PASCALF) for a list of the chapters in the writeup, or WRITEUP(PASCALF=*) for the entire writeup (30+ pages).

The new Pascal conforms to the ISO standard for Pascal, and has several new features. The most visible changes are summarized below.

ISO-Standard Required Changes

1. Standard switch (compiler option S+/S–; S+ is the default). The compiler now prints out warning messages indicating the use of nonstandard features in Pascal-6000 when the "S" option is turned on.
2. New syntax for conformant-array parameters. DYNAMIC is no longer a reserved word; LOW and HIGH are no longer used.
3. Enforcement of new scope rules for identifiers.
4. Implementation of more compile-time and run-time checks.
5. Identifiers are unique to their length. Previous versions of Pascal-6000 compared only the first ten characters of an identifier.
6. Restrictions on for-statement control variables.
7. Requirement of complete specification of variant records.
8. Formatted writes must use field widths greater than zero.
9. The type TEXT is no longer equivalent to FILE OF CHAR.

Other Changes

1. Pascal control-statement format. The new Pascal uses a control-statement format that conforms to the structure of other Cyber language processor control statements.

   PASCAL(I=sfn,L=lfn,B=bfn,
   GO,PD=8,PS=n,PL=x/options)

   See WRITEUP(PASCALF=ACCESS) for a more detailed description of these parameters.
2. Automatic field length management. Programs that need more memory will automatically request more from the system. The old R and W compiler options are replaced by a set of M options: MB, MD, MF, MI, MR, MS, MV, and MX.
3. Elimination of the LINELIMIT procedure. The LINELIMIT concept only applies to OUTPUT and is now specified by the control-statement parameter PL.
4. Compiler error messages in English, French, and German. Using the compiler option D, you can specify an alternate language to print error messages. For example, **(*$D'GERMAN'*)** will select German error messages.
5. Bug fixes and better code generation. Bugs found while writing the ISO-Standard related modsets and those found while running the Validation Suite were fixed.

(Dave Bianchi)

System News/Notes

Correction

The list of keypunch facilities in the August *Newsletter*, although correct at the time of publication, has since changed. There are now two keypunches in 131 Experimental Engineering, two in 140b Experimental Engineering, and four in 14 Folwell Hall.

IMS Journal

SIR/DBMS CONVERSION DATE SET

Our current version of SIR (data base management system) on the CYBER systems is no longer being supported by SIR, Inc. We will replace Version 1.1 with Version 2.1.1 on October 25, 1983. The two versions are not compatible. You must convert your files from one version to the other using the TRANSPORT FILE utility if you choose to use the new version. You can get information on this utility by calling the Data Base HELP-Line (376-1761, 1-3 p.m., Monday-Friday) or stopping in 208 Experimental Engineering on the Minneapolis campus.

The future version of SIR is now available with the control statements:

FETCH(SIR)
SIR.

We encourage you to access this version now and re-create your data bases with the next release of SIR. SIR Version 2 is a significant upgrade of SIR Version 1. Some commands are no longer valid. They include FOR EACH REC, FOR EACH CASE, MOVE VAR LIST, WRITE IF, and SELECT statements. Replacements for these commands are explained in the document *Converting from SIR/DBMS 1.1 to SIR/DBMS 2*, available in 208 Experimental Engineering, or by calling the Data Base HELP-Line.

After the conversion, you will access Version 1 with the control statement **SIR1**. You will access Version 2 with the control statement **SIR**. FETCH(SIR) will no longer be necessary.

SIR USERS' GROUP MEETING

A SIR User's Group has been formed at the University Computer Center. The group includes both experienced and beginning SIR users. They have met over the last few months to discuss various aspects of the SIR data base management system. Typical discussions center around problems they encounter when writing retrievals in the new version, or ways to work around bugs in the software. The next meeting is scheduled for 3:30 p.m. September 21, in 128 Management/Economics Building, on the west bank of the Minneapolis campus. The October meeting will be at 3:30 p.m. October 12, in 110 Management/Economics.

If you'd like to be put on the mailing list for these meetings, write to Jerold Hahn, 513 North Central Forestry, 1271 Raymond Avenue, St. Paul campus, St. Paul, MN 55108, or call the Data Base HELP-Line at 376-1761 (1-3 p.m. Monday-Friday).

News/Notes to 93

FORTRAN Compiler Validation

We recently tested UCC's M77 FORTRAN compiler (version 2.4) against the recently released 1978 FORTRAN Compiler Validation system. This article discusses that process and its results.

Background and Methods

All users of a computer language on a specific system must be assured that the current compiler implementation satisfies their program's requirements. This assurance includes:

- correct execution code produced from the source statements of the program
- informative and complete diagnostics produced for incorrect language usage
- full accuracy for the arithmetic operations and functions provided by the language
- availability of debugging and post mortem dump analysis for program failures
- program execution speed comparable to current industrial implementations of the language for similar execution speed processors

Historically, shops with a few dominant production programs would run selected known data sets of the programs against a new language version to ensure that the new version obtained the correct answers. Another method was to save collections of sample programs with known answers to run against a new language version. Both methods suffered from incompleteness (a user might try to use a standard language feature not tested in the sample programs) and output verbosity (the sample programs usually provided many pages of sample results that had to be checked by hand or by another computer program against the "standard" expected output).

As implementers of MNF and M77 FORTRAN compilers, we needed better tools to ensure that each new version was indeed a giant step forward rather than two steps sideways and a minor back step. So E. J. Mundstock of the UCC staff designed four validation programs, VALIDA through VALIDD, that checked 1966 FORTRAN Language standards and special MNF features internally. These validation programs only provided error messages for specific failures; they have been used to ensure that each new MNF and M77 compiler version met minimum standards. For M77 we implemented a complete set of DOUBLE PRECISION and COMPLEX arithmetic functions to complement the REAL functions developed for MNF. To test all these functions a set of programs for each type was developed to ensure functional accuracy, execution speed, and error diagnostic ability.

At the same time the federal government (a large purchaser of computing systems) developed validation suites of programs to ensure that compiler manufacturers met language standards. COBOL came first, followed by a 1966 FORTRAN standard version that became a 1978 FORTRAN subset standard. These validation suites were tested by the Federal Software Testing Center against manufacturers' compilers, and the Validation Summary Reports (VSRs) were printed in the National Technical Information Service's abstract newsletters on *Computers, Control and Information Theory*. When the 1978 FORTRAN Compiler Validation System became available early this year, we purchased it to check and provide confidence in the FORTRAN compilers available at UCC. (The enthusiastic Pascal community implemented its own validation suite; it is used to validate our Pascal compilers.)

But a validation suite only provides approximate assurance that all standard features are implemented and that correctly written source statements produce correct results. It does not check that incorrect source statements or constructs are diagnosed nor does it report on execution speed or debugging ability. Since M77's forte was to provide maximum information per run and good error diagnostics, we have also been involved in writing a suite of programs to fully test all 450 M77 error messages and the 1 to 20 ways that each one can be produced. This error message suite will also be used to check out each new version of the compiler. You may notice the results of this effort in the additional messages generated by the 2.4 version of M77 made current in late August.

M77 errors pointed out by the validation suite and since corrected

All compiler writers would like users to find bugs in a compiler with a program that has no more that 20 statements and at most one record of input. Often real life implies that this 20 statement program is a null set, but it is usually possible to construct such a program after a few hours or days of effort. It was gratifying to run the 1978 FORTRAN Compiler Validation System and come up with about 25 errors that were easily localized to about 20 statements. Most of the errors were corrected and the validation suite run again to produce the 2.4 version of M77.

Some of the version 2.3 M77 bugs that were discovered by the validation suite included the following:

1. M77 had been designed to look at the set

   (item)

   as redundant parentheses around an item to be eliminated. The validation suite produced the legal construct

   name (subscript) (:)

   which M77 carefully changed to

   name (subscript) :

   and then gave an error for the colon sitting outside a parenthesis group. M77 now changes

   name (subscript) (:)

   to

   name (subscript)

2. EQUIVALENCE referencing a CHARACTER COMMON block that was defined in another routine without the EQUIVALENCE showed up the fact that in CHARACTER EQUIVALENCE, M77 assigned storage correctly but computed the block size as a number of characters' value rather than words plus offset. This meant that EQUIVALENCED CHARACTER COMMON blocks were 10 times longer than required and in the validation example a FATAL error message about longer block was given.
3. M77 translates all hollerith, octal, and string data as BOOLEAN constants and translates to CHARACTER only if the constants were associated with a CHARACTER variable in a binary operation. Thus M77 did a BOOLEAN relation for ' '.LT.'A' rather than the ASCII comparison required. Long BOOLEAN constant string comparison is now done as an ASCII character compare.

M77 non-conformance with the 1978 FORTRAN standard

Deliberate input/output violations

1. The standard 13.5.9(5) states "On output if the number of characters produced exceeds the field width or if an exponent exceeds its specified length..., the processor will fill the entire field of width w with asterisks." M77's design of maximum information per run means that for REAL or DOUBLE PRECISION fields of width w=8 or greater, M77 produces a single asterisk followed by an E field [E(w–1).(w–7)], so that another run does not have to be made. Similarly for I fields, the leading digits are given after a single *.
2. The CDC CYBER systems have special REAL values for infinite (R), indefinite (I), not normalized (N), and zero (0). M77 puts out that single character rather than following the field specification. A single X character is put out for an I field whose corresponding list element is greater than or equal to 2**48.
3. The standard 13.5.9(3) states "However, the processor must not produce a negative signed zero in a formatted output record." Use of a single value 0 in item 2 above allows M77 to produce additional information in F fields. For example the internal values of 0.0, –.0043, and .0037 produce the corresponding output values of 0, –.00, .00 under an F4.2 field rather than the uniform .00 required by the standard. This allows differentiation between true zero and signed non-zero values having no significant digits for the given field.
4. M77 does not allow repeated list directed string or complex input data to extend over more than one record. M77 with a single record input buffer gives a fatal error message for this case.
5. The standard is inconsistent in the number of significant digits to put out in a Gw.d field. If the F field applies, then d digits rounded in the d+1 place are output; otherwise, when the E field applies, d significant digits are output unless a 1P or greater scaling is in effect, in which case d +1 significant digits should be produced. M77 always produces d rounded digits for this case.

FORCE STORES (FS) parameter required for entities known by two or more names

6. MNF and M77 were designed for parallel operating CYBER processors such as the CYBER 74 and 7600. For both systems an average of 10% execution improvement is obtained if the loads of operands to the fast registers is done before the store of the previous operation from a fast register to central memory. This means a load of an EQUIVALENCED, COMMONed, or DUMMY ARGUMENT operand will be done before the last operand store takes place for operands having different names but physically in the same location. To avoid errors of these equivalenced names in the FORTRAN 78 validation suite, M77(FS) was required. Thus if EQUIVALENCE is used in a FORTRAN program to share data, use the FS parameter.

Uncorrected M77 validation errors with no error diagnostics

The following legal FORTRAN construct produced bad execution code without an error diagnostic and thus is a serious error.

7. Lower bound expressions (i.e., more than a simple name or constant) in adjustable dimensions are incorrectly computed.

Uncorrected M77 validation errors that produce error diagnostics

The following legal FORTRAN constructs are not allowed, but produce error diagnostics.

8. Concatenation is not allowed for the FMT= specifier.
9. Substringed concatenation elements are not allowed in I/O lists or as actual arguments.
10. Files other than INPUT, TAPE5=INPUT, OUTPUT,TAPE6=OUTPUT must be on the PROGRAM card, i.e., M77 currently does not have dynamic opens of files. In addition, the maximum record length for any OPEN in a program must be specified at least once on the PROGRAM card in the form File=/maximum record length.
11. Adjustable dimension array names must be dummy arguments on the SUBROUTINE or FUNCTION or ENTRY statements before the adjustable dimension declaration. They are not allowed to be only on ENTRY statements after the declaration of the adjustable dimension.

(L. Liddiard and J. Eberhart)

A NOTE ON OUR COVER

The gentleman with the penetrating gaze on this month's cover is Blaise Pascal (1623-1662), French mathematician and philosopher. He developed one of the earliest calculating machines and the Pascal language was named in his honor.

SYSTEM 2000 CONVERSION DATE

The future release of System 2000 (Version 2.80) will become current on October 4, 1983. The current release (Version 2.60) will become a PAST version. Data bases that have been created under Version 2.60 are upward compatible with Version 2.80. You need not make any changes to your data bases to access them with the interactive language. You *must*, however, change the terminator used to end commands from a colon (:) to a semicolon (;). This change complies with the new documentation produced by Intel Corporation.

New features include a WHERE-clause CONTAINS operator, which allows text search within character values. Disjoint data sets can be accessed in Report Writer, with the REPEAT FOR <SR> and END PATH WITH <SR> statements. Release 2.80 contains new terminology, but the old terminology is still supported. NAME item types have been changed to CHAR item types. REPEATING GROUPS (RG) have been changed to SCHEMA RECORDS (SR).

These changes and others are discussed in the *System 2000 Newsletter - Release 2.80* available for one dollar in the Computer Store. We urge everyone who uses the current version to pick up a copy of this newsletter. All documentation sold in the Computer Store is written for Release 2.80. We have the next release available as a FUTURE version. To access it now, use the control statements:

```
FUTURE(S2000)
S2000.a
```

After the conversion, the control statement **S2000** will access the Version 2.80 software.

SYSTEM 2000 GROUP TO MEET IN MONTREAL

The Association of System 2000 Users for Technical Exchange (ASTUTE) will hold its fall meeting in Montreal, Quebec October 19-22, 1983. ASTUTE provides an opportunity for System 2000 users and technical staff from Intel Corporation to share information and ideas. Detailed information on the convention program and arrangements is now available. Direct your questions about the organization to Cheryl Vollhaber at UCC (376-1761, from 1 to 3 p.m. weekdays) or to ASTUTE President Roberta A. Armstrong (373-2106). If you would like to be on the ASTUTE mailing list, call Dr. Armstrong.

(Cheryl Vollhaber)

Consulting

NON-TRADITIONAL/TEXT AND WORD PROCESSING CONSULTING SERVICE

Beginning in fall quarter, we will offer a new consulting service in the areas of non-traditional computing, text analysis, and text and word processing. Consultants will be available to answer your questions about these areas from 1-3 p.m., Monday through Friday, in 124 Shepherd Labs. You can also have your questions answered by phone during those hours; call 376-2944.

(Lenief Heimstead)

Graphics

NEW VIEW PROGRAMS

New VIEW programs are now available as future products on the CYBER CA, the MERITSS CYBER (ME), and the VAX (VA). VIEW displays plot metafiles on a variety of graphics terminals. These new programs feature many new options: you can browse interactively through a multi-image metafile, or specify how your colors should appear, for example.

FUTURE VIEW supports the DEC VT100g and the Ramtek 6211, in addition to the Tektronics 40XX series.

For complete information on these new utilities, see WRITEUP(VIEWF) on the CYBERs or HELP VIEW FUTURE on the VAX.

(Michael Regan)

Terminal Information

REMOTE JOB ENTRY SURVEY

UCC is distributing a survey to help us meet your remote job entry (RJE) needs in the near and distant future. The survey will be mailed to all current RJE site coordinators, but we also want information from prospective users. If your organization does not have an RJE at the present time, but may set one up, please contact Joe Cornell (376-2703) for a copy of the survey.

CYBER Notes

TELENET CHANGES

Information has been mailed to current Telenet users concerning upcoming changes. If you use Telenet, and have not received this information or if you have questions about Telenet service, please call Liz Stadther at (612) 373-4920.

NOS 1.3-485 LEFTOVERS TO BE REMOVED

Our current CYBER operating system is NOS 1.4-552. Prior to March 20, 1983, our CYBER operating system was NOS 1.3-485. We ran articles in the February, March, and April *Newsletters* that described the few areas of incompatibility between the two systems and told you what parameters you must change. In fact, we temporarily installed these old parameters and commands in the new version of the system to smooth the transition. On Wednesday, September 14, we will remove these temporary patches permanently.

The MI parameter (for specifying the machine ID of the machine on which a job should run) will be removed from ROUTE, SEND, SUBMIT, ARCLIST, RESTORE, QUEUE, and the job card. In its place, use the ST parameter, which specifies the logical ID (LID) of the machine on which you want a job to run. Form the logical ID with the letter "M" and the machine ID. If, for example, you have used

```
SUBMIT(FILE,MI=CR)
```

you should now use:

```
SUBMIT(FILE,ST=MCR)
```

instead.

If you still use the old NOS numerical machine IDs, these will no longer be available. Exhibit 1 lists the old machine IDs and their re-

Old MI=ID	Current ST=LID	Meaning	Current job card parameter
64	ST=MME	MERITSS (instructional CYBER)	STMME
72	ST=MCA	CYBER A (research CYBER)	STMCA
74	ST=MCB	CYBER B (research CYBER)	STMCB
CRAY	ST=MCR	CRAY-1	STMCR

Exhibit 1. Old NOS numerical IDs and their replacements.

Old control statement	Current control statement	Meaning
ECS(xxx)	RFL,EC=xxx.	Request ECS field length
RTF.	RETURN,*.	Return all local files
RWF.	REWIND,*. or R,*. or R.	Rewind all local files
ULF.	UNLOAD,*. or U,*.	Unload all local files

Exhibit 2. Non-standard NOS control statements that will be deleted September 14, and their replacements.

placements. Thus, if you have used:

SUBMIT(FILE,MI=74)

you should now use:

SUBMIT(FILE,ST=MCB)

instead.

We will also remove several non-standard control statements that we have simulated using CALLPRG. These control statements and their replacements are listed in Exhibit 2.

(A. B. Hastings)

PL PARAMETER CHANGE

On August 14, we changed the function of the PL parameter that is used on several CYBER permanent file commands. The parameter no longer accesses reference files on the catalog of the LIBRARY account.

Since the PL parameter is used mainly for system maintenance, the change does not affect many users. If you do use it, however, you can replace it with the PN and UN parameters. For example, replace:

GET(FN/PL=0)

with

GET(FN/PN=0,UN=LIBRARY)

If you have questions about this change, call Marisa Riviere, 376-9718.

M77 VERSIONS HAVE CHANGED

We rotated the versions of M77 on Sunday, August 28, 1983. Future M77 (2.4) became current, and current became past. Most of the differences between future and current are the result of bug corrections. You may, however, notice additional error diagnostics. See the article "FORTRAN Compiler Validation," elsewhere in this issue of the *Newsletter*.

(J. Eberhart, 373-5907)

FORTRAN SUBSYSTEM CHANGES

In late August, we made some changes to two FORTRAN subsystems, MNFTS and FORTRAN, and removed TSF. The MNFTS subsystem had used a subset of MNF (TSF) that:

- did not produce a cross reference map
- did not list object code
- did not optimize DO loops
- did not issue compiler DAYFILE messages
- did not support DOUBLE PRECISION or COMPLEX numbers

We removed the TSF control statement and the MNFTS subsystem now uses the full MNF compiler. The only user-visible change is a slight increase in the field length necessary to run a FORTRAN job under MNFTS.

The FORTRAN subsystem now calls the Minnesota 1977 FORTRAN compiler (M77) rather than MNF. Programs created using the FORTRAN subsystem prior to fall quarter continue to use the full MNF compiler. Differences between MNF and M77 are documented in WRITEUP(M77).

(Janet Eberhart)

COBOL 4 DISAPPEARS THIS MONTH

We will remove the COBOL 4 compiler from the system on Sunday, September 25, 1983. You must convert your COBOL 4 programs to COBOL 5 by then.

WARNING: When the new CYBER 825 replaces the CYBER 74 (system CB) COBOL 4 will no longer work correctly. The compiled code from COBOL 4 modifies instructions, then attempts to execute them. This technique can fail on the lower 800- series of Control Data CYBER computers. Any of you with COBOL 4-compiled binary programs will have this problem.

If you have such a program that you still want to run on the CB system, include the control statement:

MACHINE(EP=ON)

This control statement inhibits the instruction look-ahead features in the 800-series hardware and solves the problem described above. It also increases the program execution time by about 50%, however.

If you have questions about this change or need assistance, call our Information Management Systems Group at 376-1761 (from 1 to 3 p.m. weekdays). COBOL 5 documentation is available at the Computer Store. We will also remove CYBER Record Manager Advanced Access Methods Version 1 at that time.

(Systems Staff)

VAX Service

USING TAPES ON VMS

Using tapes on VMS in the standard, "stock" system way is a many-step process. To use a tape on VMS, you must:

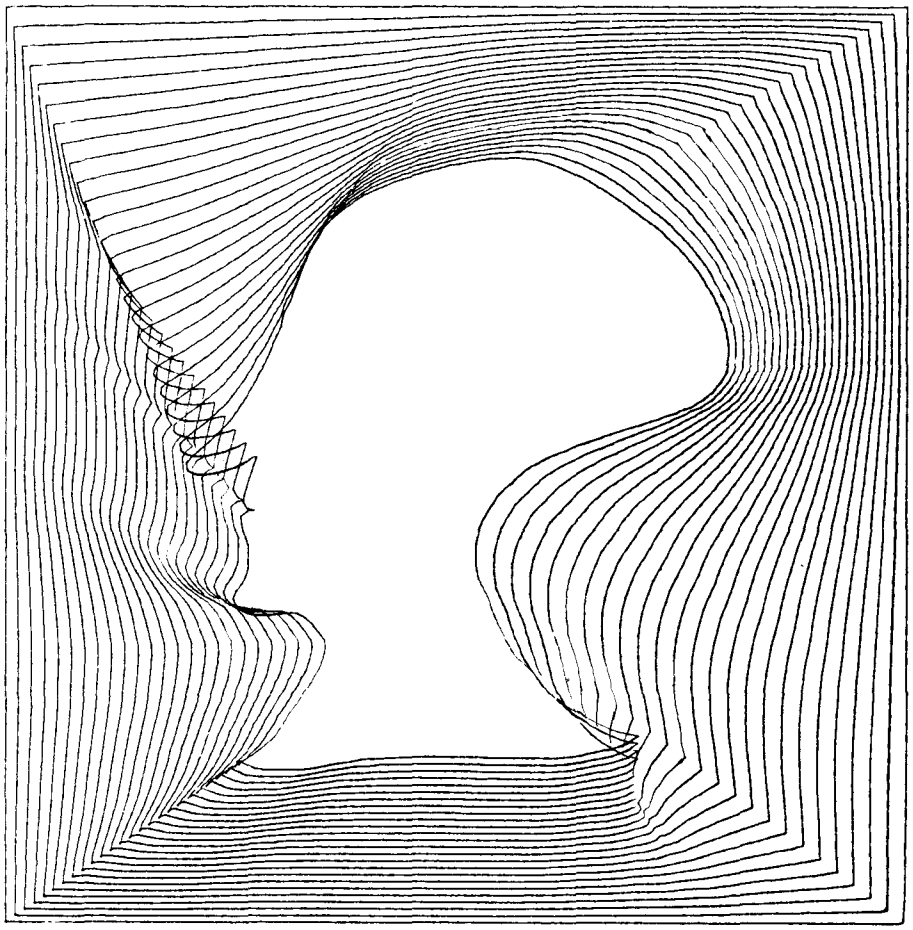
1. Establish a tape drive as owned by your process: the ALLOCATE command.
2. Get the operator to physically mount the right tape on the tape drive, with the write ring in, and the density set correctly: the REQUEST/REPLY command.
3. Logically mount the tape with the proper characteristics: the MOUNT command.
4. Use the tape in whatever way you intended: COPY, BACKUP, etc.
5. Logically dismount the tape: the DISMOUNT command.
6. Get the tape physically dismounted from the drive (there is no command to do this explicitly).
7. Give the tape drive back to the system so that others can use it: the DEALLOCATE command.

Because there are so many steps, none well-documented, there has been a great deal of confusion about using tapes on VMS. We plan to simplify and standardize the procedure. On September 12, we will:

- add two commands to the system: GETTAPE and DROPTAPE. GETTAPE will perform steps 1 and 2 above; DROPTAPE will perform steps 5, 6, and 7
- require that users specify a tape ID as well as its VSN
- put up documentation explaining how to use tapes under MOREHELP TAPES.

After September 12, everyone should use the GETTAPE command to allocate the drive and get their tape physically mounted. The general form of the GETTAPE command is:

GETTAPE vsn id logicalname /[NO]WRITE /DENSITY=xx /[NO]WAIT



—Graphic by the UCC Graphics Group

In this command, **vsn** and **id** identify the tape, and **logicalname** is a name (commonly TAPE) that you use to refer to the tape; if you use it you need not know which tape drive you were given. After GETTAPE has retrieved the tape, you MOUNT it and proceed as before. When you have finished using the tape, use the DROPTAPE command to dismount and deallocate the tape and drive.

If for some reason you cannot use GETTAPE and must request a physical tape mount with the REQUEST/REPLY command, you must follow a rigid form after September 12 or your tape will not be mounted:

"Please mount VSN=vsn, ID=id at xxx BPI for (reading/writing) on MTAn"

As before, to use a tape at all, you must first enter it into UCC's transient or permanent tape libraries, just as you do for the CYBERs. The *Guide to Magnetic Tape Usage* describes the procedure. It is available for use in our Reference Room, 140 Experimental Engineering, or for purchase at our Computer Store.

For more information on these procedures, enter the command:

morehelp tapes

(Stuart Levy)

VMS SYSTEM UPGRADE

We upgraded the VAX to VMS version 3.3 on August 21. The upgrade consisted mainly of bugfixes and maintenance changes with some adjustments to the BACKUP facility. There are no major documentation changes. For more information, print or type the file:

usera:[writeups]vms33.lis

Grants for Research

NEH EDUCATION PROGRAMS

An article by Christina Bitting and Mary E. Stephens in the March 1983 issue of *Grants Magazine* (Volume 6, Number 1) discussed a major redesign of the education programs of the National Endowment for the Humanities (NEH). The broad goal of the new programs is to improve both the quantity and the quality of humanities study required of students.

According to Bitting and Stephens:

A central theme of the new guidelines is an emphasis on the improvement of existing activities, in the context of overall institutional or regional/national needs, rather than on the creation of new programs.

. . . The means by which existing programs can be improved are also greatly expanded. Although the new guidelines do suggest a variety of ways in which NEH funds can help to strengthen humanities programs (such as faculty workshops, sabbatical programs, or expansion of library holdings), they leave the specific means to be used to the judgment of each applicant.

Specific NEH education programs include: Central Disciplines in Undergraduate Education, Exemplary Projects in Undergraduate and Graduate Education, the Humanities Program for Nontraditional Learners, Teaching Materials from Recent Research, and Humanities Instruction in Elementary and Secondary Schools.

If you would like to read the complete article, *Grants Magazine* is available in the UCC Reference Room, 140 Experimental Engineering. For additional information about the programs, contact:

Division of Education Programs
National Endowment for the Humanities
1100 Pennsylvania Avenue NW
Washington, DC 20506
(202) 786-0373

Microcosm

LAB CARDS

Anyone who uses the microcomputer labs in 14 Folwell Hall and 160 Architecture must have a lab card. You can purchase a card at UCC's Computer Store. The charge is $10/quarter.

Because of a change in lab fee policies, anyone with a card from fall quarter 1982 or winter quarter 1983 may exchange it for a card for this fall or winter at no charge.

Statistics

BMD SERIES IS NO MORE

The forerunner of the present BMDP statistical programs was a series called BMD. Use of BMD declined steadily since BMDP's implementation several years ago. For that reason, the BMD series has been removed from the system. BMDP81 is now the current version. MINITAB81 (formerly available as the future version) is now current also.

(Patricia C. Bland)

Writeupdate

CHANGES IN WRITEUP

On September 14, the internal index file used by the writeup utility will change. Most users will not notice any difference; however, if you have private writeup indices, you must change them to match the new format. For the first three months following this change, you will be able to use both the old and the new formats; after that, however, only the new format will be accepted.

The change ensures that when you use the DT (date) parameter to find out what writeups have changed since a certain date, you will receive an accurate report. All changes made to a writeup, whether major or minor, will result in an updated version. WRITEUP(INDEX) will also reflect the accurate dates.

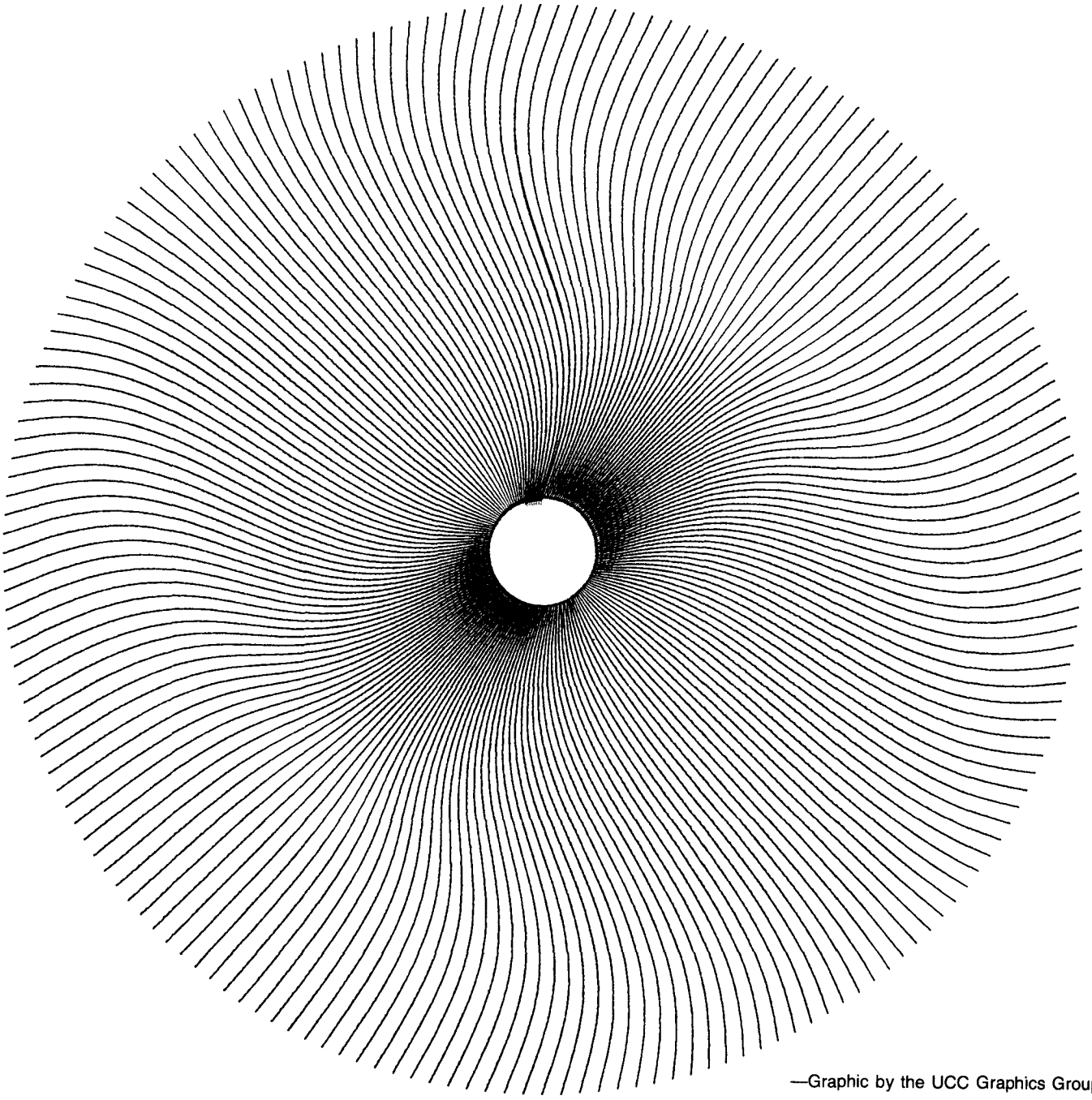
WRITEUP(WRITEUP)	Index for writeup.
WRITEUP(WRITEUP=FORMAT)	New directive file format, changes from old format.
WRITEUP(WRITEUP=WRITEUP)	Introduction to WRITEUP (similar to what you get with WRITEUP).

Exhibit 1. WRITEUPS that provide information about the September 14 change in the WRITEUP utility.

More information about this change is available from the writeups listed in Exhibit 1.

If you have questions about this change, call Marisa Riviere, 376-9718.

(John Stueve)



—Graphic by the UCC Graphics Group

Computer Store

NEW HOURS

On July 18, the Computer Store extended its daytime hours. The Store is now open Monday through Thursday, 9 a.m. to 4:30 p.m. It closes at 3 p.m. on Fridays. The Store will be open from 5 to 7 p.m. Monday through Thursday during the first week of each quarter only. Evening hours have been discontinued for the rest of each quarter.

(Marvin Mohr)

HALF PRICE SALE

We have a good supply of reprints of the Berkeley 7th edition UNIX *Programmer's Manual*, in four volumes that we will sell for half price. (See Exhibit 1)

Take advantage of these terrific prices while the supply lasts.

Commands	was $7.50	now $3.75
Getting Started	was $7.15	now $3.60
Programming	was $9.25	now $4.65
Text Processing	was $5.00	now $2.50

Exhibit 1. Regular and sale prices for UCC reprints of the *UNIX Programmer's Manual*.

PHONE NUMBERS

Access:
- CYBER(CA)—10, 30 cps 376-5730
- —120 cps 376-5706
- MERITSS(ME)—10 cps 376-7710
- —30 cps 376-7730
- —120 cps 376-7120
- VAX/VMS(VA)—(autobaud) 376-9770

Budgets 373-2521
Computer-Aided Instruction 376-2975
Computer Hours (recorded message) 373-4927
Computer Store 373-4877
Consulting
- HELP-line 376-5592
  9 a.m.-5 p.m., Monday-Friday
- Business Data Products 376-1761
  1-3 p.m., Monday-Friday
- Statistics Packages 376-5062
  1-2 p.m., Monday-Friday
- Data Bases 376-1761
  1-3 p.m., Monday-Friday
- Microcomputers 376-4276
  10-12 a.m. and 2-4 p.m., Monday-Friday
- Non-Traditional Computing 376-2944
  1-3 p.m., Monday-Friday
- TELL-A-GRAF/DISSPLA 376-2663
  1-3 p.m. Tuesday, Thursday
- Text Processing 376-2944

Contract Programming 376-1764
Data Base Applications 376-1764
EDUNET Liaison 373-7745
Engineering Services 376-1023, 376-8153
Equipment Purchase/Information 376-8153
Experimental Engineering I/O 373-4596
Graphics Software 376-5592
HELP-line 376-5592
9 a.m.-5 p.m., Monday-Friday
HOURS-line (recorded message) 373-4927
Information, Experimental Engineering 373-4360
Information, Lauderdale 373-4912
Instructional Labs 376-2703
Instructional Services 373-7745
Lauderdale Computer Room 373-4940
Lauderdale Services 373-4995
Lauderdale Services Manager 373-7538
Lauderdale Users' Room 373-4921
MECC Liaison 373-7745
Newsletter Subscription 373-4912
Permanent File Restoration 376-5605
Professional Services Division (PSD) 376-1764
Project Assistance 376-1764
Reference Room 373-7744
Remote Batch (RJE) Services 376-2703
Short Courses 376-8806
Shuttle Bus Service 376-3068
System Status (recorded message) 373-4927
Tape Librarian: see Lauderdale Services
Text Processing Services 376-2943
User Accounts 373-4548

OPERATING HOURS

	CYBER CA/CB	Low rate	CRAY (CR)	MERITSS (ME)	VAX (VA)
M-F	7 a.m. - 4 a.m.	8 p.m. - 4 a.m.	7 a.m. - midnight	7:45 a.m. - 1:30 a.m.	8 a.m. - 6 a.m.
Sat	4 a.m. - 5:15 p.m.	4 a.m. - 5:15 p.m.	7 a.m. - 5 p.m.	7:45 a.m. - 1:30 a.m.	24 hours
Sun	4 p.m. - 1 a.m.	4 p.m. - 1 a.m.	4 p.m. - midnight	4 p.m. - midnight	24 hours

PUBLIC LABS—TWIN CITIES CAMPUS

Location	Batch	Interactive	Micro
East Bank			
Arch 160		X	X
CentH		X	
ComH		X	
DiehlH 270, 207		X	
EltH 121, 125		X	
EltH N640	X		
FolH 14, 14a	X	X*	X
LindH 26	X	X	
MechE 308		X	
Physics 69		*	
SanfH		X	
TerrH		X	
VincH 4		X	
WaLib 204		X	
West Bank			
BlegH 25		*	
BlegH 90	X		
BlegH 93T			X
BlegH 140		X	
MdbH		X	
OMWL 2		X	
St. Paul			
BaH		X	
ClaOff 125	X	X	

* Research cluster; access to Cyber 730 and VAX/VMS
X in interactive column indicates access to MERITSS

Contents

Director: Peter C. Patton
Editor: Christine Mack Gordon

The *UCC Newsletter* is published monthly by the University Computer Center. Deadline for articles is the 10th of the month preceding publication; deadline for short announcements is the 15th. The *Newsletter* is edited and coded for typesetting at the Computer Center, then typeset on a Linotron 202 and printed at the University of Minnesota's Printing and Graphics Arts Department.

Comments, suggestions, articles, and announcements should be directed to the editor, 227 Experimental Engineering, (612) 376-1491.

University Computer Center Newsletter

User Services
227 Experimental Engineering
University of Minnesota
208 Union Street SE
Minneapolis, Minnesota 55455

Nonprofit Org.
U.S. Postage
PAID
Minneapolis, Mn.
Permit No. 155