The ACSS
Newsletter

University of Minnesota
Twin Cities
June 1987

Text Processing

TEX Technical Typesetting Program on the UX

Elaine Collins
UX and VX Mail: ENC

We have installed the **TEX** technical typesetting program on our UX (the ENCORE Multimax) computer. The program was developed by Professor Donald Knuth of Stanford University and he has documented it in *The TEXbook* (Addison-Wesley, 1986). LaTEX, a package of TEX formatting macros based on structured document types and standard formatting environments within documents, and BibTEX, TEX macros for bibliography data bases and citations, are also available. LaTEX was developed by Leslie Lamport and documented in *LaTEX: A Document Preparation System* (Addison-Wesley, 1986). Both books are on reserve in the ACSS Computing Information Center, 128A Lind Hall. (The LaTEX book also explains how to use BibTEX.)

The output device for TEX on UX is an Apple LaserWriter, a 300 dot/inch device. The TEX device driver for this printer is called DVI2PS. It translates the device independent (DVI) file created by TEX or LaTEX into a Postscript file which can be printed using the UNIX `lpr` command.

Use the commands **`tex`** ***`filename`***, **`latex`** ***`filename`***, or **`bibtex`** ***`filename`*** at the UNIX prompt to run one of these programs. TEX and LaTEX expect input files with file type **.tex**, which you usually create separately using a text editor such as **vi**. If you are using BibTEX, you first run LaTEX, then BibTEX, and then LaTEX twice more as explained in the manual. TEX and LaTEX will both produce **.dvi** files, which contain the formatted text, and **.log** files, which contain run-time messages from the programs.

continued on page 142

In This Issue . . .

We announce several **new software packages**: TEX on the UX and SPSS-Tables and math and engineering packages on the VX. We're also **upgrading the NOS operating system** on the CYBERs this month.

On the VX, we offer **two new writeups**: one helps administrators of instructional accounts, the other explains VMS protection schemes. This issue contains summaries of both.

This issue also contains an important announcement and article for **punch card users**, information on obtaining **Macintosh instructional software**, and our **summer short course schedule**.

Changes in statistical consulting hours. See page 158.

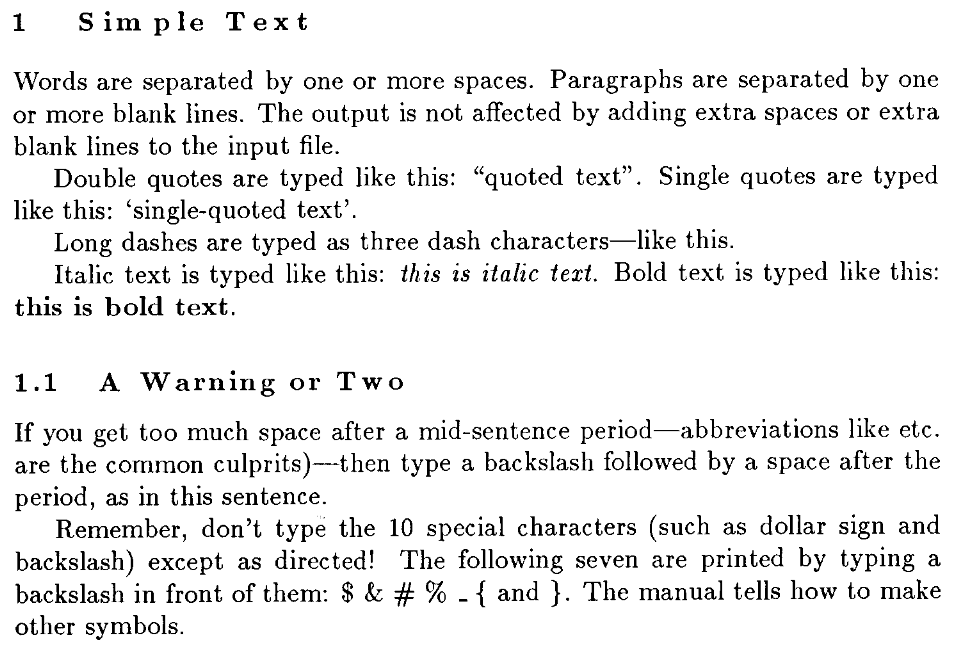
continued from page 141

Use the command **dvi2ps** ***filename >newfilename*** to create a file that can be printed on a laser printer. Then the command **lpr -Plind128b -C** ***site.bin newfilename*** will send the file to be printed and delivered to the designated site and bin.

The following LaTEX sample file–

```
\documentstyle{article}
\begin{document}

\section{Simple Text}

Words are separated by one or more spaces.  Paragraphs
are separated by one or more blank lines.  The output is
not affected by adding extra spaces or extra blank lines
to the input file.

Double quotes are typed like this: ``quoted text''.
Single quotes are typed like this: `single-quoted text'.

Long dashes are typed as three dash characters---like
this.

Italic text is typed like this: {\em this is italic
text}.
Bold text is typed like this: {\bf this is bold text}.

\subsection{A Warning or Two}

If you get too much space after a mid-sentence period---
abbreviations like etc.\ are the common culprits)---then
type a backslash followed by a space after the period,
as in this sentence.

Remember, don't type the 10 special characters (such as
dollar sign and backslash) except as directed!  The
following seven are     printed by typing a backslash in
front of them:  \$  \&  \#  \%  \_  \{  and  \}.  The
manual tells how to make other symbols.

\end{document}
```

–creates and prints this output–



1 Simple Text

Words are separated by one or more spaces. Paragraphs are separated by one or more blank lines. The output is not affected by adding extra spaces or extra blank lines to the input file.

Double quotes are typed like this: “quoted text”. Single quotes are typed like this: ‘single-quoted text’.

Long dashes are typed as three dash characters—like this.

Italic text is typed like this: *this is italic text.* Bold text is typed like this: **this is bold text.**

1.1 A Warning or Two

If you get too much space after a mid-sentence period—abbreviations like etc. are the common culprits)—then type a backslash followed by a space after the period, as in this sentence.

Remember, don't type the 10 special characters (such as dollar sign and backslash) except as directed! The following seven are printed by typing a backslash in front of them: $ & # % _ { and }. The manual tells how to make other symbols.

–from this terminal session:

```
> latex sample
> dvi2ps sample >temp
> lpr -Plindl28b -Cea.425 temp
```

Questions about TEX, LaTEX, and DVI2PS should be directed to Elaine Collins, 625-1391.

Liberal Arts Computing

Macintosh Instructional Software: The Academic Courseware Exchange

Ron Zacharski
BITNET: RAZ@UMNACVX
UX and VX Mail: RAZ

If you are interested in using the Macintosh for instructional use but lack the resources to develop your own software, you may be interested in Kinko's Academic Courseware Exchange (ACE). Kinko's and the Apple University Consortium have established this exchange to promote the sharing of Macintosh courseware developed at colleges and universities throughout the country.

continued on page 144

continued from page 143

Around 70 programs are currently available from ACE for a broad range of fields including history, English composition, geography, physics, economics, health science, and psychology. The cost of the courseware is comparable to that of textbooks (generally $10-20 a program).

Short descriptions of several programs (excerpts from the ACE catalog) follow:

> **The Would-Be Gentleman**, written at Stanford, models the economics and social life of a French bourgeois during the life and reign of Louis XIV. The student experiences the world of 17th-century France by managing income and properties, planning marriages and estates, and seeking influence through official duties and alliances with powerful figures.
>
> **Turing's World**, developed by Jon Barwise and John Etchemendy at Stanford, is a self-contained introduction to Turing machines, one of the fundamental notions of logic and computer science. It allows the student to build, debug, and run sophisticated Turing machines.
>
> **MindLab**, developed at Dartmouth College, is a system for creating, editing, and running simple psychology experiments in perception and cognition. Its strength is its ability to present pictorial stimuli generated in graphics programs such as MacPaint and MacDraw or digitized from photographs.

As the above examples illustrate, the programs are not merely drill-and-practice exercises; they are innovative programs, often simulations, that require the student to participate actively in the learning process.

For a catalog or for more information, contact

Kinko's Academic Courseware Exchange
4141 State St.
Santa Barbara, CA 93110
(800) 235-6640

A catalog is also available for reference in the Computing Information Center in 128A Lind Hall.

<u>*Consulting Corner*</u>

Convert Your Punch Cards Now

Pete Oberg
BITNET:PJO@UMNACVX
Mail CA,MD,VX,and UX: PJO

You're still using cards. They're cheap, eternal, and they stay in your office. The computing world marches on, but you are skeptical about turning over your files to a disk drive far away. You have concerns for long term storage of data. This article is for you.

Punched cards will go away–not immediately, not next month, but they will finally disappear. ACSS has reduced the number of keypunches available at its various sites and no longer provides punch card output. The number of keypunches and card readers will be further reduced during the coming year. By June 30, the only key punch and card reader provided by ACSS will be at our central site at Lauderdale.

Long-Term Storage of Card Decks

For long-term storage of card decks for use in 1990 and beyond, ACSS recommends that you store them on magnetic tape using the **TBLOCK/UNBLOCK** utility on the CYBER computer. **TBLOCK/UNBLOCK** is a program written and maintained by ACSS that follows an ANSI standard format of storing information on to tape. Then at some future date, TBLOCK/UNBLOCK at ACSS or a similar utility at a different computing site can access the information on tape. To work with **TBLOCK/UNBLOCK** on the CYBER, you will need to consult WRITEUP,TBLOCK and *The Guide to Magnetic Tape Usage*, available on-line as WRITEUP,TAPEUSE.

Frequently Used Card Decks

Using the CYBER: The most effective way to store card decks that are occasionally used on the CYBER is on magnetic tapes using the **ARCHIVE** utility. **ARCHIVE** is a program written and maintained by ACSS that makes tapes easier for users to manipulate. If you use the CYBER exclusively, **ARCHIVE** is the program for you. Archive allows you to access your decks by name from tape to disk files for your work. After you finish, remove the files from disk so you do not incur disk storage charges. To work with ARCHIVE and tapes on the CYBER , you will need to consult WRITEUP,ARCHIVE and the *Guide to Magnetic Tape Usage*, available online as WRITEUP,TAPEUSE.

Using the VAX or ENCORE in the Future: We have no card reader for the VAX and ENCORE computers. To place your card deck files on the VAX you will need to transfer your files from cards onto the CYBER as disk files, then send the files to the VAX using the **TOVAX** utility. To work with **TOVAX** , you will need to consult the CA on-line document, WRITEUP,FILETRA.

On the VAX, the most effective way to store files on magnetic tape is to use the **BACKUP** utility, written and maintained by Digital Equipment Corporation. To work with **BACKUP** and tapes on the VAX, you need to consult the file **ACSS$WRITEUP:VMSTAPES.** To read this document at your terminal, type:

```
        $ type ACSS$WRITEUP:VMSTAPES
```

If you are planning to use the ENCORE computer, you also move your card decks to the CYBER as disk files. (To maintain many disk files that are not used frequently, we recommend that you store them on magnetic tape on the CYBER or VAX.) When you are ready to work with these files, each can be accessed and transported to the ENCORE using the File Transfer Package (FTP) utility. Next month, the Consulting Corner article will explain how to use the FTP utility to transfer files to an UNIX environment.

You can also call our HELP-Line, 626-5592, to talk to a consultant about any questions you may have with converting your punch cards.

Punch Card Users: Campus Keypunches, Card Readers Departing

Jerry Larson
ACSS Lab Manager

As we've previously announced, ACSS will remove the card readers and keypunches from 128B Lind Hall and 125E Classroom Office Building on June 30, 1987. After that date, ACSS's only card reader and keypunch will be at our central site, the Lauderdale Computer Facility.

During the last year, we've encouraged card users to convert their card decks to disk or tape files, and we've offered classes and consulting to card users to help them accomplish this. If card users have any questions or problems concerning the removal of our campus card facilities, they should contact me at 14 Folwell, 625-7850.

System Note: Change in SRU Accounting

Dave Bianchi
BITNET: DJB@UMNACVX
CA, MD, UX, and VX Mail: djb

An error in SRU accounting was introduced in the change to NOS 2.5 on March 22, 1987, and was corrected on April 29, 1987 and the April ACCSTAT totals were corrected. April billings were accurate and also reflected the correction for the last nine days of March. For CYBER CA (855) the correction adds 1.3 times CP seconds to the SRU total. For the CYBER MD (830) the correction subtracts 0.5 times CP seconds from the SRU total. These corrections are approximate since they do not use memory occupancy.

ACSS apologizes for any inconvenience this error may have caused. Please contact ACSS Accounting at 625-1511 if the change causes unanticipated grant overruns or other problems.

CYBER Upgrade to NOS 2.5.2

Dave Bianchi

On June 14, we plan to upgrade our CYBER CA and MD operating systems to NOS 2.5.2-678. Users will notice relatively few changes in the new operating system. For more information on these changes, see the on-line document **WRITEUP,NOS678** on the CYBERs.

VX FORTRAN Libraries: PCGPAK Sparse Matrix Library

Michael J. Frisch
BITNET: MJFRISCH@UMNACVX

We recently installed the PCGPAK user library on the VAX 8600 (VX) under the VMS operating system. PCGPAK solves DOUBLE PRECISION sparse systems of nonsymmetric linear equations by iterative methods. The only memory PCGPAK needs is for the nonzeros, some pointers for them, and a relatively small amount of workspace. No external files are used.

PCGPAK has three iterative methods that can be used by themselves, or in conjunction with three preconditioning methods. (The PCG in the name PCGPAK stands for "preconditioned conjugate gradient," one of the combination of methods provided.) For block two-cyclic matrix problems, a preprocessing routine can perform partial elimination of some of the unknowns before it solves the rest of the problem.

Accessing PCGPAK

PCGPAK is only available on the VX machine. It is accessed by the command

```
$ LIBS PCGPAK
```

and FORTRAN programs that call routines in the library must use the **/G_FLOATING** compiler option and must set up arrays to be in DOUBLE PRECISION. Our contract with the vendor, Scientific Computing Associates, limits use of PCGPAK solely for academic and research purposes.

Advantages of PCGPAK

For certain types of problems, ACSS regards PCGPAK as a better method for sparse nonsymmetric linear equation problems than the Yale Sparse Matrix Package, YSMPLIB. (PCGPAK was designed by the authors of YSMPLIB.) PCGPAK is particularly useful with equations arising from solving partial differential equations where the system is diagonally dominant—that is, where diagonal elements are larger in absolute value than off-diagonal elements. The package also does well on problems with some sort of regular structure as opposed to just random or near-random locations of nonzeros.

For problems in its area, PCGPAK almost always takes much less storage and usually takes much less computer time than YSMPLIB. PCGPAK uses iterative methods rather than the direct nonpivoting methods used by YSMPLIB. However, the documentation for PCGPAK does not give many hints as to which of the iterative methods to use or when to use preconditioning.

continued on page 148

continued from page 147

Documentation

Documentation for the library is available by the command

```
$ TYPE ACSS$WRITEUP:PCGPAK
```

and from an 18-page *PCGPAK User's Guide* that is on reserve in the ACSS Computing Information Center (CIC), 128A Lind Hall, (612) 625-7397. For the price of reproduction, CIC staff can make you a photocopy.

Questions on PCGPAK can be directed to Michael Frisch at (612) 625-5830.

SPICE Circuit Analysis Package on the VX

Michael J. Frisch

We recently installed the SPICE version 2G.6 package on the VAX 8600 (VX) under the VMS operating system. SPICE is a general-purpose circuit simulation program for nonlinear DC, nonlinear transient, and linear AC analyses. Circuits may contain resistors, capacitors, inductors, mutual inductors, independent voltage and current sources, four types of dependent sources, transmission lines, and the four most common semiconductor devices: diodes, BJT's, DJFET's, and MOSFET's.

The command to run SPICE under VMS is:

```
$ SPICE inputfile outputfile
```

where

the file ***inputfile*** contains the input data for SPICE. If the file type is not given, then the file type is assumed to be **.DAT**.

the file ***outputfile*** contains the results from SPICE. If the file type is not given then the file type is assumed to be **.OUT** and if the file name is not present, then ***inputfile*.OUT** is used.

As an example of the above,

```
$ SPICE SAM
```

is the same as

```
$ SPICE SAM.DAT SAM.OUT
```

Documentation for SPICE can be found on the CYBER CA by typing

```
WRITEUP,SPICE
```

which is a 75-page user's manual. On the VX, the information in this article is reproduced in

```
$ MOREHELP APPLICATIONS SPICE
```

New Version of YSMPLIB for CA, MD, and VX

Michael J. Frisch

On June 17, we will install a new version of the Yale Sparse Matrix Package (YSMPLIB) on the CYBER CA and MD machines and on the VX machine. This version corrects the problem noted in the April *Newsletter* about the routine CDRV that solves nonsymmetric sparse matrix problems. The documentation says that the arrays B and Z can be the same and the new version corrects the code to make this be true.

Also, this new version has changed some of the routines that solve symmetric matrix problems. The size of the scratch arrays may have to be made larger compared to the previous version. Furthermore, on the VX, for symmetric and nonsymmetric problems in routines SDRV, NDRV, TDRV, and CDRV, where scratch arrays ISP and RSP are used, the arrays should be EQUIVALENCE'd to each other, RSP should be typed DOUBLE PRECISION, and ISP should be dimensioned twice the size of RSP.

A one-page update sheet is available in the ACSS Computing Information Center (CIC), 128A Lind Hall, (612) 625-7397.

Subscribe By Electronic Mail

You can now subscribe to the *ACSS Newsletter* using electronic mail. On the CYBER CA send mail to YZE6075@UMNACCA. On the VAX VX send mail to MAD@UMNACVX. When you request your subscription, please include the following information:

your name
department name
departmental room number, building
phone number

Computing Costs: A University Model

Lawrence Liddiard
BITNET: LIDDIARD@UMNACVX
UX Mail: lal

In the past few months University departments have been preparing Strategy for Focus documents that outline their programs and services for the next three to five years. ACSS's own response to Strategy for Focus used the following words to describe the present state of computing:

> New technology is changing the way computing is used at the University. Manufacturers are placing more and more capability on a single computer chip at the same or reduced price. As a result, increasingly powerful workstations and microcomputers are bringing low-cost computing to the user's desk.
>
> Local area networks provide interconnections for small work groups or departments. This allows electronic communications among individuals and facilitates sharing of expensive equipment and software. The end result is more control of the computing environment at the local level.
>
> Workstations and microcomputers, even if abundant, do not obviate the continuing need for a variety of centrally provided sharable computing resources, such as supercomputers, parallel processing computers, large data storage devices, high-speed printing devices, expensive software, and connections to national communications networks. Even general purpose computing facilities similar to those provided now will be needed by those persons, groups, or departments who cannot justify a workstation of adequate speed, capacity and software capability to satisfy all their computing needs.
>
> Thus, changing technology is creating a new model for computing at Minnesota: workstations and microcomputers, with differing levels of capability, linked to each other and to other devices and with gateways to a variety of shared services.

Implementation of the Strategy

Our goal is to provide the best computing services at the lowest possible cost. One way to approach this problem is to examine the approximate cost of current computing systems. Table 1 shows five typical cases among processors with two megabytes of memory per MIP (millions of instructions per second) of processing power.

Type	Workstation I	Workstation II	Multiprocessor I	Multiprocessor II	Supercomputer
Examples	Macintosh Plus IBM PS-2/30	SUN, Apollo VAXStation	ENCORE Sequent	IBM 3090	CRAY XMP
Processors	1	1	16	4	2
MIPS/Processor	1/2	2	2	13	60
Total MIPS	1/2	2	32	50	120
Memory in Megabytes	1	4	64	100	240
Average Cost	$1.5K	$8K	$480K	$7,500K	$7,200K
Cost /MIP	$2K-$4K	$3K-$5K	$12K-$18K	$120K-$180K	$50K-$70K

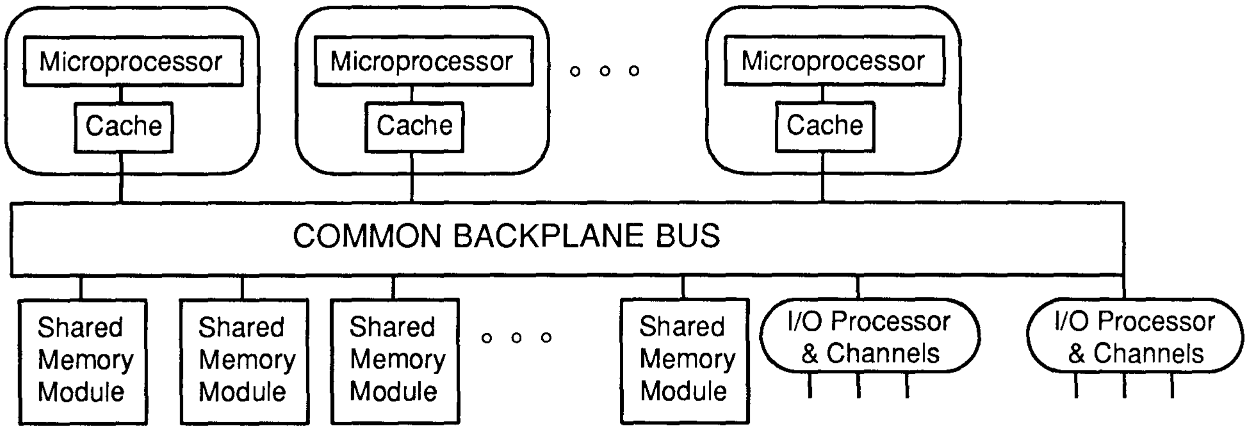
Table 1: Typical Processor Cost per MIP

Trends in Design and Cost

In the past few years the trend in computer design has been to achieve systems that lower the cost of MIPS by providing N MIPS with K commodity processors of N/K MIPS each sharing a common memory. (I discuss other aspects of this architecture in my article, "MIPS/Dollar: The VLSI Revolution" in the September 1986 issue of this *Newsletter.*)

The following diagram illustrates the typical parts and connections of these systems.

SHARED MEMORY MIMD SYSTEM



The advantages of this design are:

- a cost-effective implemention using the fastest memory and microprocessor chips.
- the backplane bus providing a standard interconnect for individual boards.
- the incremental expandability of required memory, microprocessor, and I/O processor resources.
- a cache that reduces average memory access time and required backplane bandwidth.
- standard software existing for off-the-shelf microprocessors.

The ENCORE Multimax systems installed last year at ACSS and Duluth Computing Services are examples of this architecture. Since universities require the maximum computing for the dollar, we expect to see these types of computers become the dominant central processors on campuses.

continued on page 152

continued from page 151

Some Poor Bargains

The IBM 3090 series has a shared memory design with up to 6 processors; but since they are not built with commodity microprocessor chips, the resulting MIPS are priced ten times higher. When ACSS acquired the VAX 8600 last year, I was sure that the Digital Equipment Corporation (manufacturer of the VAX) would next deliver VAX computers with 8 to 16 MicroVAX II processor chips. Instead DEC introduced VAX 82XX, 83XX, and 85XX systems priced at $40,000 per MIP in the lower ranges and $80,000 in the upper ranges, with at most two processors to a system.

These VAX MIPS invert Grosch's Law–"Twice the cost provides four times the computer power"–to "Four times the cost gives twice the power," the equivalent of asking for four five-dollar bills in change for a ten.

Supercomputers do not solve the problem entirely. Although supercomputers can be cost-effective for massive vector problems, for typical scalar problems they have costs 3 to 5 times higher than systems with commodity processors and shared memory.

Our Recommendations

- By 1990 every University faculty member should have a graphics-based workstation connected to a local area network with file and laser printer servers. This configuration would support electronic mail, computer conferencing, and departmental and University bulletin service from the University central network. In addition, faculty could use these workstations for word processing, text formatting, and small application packages (programming languages, data base, spread sheet, small expert systems, business charts and graphics).

- Every student, graduate and undergraduate, should have access to a similar graphics-based workstation. There should be one workstation for every two graduate students and an increased number of micro and terminal labs for undergraduates to match their computing-based assignments.

- The initial recommendations of the All-University Committee for Network and Communications Planning should be implemented; especially those on charging and support of standards. The committee supports a reliable backbone University network providing the required interconnection and ubiquitous service needed for faculty and student computing in the 1990's.

- In this University model, central funding should support a minimum environment for every faculty member and student. This will include the workstation, networking, and centralized services. This may involve contracting for deep discounts on workstations, subsidizing lower telecommunications rates, and increasing grants for central services.

SPSSX-TABLES on the VX

Sharon Krmpotich
VX Mail: sak

The TABLES procedure in SPSSX release 2.1 is a comprehensive program for producing simple or complex tabulations in camera-ready form, suitable for immediate presentation. TABLES simplifies the production of complex stub and banner cross tabulations, frequency counts, statistical breakdowns, and expanded statistical tabulations. It gives you complete control over the structure of the table, its rows, columns, and layers. TABLES provides flexible percentaging, allowing you to define the base to get any combination of table, column, row, and subtable percentages.

Another unique feature in TABLES is that it retains variable labels up to 120 characters long and value labels up to 60 characters. There is no limit on the number of unique values for any variable, nor on the number of cases. For maximum efficiency, TABLES reads the data only once for all the tables that you are creating from the same file.

SPSSX has chosen default settings for formatting, statistics, and labeling; these settings allow you to produce high quality tables with little effort. TABLES derives most default print formats for statistics from the print formats of the variables, but allows you to specify what you want within TABLES from a list of format elements including commas, dollar signs, percent signs, and so forth.

The procedure uses default line-drawing, spacing, column widths, and special characters, but allows you to override defaults with a wide range of choices. Using the combined power of the SPSSX language and the TABLES procedure, you can easily assume full control over the appearance of your tables.

Who Should Use TABLES?

The procedure has something for any user of statistics. Market researchers often need to condense a great deal of information in a single table, which means using multiple variables in the same display. Survey researchers need missing-value handling and long labels. Most professional publications have standards for the presentation of tabular data in articles. TABLES provides for all of these requirements, and offers a special tabular format that conforms to the dissertation style requirements of the University of Chicago.

A Sample TABLES Job

Here is an example of a simple TABLES program and its output. Our SPSSX command file is named **TABLESEXAMPLE.SPS**.

continued on page 154

continued from page 153

To execute this job on the VX, type

```
SPSSX/OUTPUT=outfile  TABLESEXAMPLE.SPS
```

Replace ***outfile*** with the name of the file you want to receive the output from the job.

The file named **DRINK.FIL**—named in the following job—is an SPSSX system file that contains our data.

The contents of the SPSSX command file TABLESEXAMPLE.SPS are:

```
FILE HANDLE DRINK/ NAME='DRINK.FIL'
GET FILE = DRINK/ KEEP = AGE YEARSED STAR BUY SEX
VARIABLE LABELS AGE 'Age' YEARSED 'Education'
TABLES OBSERVATION = YEARSED AGE/
  FORMAT = CWIDTH(20,5) MARGINS(1,70) OFFSET(1) BOX/
  TABLE = STAR + BUY BY SEX > (AGE + YEARSED)/
  STATISTICS = VALIDN('') MEAN((F4.1)'')/
  TTITLE 'Count and Mean for Age and Years of Education'
FINISH
```

Let's briefly examine some of the contents of this command file:

The `VARIABLE LABELS` command supplies `AGE` with the label `'Age'`, and `YEARSED` with the label `'Education'`.

The `OBSERVATION` subcommand on the `TABLES` command declares `YEARSED` and `AGE` as observation variables. Otherwise, they would be considered category variables like `SEX` and would create one column for each unique age and years of education value. Category variables are used to classify the data into groups, while observation variables are used for summary statistics within the groups defined by category variables.

The `FORMAT` subcommand sets up column widths, margins, a cell offset (from the right cell margin), and asks for the `BOX` format to include additional boxing lines within the table.

The `TABLE` subcommand concatenates variables `STAR` and `BUY` in the stub (left-side section) as row variables, and nests the observation variables (YEARSED and AGE) within `SEX` (the column variable) in the banner. The parentheses around `AGE` and `YEARSED` tell `TABLES` to nest both variables.

The `STATISTICS` subcommand asks for two statistics, percent of valid cases and the mean. The mean is assigned a format and both statistics are given null labels. Since the observation variables are in the banner, the statistics are presented in separate columns.

The `TTITLE` subcommand gives information about cell contents.

Output

After you've executed the job above, SPSSX-TABLES produces this table:

```
                Count and Mean for Age and Years of Education

+-------------------+-------------------------------------------------+
|                   |                 Sex of respondent               |
|                   +-----------------------+-------------------------+
|                   |         Female        |          Male           |
|                   +----------+------------+-----------+-------------+
|                   |   Age    | Education  |   Age     |  Education  |
+-------------------+----------+------------+-----------+-------------+
|Player admired most|          |            |           |             |
|Ivan Lendl         | 64  27.9 |  59  12.6  |  96  29.4 |  93  13.4   |
|John McEnroe       | 59  28.6 |  57  12.7  | 104  29.1 |  97  13.4   |
|Martina Navratilova|112  29.4 | 114  12.4  |  64  28.9 |  68  12.7   |
|Jimmy Connors      | 34  31.2 |  33  12.3  |  77  30.4 |  74  13.0   |
|Chris Evert Lloyd  | 55  29.9 |  55  12.3  |  24  28.6 |  24  12.6   |
|Yannick Noah       | 31  28.0 |  31  13.1  |  49  29.3 |  47  12.8   |
+-------------------+----------+------------+-----------+-------------+
|Buy endorsed drink |          |            |           |             |
|Would buy          |104  27.2 | 103  12.5  |  95  27.0 |  91  12.7   |
|Probably would buy | 98  29.8 |  95  12.4  |  82  26.9 |  79  12.7   |
|Neutral            | 96  29.6 |  94  12.5  | 120  30.4 | 119  13.1   |
|Probably not buy   | 44  30.8 |  44  12.5  |  89  32.1 |  87  13.7   |
|Not buy            |  7  31.6 |   6  14.0  |  19  32.7 |  19  13.5   |
+-------------------+----------+------------+-----------+-------------+
```

The information presented in our table could also have been produced by the SPSSX procedure BREAKDOWN, using the CROSSBREAK facility, but the output would not be as condensed or readable.

If you have questions concerning SPSSX-TABLES, call the Statistics HELP-Line at 626-1887, Monday through Friday, 1:00 to 3:00 p.m.

References

SPSS Tables Primer, SPSS, Inc.
SPSSX Tables User's Guide, SPSS, Inc.

Holiday Hours

Independence Day Hours

In observance of Independence Day, the following ACSS systems, CYBER 855 (CA), VAX 8600 (VX), ENCORE (UX), and CYBER 830 (MD), will run in unattended mode from 6 a.m. Saturday, July 4, to 6 a.m. Sunday, July 5. It is unlikely that any tape requests or printing will be processed during these hours. Normal operations on all systems will resume at 6 a.m. Sunday, July 5.

Classifieds

For Sale

A Plessey 600 series minicomputer (PDP-11/23+ with a 22 bit bus) with an expansion box (6 empty slots) and an attached Kennedy 9220 9-track magnetic tape drive. Peripheral equipment includes two Plessey PT-100B display monitors with Selanar graphics boards, an Anadex Silent Scribe dot matix printer, ADAC Corp. high level analog input data aquisition cards models 1023AD, 1622DMA, 1023EX, ADAC Corp. general purpose timer card model 1601GPT, and complete manuals for for RT-11 (v5) and RSX-11M (v4.1) systems. Price is negotiable as a package or individual items. A within university sale is preferred. Contact the Department of Fisheries and Wildlife, Jeff Stone at 624-3785 or Yosef Cohen at 624-2255.

ACADEMIC COMPUTING SERVICES & SYSTEMS
Summer Session Short Courses
1987

INTRODUCTORY COURSES

(Introductory Courses are FREE. To register call 625-7397)
(Classrooms listed *may* be for the first day of class only.)

Introduction to Computers	June 23-July 9	(TTh)	2:15-4 pm	SciCB 125
FSE (NOS Full-Screen Editor)	June 29-July 1	(MW)	2:15-4 pm	SciCB 125
Introduction to VAX/VMS Operating System	July 6-15	(MWF)	2:15-4 pm	SciCB 125
Introduction to UNIX Operating System	July 14-30	(TTh)	2:15-4 pm	SciCB 125
Electronic Mail (VAX Mail)	July 27-29	(MW)	2:15-4 pm	SciCB 125

ELECTIVE COURSES

FEES: 1) U Students, 2) U Staff/Faculty, 3) Non-University persons

Using SPSSX (Statistics)	July 13-17	(MWF)	3:15-5 pm	$10,$10,$20
Programming in Pascal	July 20-31	(MWF)	2:15-4 pm	$15,$15,$40
Using SAS (Statistics)	Aug 3-7	(MWF)	3:15-5 pm	$10,$10,$20
Programming in 'C' Language	Aug 3-17	(MW)	2:15-4 pm	$15,$15,$40
SIR (Data Base Mgmt System)	Aug 4-20	(TTh)	2:15-4 pm	$15,$15,$40

MICROCOMPUTER APPLICATIONS COURSES

FEES: 1) U Students, 2) U staff/Faculty, 3) Non-University persons
(IBM and Compatible courses are absolutely limited to 10 per class.)

Introduction to Micros: DOS	section 1	June 23-25	(TWTh)	10:00 am-noon	$40,$50,$80
	section 2	August 3-7	(MWF)	1:30-3:30 pm	$40,$50,$80

(*Introduction to Micros* or equivalent knowledge is required for IBM courses below.)

Managing Your Hard Disk		August 11-13	(TTh)	1:30-3:30 pm	$40,$50,$80
Introduction to WordPerfect	section 1	July 9	(Th)	9:30 am-noon	$25,$35,$60
	section 2	August 19	(W)	1:30-4 pm	$25,$35,$60

(These courses are OVERVIEWS ONLY for the Apple Macintosh. Limited to 12 people.)

Microsoft Word for the Macintosh	section 1	July 15	(W)	1:30-4 pm	$15,$25,$40
	section 2	August 25	(T)	1:30-4 pm	$15,$25,$40

ACSS SHORT COURSE INFORMATION

PREREQUISITES: Please note any prerequisites for the class you are interested in. (See the ACSS Short Course *Brief* or call the Computing Information Center at 625-7397.) Instructors will **not** be able to review any prerequisite requirements.

REGISTRATION: Registration is located at the Academic Computing Services and System's Computer Information Center, 128A Lind Hall. (Registration hours: 8:00 am to 4:00 pm, Monday through Friday). We accept mail registrations. Deadline for registering is 4:00 pm on the last working day *before* the class begins. Please call and give us your name if you plan to attend a free class, so we know how many to expect. For registration information call 625-7397.

FEES: Fees must accompany registration. Fees are listed in order for the following groups: 1) University students, 2) University staff and faculty, and 3) non-University persons. Course fees may be paid by cash, check, or a signed University journal voucher, brought or sent to 128A Lind. No refunds will be made after the class has begun.

NOTE: Friday, July 3, is a University holiday. No classes will be held.

Statistics Consulting

Pete Oberg

Starting Monday, June 15, the Statistics HELP-Line hours will be 1:00 to 3:00 p.m., Monday through Friday.

VX Writeup Helps Instructional Administrators

Marisa Riviere
VX Mail: Marisa

A new writeup on the VX includes information previously published in this *Newsletter* about ACSS facilities that help departmental and class administrators oversee instructional accounts on the VAX 8600 (the VX). It also includes information about new administrative software developed since the newsletter article was published.

The document is available in the file Instructional.Admin on the account ACSS$Writeup. To read this account at your terminal, use this command:

```
$ type acss$writeup:instructional.admin
```

The file contains four sections:

Section 1 explains how to use general administrative software for changing passwords, controlling the login environment, communicating with student users, and monitoring file space and system usage.

Session 2 describes the standard validation for instructional users.

Session 3 describes in detail the function and use of two new ACSS utilities, DQMU and TLMU, utilities that monitor disk space and limit CPU and connect time.

Session 4 documents the MTOOLS procedure, which creates procedure files for implementing validation changes and other changes for large groups of users.

The ACSS Instructional Coordinator will work with instructional administrators and will consider suggestions for additions and modifications to the administrative software. If a department requests it, ACSS can schedule training sessions for instructors and administrators, to further explain the software described in this writeup. For more information, call the ACSS HELP-Line, 626-5592.

Protections on the VX: A Writeup and Summary

Jill McAllister
BITNET: JLM@UMNACVX

Users of our VAX/VMS system, the VX, frequently ask questions about file and directory protections. We have prepared a new VX WRITEUP on protection mechanisms on the VAX/VMS system. This WRITEUP explains in detail how to use the various protection mechanisms provided by the VMS operating system, and the implications of each.

The WRITEUP is available under the name **ACSS$WRITEUP_:PROTECTION.LIS**. To read this file at your terminal, use the VMS command **TYPE**, like so:

```
$ TYPE ACSS$WRITEUP_:PROTECTION.LIS
```

You can use other VMS commands as well to access this file, like **COPY**, **EDIT/READ**, or **PRINT**, as you prefer.

VMS protections can be useful to every VX user, so I've summarized the WRITEUP below. Inexperienced VX users should see the complete WRITEUP, which contains more definitions, explanations, and examples.

A Summary

There are two main protection mechanisms in the VMS operating system. The first is standard *UIC-based protection*. (UIC is for "user identification code.") This protection controls access according to the four user categories of:

SYSTEM
OWNER
GROUP
WORLD

The second protection mechanism is Access Control Lists, or ACLs. ACLs allow complex patterns of file sharing, by permitting files to individual users with individually specified types of accessibility.

Using standard UIC-based protection, there are four types of access to an *object*. (An object is any entity such as a file, a directory, a device, a volume, or a mailbox.) These access types are READ, WRITE, EXECUTE, and DELETE. EXECUTE access performs different functions, depending on whether the object accessed is a general file, a directory file, or a volume.

When anyone requests access to an object that they do not own, evaluation occurs in a specific order. ACLs, if any, are always evaluated first. Next, when existing ACLs fail to specifically grant access, the system checks the UIC-based protection. When an ACL specifically denies access to a user, that user is denied access even though the UIC-based protection gives him or her access. In other words, ACLs take precedence over the UIC-based protection scheme.

To see the protection on your files, use the command:

```
$ DIR/PROT
```

To see the protection on your root directory, you have to look back a level by using the `[-]` which backs up one directory. If, for example, your login name is **ABC**, you can see the protection of your root directory with this command:

```
$ DIR/PROT [-]ABC.DIR
```

Within parentheses, four fields will be listed, separated by commas. (Some may be blank fields, which causes commas to be right next to each other. Blank fields mean that that particular group has no access.) The four fields are not labelled, but are always listed in the same order: SYSTEM, OWNER, GROUP, WORLD.

To change this protection, use the command:

```
$ SET PROTECTION=(SYSTEM:RWED,OWNER:RWED,GROUP:R,WORLD:R)
  filename.type
```

Of course, you must use the protection you actually want to give. There are several different ways to set up ACLs. One way is to use the **SET ACL** command. This command allows you to modify or create an Access Control List for an object.

The **SET ACL** command is followed by the object name. The exact format is

```
$ SET ACL/ACL=(IDENTIFIER=xyz,ACCESS=READ) filename.type
```

where ***xyz*** is a user name or a group identifier.

The various access types that you can allow are READ, WRITE, EXECUTE, DELETE, and CONTROL. CONTROL gives owner-like access.

There are also several different commands which can be used to display the ACLs associated with files on your directory. One is **SHOW ACL**. This command must be followed by an object name.

```
$ SHOW ACL filename.type
```

Another command is **DIRECTORY/SECURITY**. This command lists each file on your directory, plus its UIC-based protections and any ACLs that have been set up for that file.

If you have any questions or problems using VAX/VMS protections, please call the HELP-Line at 626-5592, Monday through Friday, between the hours of 8:00 a.m. and 5:00 p.m.

The Help Page

General Consulting

Walk-in: 128C Lind Hall, East Bank; 10 am to 4 pm weekdays except 10 am to 11 am Wednesday

140 Blegen Hall, West Bank; 2:15 to 4:45 Monday, 12:15 to 2:45 Tuesday, 11:15 to 2:45 Wednesday, 1:15 to 4:15 Thursday, 2:30 to 4:30 Friday

HELP-Line: 626-5592, 8 am to 5 pm weekdays

Specialized HELP-Lines ☎

HELP-Line	Phone Number	Hours
Artificial Intelligence	625-8332	3 to 4 pm weekdays.
Data Bases	626-1887	10 to 11 am weekdays.
Microcomputers	626-4276	9 am to noon & 1:30 to 4 pm, Monday, Tuesday, Friday; 9 am to 4 pm, Wednesday and Thursday.
Statistics	626-1893	1 to 3 pm weekdays.
Text Analysis	625-8332	3 to 4 pm weekdays.
Text Processing	625-1391	9:30 to 11:30 am, Monday through Thursday.

Consulting by Mail ✉

Consulting is now available via the mail facility on all ACSS systems (the CA, MD, VX, and UX). Send mail to user name CONSULT for questions after hours and for low-priority questions that are not critical to your immediate computing work. Replies will be sent to your account through the mail facility on your system.

Instructional Computing Consultant

Department instructors may call 626-1085 for assistance in choosing ACSS systems (CYBER/NOS, VAX/VMS, ENCORE/UNIX), software, and for answers to any other inquiries on using computers for instructional computing.

Computing Information Center

128A Lind Hall, 625-7397, YZE6075@UMNACCA or MAD@UMNACVX

Computing account and grant applications available for CYBER, ENCORE, and VAX computers.

Short course enrollment. Short course schedules and class descriptions available.

Assistance in ordering vendor documentation. Vendor documentation is not always available in the University bookstores and may be ordered directly from the company.

Complete documentation collection. Reference copies of vendor and all other documentation for ACSS software.

Free ACSS documentation available.

Computing Newsletters. Subscribe to the *ACSS Newsletter* or the *ACSS Microcomputer Newsletter.* Newsletters from other computing centers are also available for reference.

ACSS PHONE NUMBERS

Administrative Office: 626-1600 **HELP-Line: 626-5592**

Access:			
CYBER (CA)	626-1620	Lauderdale Services	626-1838
CYBER (MD)	626-1622	Magnetic Tape Librarian	626-1838
ENCORE (UX)	626-1681	Math and Statistics Packages	625-5830
VAX (VX)	626-1641	Micro Information	626-4276
RJE (2400 baud)	626-1656	Newsletter Subscription	625-7397
RJE (4800 baud)	626-1663	Permanent File Restoration	626-0595
Accounts:		Project Assist	626-1090
CYBER, ENCORE, VAX	625-1511	Public Labs (Managed by ACSS)	
Computer Hours (recorded message)	626-1819	14 Folwell Hall	625-4896
Computing Information Center	625-7397	306B Lind Hall	625-9032
Contract Services	625-2303	130 Physics	625-6820
East Bank I/O, 128B Lind Hall	625-5082	9 Walter Library	626-1899
Engineering Services	627-4177	Public Labs/Remote Batch (RJE) Mgr.	625-7850
Equipment Maintenance/Repair	627-4357	Publications Information	626-1093
Graphics Software	626-5592	Short Course Registration	625-7397
Information, Lauderdale	626-1600	Shuttle Bus Service	625-9525
Lauderdale Computer Room	626-0550	System Status (recorded message)	626-1819

SYSTEM OPERATING HOURS

	CYBER (CA), ENCORE (UX), and VAX (VX)	CA and VX Low Rate	CYBER (MD)
M-F	7 am - 4 am	8 pm - 4 am, 7 am- 8 am	7 am - 1 am
Sat	4 am - 9 pm	all operating hours	7 am - 10 pm
Sun	6 pm - 4 am	all operating hours	6 pm - 1 am

PUBLIC LABS-TWIN CITIES CAMPUS

Location	Batch	Interactive	Micro	Location	Batch	Interactive	Micro
East Bank				TerrH		X	
ApH 204			X	VinH 4		X	
Arch 148			X	VinH 203			X
CenH		X		WaLib 9		X	X
ComH		X					
DiehlH 207, 270		X		*West Bank*			
EddyH Annex 54			X	AndH 170	P		X
EltH 121, 124	P	X		BlegH 140	P	X	
FolH 14, 14a	P	X	X	MdbH		X	
FronH		X		OMWL 2	P	X	
LindH 26		X					
LindH 128B	X	X		*St. Paul*			
LindH 306B			X	BaH		X	
MechE 308		X		CentLib B50			X
MoosT 8-425			X	ClaOff 125 E	X	X	
Phys 130		X	X	ClaOff B22			X
PioH		X					
SanH		X					

P means Printer only.
For more information see WRITEUP,LABS.

Contents

Text Processing

Liberal Arts Computing

Consulting Corner

NOS News

Math and Engineering Packages

Computing Reflections

Statistics

Holiday Hours

Classifieds

Short Courses

User Services

VMS News

The ACSS Newsletter
June 1987
Volume 22, Number 6

Acting Director: *Michael M. Skow*
Editors: *Steven Brehe, Paula Goblirsch*

The *ACSS Newsletter* is published monthly by Academic Computing Services and Systems of the University of Minnesota, Twin Cities. Deadline for articles is the 10th of the month preceding publication; deadline for short announcements is the 15th. The *Newsletter* is produced with an Apple Macintosh Plus running Microsoft Word, Full Paint, MacDraw, and Aldus PageMaker software, with camera-ready copy produced on the Apple LaserWriter Plus.

Direct comments, suggestions, articles, and announcements to the editors at the address below, or call (612) 626-1828 or 626-1093. For a free subscription call (612) 625-7397, or send your name and address to the Computing Information Center, 128A Lind Hall. Electronic mail: YZE6075@UMNACCA or MAD@UMNACVX. On-campus address changes *must* include your department's name and your *departmental* address.

The University of Minnesota is committed to the policy that all persons shall have equal access to its programs, facilities, and employment without regard to race, religion, color, sex, national origin, handicap, age, veteran status, or sexual orientation.

The ACSS
Newsletter

**Academic
Computing
Services and
Systems**

Technical Publications
100 Lauderdale CF
University of Minnesota
2520 Broadway Drive
Lauderdale, Minnesota 55113

Deliver to current occupant.