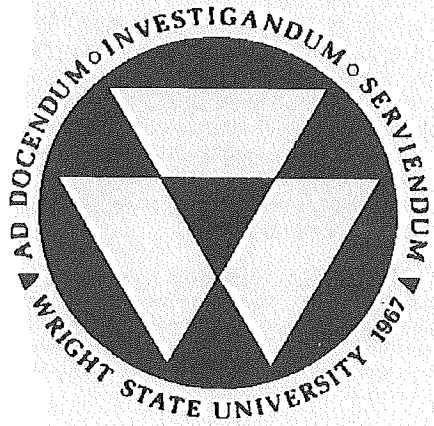
TEXTECHNOLOGY



Volume 1 Number 2
March 1991

In this issue . . .

EDITOR'S CHOICE

Jim Schwartz

Many readers have requested that we make our "TEXTechography" available on disk in ASCII format. Our "TEXTechnician," Editor Arthur A. Molitierno, and I have had numerous discussions regarding not only that but also about putting the entire *TEXT Technology* on disk.

Beginning this month, "TEXTechography" will be made available to subscribers for $6.00 per issue. This price includes first-class postage and handling and your choice of an MS-DOS formatted disk in either 3.5" (720k) or 5.25" (360k) size. You ***MUST BE A SUBSCRIBER*** to receive "TEXTechography" on disk, and your payment must be either in cash or by check—no purchase orders, please, since they cost us almost as much to process as the amount you are paying for the on-disk "TEXTechography."

As far as our putting the complete *TEXT Technology* on disk, I have some reservations. First, one of the reasons for *TEXT Technology* is to show how a newsletter can be produced from one's desktop. Second, I'm too vain to give up complete contextual control. I enjoy laying text and graphics down in Aldus *PageMaker*. It's theraputic, a welcomed break from teaching.

This doesn't mean that someday *TEXT Technology* won't be available electronically. It just means that for now, enjoy the issues as they come, when they come. If you ever experience problems with receiving your issues (January, March, May, July, September, and November), give our editorial assistant, Stephanie, a call. She will correct the problem. Thanks.

TEXT TECHNOLOGY

AD DOCENDUM • INVESTIGANDUM • SERVIENDUM • WRIGHT STATE UNIVERSITY 1967

Editors

Jim Schwartz
Arthur A. Molitierno

Contributing Editors

Randal Baier
Cornell University
Paul Fritz
University of Toledo
Eric Johnson
Dakota State University
Stephen Miller
Oxford University
Brad Morgan
South Dakota School of
Mines & Technology
Guy Pace
Washington State University
Ian Richmond
University of Western Ontario

Editorial Assistant

Stephanie Hampshire

ISSN: 1053–900X. Also indexed by ERIC, COMPUTER AND CONTROL ABSTRACTS (INSPEC), COMPUTER LITERATURE INDEX, LITERARY CRITICISM REGISTER, and SOFTWARE REVIEWS ON FILE. *TEXT Technology* is published bimonthly by Wright State University, 7600 State Route 703, Celina, Ohio, USA 45822-2921; (419) 586-2365. Postage paid at Celina, Ohio. SUBSCRIPTION RATES US funds: $20 (US); $27 (Canada); $35 (Foreign). Address all subscription inquiries and manuscripts to the Editors, *TEXT Technology*, Wright State University, Lake Campus, 7600 State Route 703, Celina, Ohio, USA 45822-2921. Note that the Editors are not responsible for any unsolicited manuscripts. Allow 4 to 6 weeks for subscription processing. Selected back issues are also available; contact the Editors for more details. POSTMASTER: Send address changes to *TEXT Technology*, Wright State University, 7600 State Route 703, Celina, Ohio, USA 45822-2921.

TEXTTechnology is composed on a Northgate Elegance 486-25 computer and printed using an IBM Personal Page Printer II Postscript printer. Images and text are scanned with a Hewlett-Packard ScanJet Plus scanner. Software used includes Microsoft *Windows*, Aldus *PageMaker*, *Caere Omnipage 386*, *WordPerfect*, *H-P Scan Gallery*, Micrografx *Designer*, *CorelDRAW!*, Lotus *Ami Professional*, and Symantec's *Q&A*.

WRIGHT STATE LAKE CAMPUS

Wright State University
Lake Campus
Celina, Ohio, USA 45822

This publication is funded in part by a research grant from the *Wright State University President's Club.*

COLUMN ONE

Eric Johnson

Icon for the Macintosh

Icon is a relatively new computer programming language that is well suited for text processing. The name "*Icon*" is not an acronym; it's an appropriate name for a language that can represent and generate strings of characters that serve symbolic functions as text, music, or graphics. It is an accident that the word "icon" is also used for the little pictorial representations on the screens of the Macintosh and other computers.

The *Icon* programming language was developed by Ralph Griswold and others at the University of Arizona. It is obviously related to the *SNOBOL4* language which Griswold and others created at Bell Laboratories. *Icon* (like *SNOBOL4*) is designed expressly for non-numeric computing: searching and analyzing texts, formatting and converting files, and generating strings (of music or poetry).

Although it was first developed for UNIX systems, *Icon* is now available in implementations for virtually all mainframe operating systems and for all microcomputers with sufficient processing power and memory. There is a good MS-DOS version of *Icon*, and a special version was implemented for 386 and 486 machines; the best *Icon* for the Macintosh is called *ProIcon*.

Most of the implementations of *Icon* are public domain and are available for a modest distribution fee from the University of Arizona, although *ProIcon* was commercially developed and is sold for $175. Far and away, the most interesting version of *Icon* is *ProIcon* for the Macintosh.

Users launch *ProIcon* by double clicking on its icon. A logo briefly appears, and then users are given either an open untitled file in which to create an *Icon* program, or they are given a menu from which a program file can be selected. Programs in *Icon* need not be long. On the next page is the complete *Icon* code to count the number of words in a file (see *FIG. 1*).

The fact that *Icon* programs can be written in lower-case letters—reserved words like "while" and "do" must be lower case—identifies it as a language developed after the days when the only real computing was done on mainframes. While all mainframe computers have been able to store and print both upper- and lower-case letters for years and years, it is still often a terrific bother to program in lower case on a mainframe.

Perhaps because compilers for all three languages were developed at about the same time, *Icon*, *Pascal*, and *C* programs look somewhat alike. Like *Pascal* and *C*, *Icon* is a structured language that uses control structures such as "while...do..." to direct the execution of the program.

The sample program to count words works like this. An *Icon* program always starts with a main procedure; other procedures may be called from there, although there are no others in this program. The variable "letters" holds the characters that can be parts of words: the 26 lower-case letters (supplied by the keyword "&lcase") and the 26 upper-case letters (supplied by the keyword "&ucase") plus the numbers, apostrophe, and hyphen. A backslash must be used before the apostrophe to indicate that it is not an enclosing single quotation mark.

The program's fourth line is a "while...do..." loop that controls the rest of the program, except for the last two lines. It works in this way: *while* input can be read into the variable called "line," and

```
procedure main()
   letters := &lcase ++ &ucase ++ '0123456789\'-'
  wcount := 0
  while line := read() do
    {
    position := 1
    while position2 := upto(letters,line,position) do
      {
     position := many(letters,line,position2)
      wcount +:= 1
      }
    }
  write("There are ",wcount," words in the ms.")
end
```

Fig. 1: *ICON* program code.

each time input is read, *do* the following. Each time variables "position" and "position2" indicate the starting and ending position of an uninterrupted span of letters (that is, when they indicate a word), one is added to "wcount." When there is no further input to read, the program drops out of the loop, writes the statement telling the number of words counted, and then stops.

ProIcon programs are very straight forward to run on the Macintosh. A menu can be pulled down to select a file for input. Other menus can be used to select various compiling options. Then another menu is used to run the program. A window opens to show the output. Using this article as input produces the following result: "There are 970 words in the ms." Using a Macintosh IIcx which has a 68030 processor, the program runs at a blinding speed, and the output appears almost at once.

If there were a blunder in the program, an error window would pop up with a message. When the user returns to edit the program, the cursor is put on or near the offending line. Help is available online, either from a menu or by highlighting a keyword in the program and asking for help on that specific topic.

It would be very easy to edit the sample program to make it print each word as it is found: only a single line of *Icon* coding would have to be added. However, a *ProIcon* program that writes a series of lines directly to the screen of a Macintosh will run rather slowly. It would also be easy to collect the words as they are found and sort them alphabetically before output.

In addition to its performance on a Macintosh, *ProIcon* is interesting because it has a series of extensions to the *Icon* language: 16 window functions and 12 other functions. From within an *Icon* program, the user can open, close, move, and arrange windows; the user can launch other applications, change the font in a window, create a dialog box, get keyboard input, or use trigonometric functions.

More information about implementations of *Icon* can be obtained from *Icon* Project, Department of Computer Science, Gould-Simpson Building, University of Arizona, Tucson, AZ 85721.

* * * * * * * * * *

Eric Johnson is Professor of English and Dean of the College of Liberal Arts at Dakota State University, Madison, SD 57042. He is the Director of the International Conference on Symbolic and Logical Computing, and he has published articles and reviews about computer programming, writing, and literary study. Eric's BITNET address is ERIC@SDNET

THE RESEARCH INTERN

Paul A. Fritz and Jeffrey D. Peters

The "Paperless" Desk

Students' study desks may resemble Alpine peaks ripe for an avalanche. Pity a student cramming for exams amid a desktop weighted down by syllabi, class notes, 3x5 index cards, ragged spiral notebooks, and form letters from Dean, Registrar, Bursar, and Parking Services (Hockey, 1983; 1985). The computer could help students control this "data storm." Using a computer as an "electronic desk," students could function in a class more efficiently, manage research more productively, and prepare systematically for their careers.

Class Notes

Students who manage lecture notes effectively generally earn good grades. Instead of copying down "everything the professor says," the real skill in note taking lies in revising them (Gerrard, 1987).

During lectures, students should jot down class notes on scratch paper. Immediately after a lecture students revise their notes at the campus computer lab or their residences. Students make a file for each course and revise the lecture into "cepts"—the smallest building block of a concept.

As an example, a professor's lecture may contain this "cept": "Proofs of proximity." These arguments are recognized by audience members as "in-statements"—known by a specific audience sub-group. The more proximity proofs a speaker uses, the higher the chances of audience compliance with the speaker's goals.

For instance, auto workers know the expression "the job you save may be your own" as an implied plea to purchase American-made cars. A union leader may use this phrase when speaking in Detroit but may omit it when speaking to a free trade organization in Washington, D.C. This "cept" would be revised on computer to read:

Proximity: "in-language" —> compliance
Detroit = "save job"

Each cept should contain the following: an identification or definition; an explanation showing how the term works; an illustration in the student's own words. Each cept is ranked and ordered into main and subheadings of larger categories (called "concepts," of course) mentioned in the lecture. Thus, a student's notes may look like this:

[CONCEPT] Appeals to involuntary attention:

[CEPT] Significance proofs: appeals to universal audience flags/freedom

[CEPT] Proximity: "in-language" —> compliance
Detroit = "save job"

[CEPT] Concrete proofs: picture what CAN happen to audience—car accident drunk driving

Studying Class Notes

Daily review of notes helps students avoid last minute "cram" sessions. Our students blanch visibly when we ask them to read lecture notes over each day prior to class. However, if notes are correctly abstracted, the daily review can take just a few minutes and exponentially reduce exam preparation time.

Students could reduce exam study time by making a small "memory chart" file of each course's notes. Students duplicate the "cepts" notes above. They erase all definitions and illustrations and leave only the term names. Now they can scroll through the memory chart testing

their recall of definitions, explanations, and illustrations. When students draw a blank on a term, they "window" the master cept file below the memory file on the terminal screen and restudy the troublesome term.

Data Storage

Term papers generally come due during the last week of a quarter. This deadline couples with approaching exam dates and increases student stress. More to the point, student research frequently is viewed as a detached exercise to demonstrate documentation techniques. Once the papers are graded, students may toss the research paper away.

We suggest a different orientation for student research. Universities, at the start of students' careers, should ask them to identify six to eight broad, socially significant topics for research throughout their college careers. They read and abstract information about these topics and then write research papers drawn from this growing store of current information. Students should read a source, make an abstract of the information, and fold it into their growing databases. This exercise also trains students using online databases since many colleges now link student computers to their library card files and periodical collections (Purcell, 1988).

Each entry of the database could be stored with a ready-made citation heading on the top of the abstract file. This heading could be copied to the "References Cited" section of the student's term paper. The heading could be written in both American Psychological Association or Modern Language Association formats, ready for easy insertion to a waiting term paper, if the student could not afford more sophisticated bibliographic database programs.

If you are an academic advisor, you could help students make wiser vocational choices if you could view students' databases to learn of their reading interests. Instead of intuiting what courses a student may need for career preparation, the advisor could survey the quality and direction of the student's abstracts and make suggestions based on student interest. Instead of charting students on "career paths," advisors could begin to chart them on "interest paths."

Networks

As campuses become larger, one way to build a sense of campus community may be through computer bulletin boards and e-mail. The bulletin board, of course, would give students a chance to communicate with each other and also allow students to communicate with their professors (Robbat, 1986). As a student studies lecture notes, he or she may ask, "I wonder what Doc Jones means by this idea? I don't get it!" At that point, instead of waiting until the morning to call Professor Jones, the student's question could be rewarded immediately. The student types out a memo to the professor and sends the question to the prof by e-mail. This written communique would be a good exercise in clarifying the question in the student's own mind.

Administrative Functions

The flood of catalogues, semester course offerings, and form letters could be stemmed if students could link directly to the college's central administration. If students had a course schedule file shipped to their terminals at registration time, they could select courses according to time preferences and register for courses of the next term. If a student's financial account at the university were current, the course fees could simply be charged to his or her account.

"Will this course fit my degree program?" is a common question juniors and seniors ask advisors. As students approach graduation, they must select courses that fit departmental and university requirements for the baccalaureate. The university audit of final courses would save both students and advisors the time and frustration of filling out final "check sheets" as students prepare for graduation. Several degree audit programs exist. An excellent one is the Miami (of Ohio) University Audit program which enables students to check course degree progress as they move from freshman to senior level. If this progress could be checked at the student's terminal, the student might be spared unpleasant surprises at graduation.

Career Focus

Students could make a file or subdirectory of vocational concerns. A small file could be made of sample term papers to illustrate their writing skills. Another file could contain letters of reference. Another could contain lists of standardized test scores (GMAT, LSAT, GRE, etc.), lists of courses taken arranged by general topic, and grades. Still another file could contain descriptions of extracurricular activities and intern projects. Of course the students would include their resumes in the career file. Students could also create a file of letter templates for Contact letters, potential contacts' addresses, "thank you" letters, and follow up letters. The flexibility of this file arrangement permits students to compile a vocational "portfolio" of credentials appropriate to different job recruiters.

How To Get This Idea Started

One way to generate interest in the "electric desk" is to involve the students themselves. Announce a campus-wide contest for proposals on "How to Eliminate Paper on Campus." Students would need to estimate the dollar savings achieved if their proposals were implemented (Groff, 1983; Skarulis & Thomas, 1980). Play off inter-university rivals. Two neighboring universities could compete and offer a substantial prize (say $500 worth of tuition vouchers) to winning proposals. The contest could be a way for students to display their problem-solving skills to the community.

Professors may want to submit similar proposals for industrial funding. Tandy corporation encourages teaching experiments using computers. Of special interest in academic literature is analysis of qualitative data generated in class settings (Dow, 1982). For more information, contact Tandy Educational Grants Program, 1600 One Tandy Center, Fort Worth, Texas 76102 for the grant topic "Using Microcomputers for Classroom Management to Increase Teacher/Student Productivity. The Dean of the Faculty at University of Arizona (Kolodny, 1991) conducts an annual contest among her departments to propose innovative ways to move professors' classrooms into the 21st century.

A paperless desk would benefit professors, students, and administrators. After all—students are not the only academics who might encounter a desktop avalanche.

References

Dow, J. (1982). "The combined use of computers and audio tape recorders in storing, managing, and using qualitative verbal ethnographic data." Paper presented at the Annual Meeting of the American Anthropological Association, Washington, DC.

Gerrard, L. (Ed.), (1987). *Writing at Century's End: Essays on Computer-Assisted Composition*. Manchester, MO: McGraw-Hill.

Groff, W. H. (1983). "Assisting a college's service area in the transition to the new technology society through strategic planning and manage-ment." Position paper. ED231453. Marion, OH: North Central Technical College.

Hockey, S. M. (1983). *A Guide to Computer Applications in the Humanities*. Baltimore: Johns Hopkins Press.

Hockey, S. M. (1985). *Snobol Programming for the Humanities*. NY: Oxford University Press.

Kolodny, A. (1991, February 6). Colleges must recognize students' cognitive styles and cultural backgrounds. *The Chronicle of Higher Education*, 37.

Purcell, R. (1988). "Both Automation and Paper." *Library Software Review*, 7, 399-401.

Robbat, R. J. (1986). *Computers and Individual-ized Instruction: Moving to Alternative Learning Environments*. Eugene, OR: International Council for Computers in Education, University of Oregon.

Skarulis, P. C. & Thomas, C. R. (Eds.) (1980). *Productivity—A Key to Survival in the 1980: Proceedings of the 1980 CAUSE National Conference*. (Phoenix, Arizona, December 7-10, 1980). Boulder, CO: CAUSE.

* * * * * * * * * *

Paul A. Fritz is an Associate Professor of Communication at the University of Toledo. Jeffrey D. Peters is one of Paul's research assistants. Besides being an "avid" Mudhens fan, Paul does research in critical thinking methods and in medical communication (aged/frail adults). He may be reached at (419) 537-2006 (office) or through BITNET--FAC0287@UofT01

THE CLIPBOARD

Ian Richmond

Parlez-vous . . .

Parlez-vous français? Sprechen Sie Deutsche? ¿Habla Usted español? Whichever language you speak, if it's not English, entering it from your MS-DOS computer's keyboard can be a royal pain in the *derrière*.

If you use only a wordprocessor, you may not have much of a problem. *Nota Bene* and *WordPerfect*, in particular, have superb foreign-text handling capabilities, including correct alphabetical sorting, spellchecking, and keyboard redefinition. Even the inexpensive *PC-Write* has a well-designed method for entering foreign characters from the keyboard. Most others let you figure out how to enter foreign text for yourself.

Even if your wordprocessor is a plurilinguist's dream, once you leave it to run another application—a database, outliner, or note-taking program—you'll find that computing is still very much an English-only world. Most often, such programs provide no facility for entering foreign text from the keyboard, short of pressing the Alt key while laboriously typing each character's ASCII code on the numeric keypad.

Some of these programs do claim to "facilitate" (read: "grudgingly allow") entry of foreign characters. For example, *GrandView* lets you call up an ASCII chart from a menu three levels deep and then select the character you want by highlighting it or typing its ASCII code. The recently released version 2 of this program greatly improves its predecessor by incorporating recognition of the MS-DOS foreign keyboards.

On the other hand, Pro/Tem's *Notebook II* wants no truck with any un-American keyboards. It makes you define Alt-function-key combinations or else use a special entry mode in which you can use the Alt-numberpad method or else enter the character from one of the MS-DOS keyboards. Of course, until you exit from the special entry mode, the normal editing keys, including the cursor keys, Backspace, Delete, and Enter, do nothing but place their corresponding ASCII symbols into your text. So don't try to correct any typing errors.

For typing an occasional foreign character, this method may be barely acceptable, but if you take notes mainly from foreign-language sources, you will quickly tire of it. I used *Notebook II* for about two weeks before retiring it in disgust. Obviously unrepentant, Pro/Tem recently sent me an upgrade notice promising that the new version allows even more definable Alt key combinations. Let's see. Is "é" Alt-E, or is it Alt-A for "acute"? No, that's "à", unless it's Alt-F5... I'll pass, thanks.

It's a mystery to me why some programmers take such pains to prevent people from using the foreign-language keyboards included with most versions of MS-DOS. The beauty of these keyboards is that they enable those working in a European language other than English to use the *same* keyboard with all their applications. You no longer have to learn one keyboard layout for, say, *WordPerfect*, a set of arcane Alt key combinations for your note-taking program, and yet another set of Alt or Ctrl macros for your outliner.

As long as your applications support the DOS foreign keyboards, you need only load the one you want through your AUTOEXEC batch file. With so many programs available that do support these keyboards, it's absurd to waste time with one that makes you use a crib sheet or mess around with special entry modes just so you can use it to do what you bought it for.

One drawback to the MS-DOS keyboards is that you can't reconfigure them, unless you're familiar with assembly language and the wondrous ways of DOS's DEBUG program. A perhaps more serious problem for most people is that the lack of several common symbols on the foreign-language keyboards requires you to switch back and forth between the default US layout and the selected foreign one. If the latter is not in the familiar QWERTY format, a lot of typos can result from this switching.

Despite these problems, you may well find that a DOS keyboard is just what you need. If the only foreign language you use is French and you like the exoticism of the AZERTY keyboard layout, then DOS's French keyboard is just the ticket. Similarly, if you need only French but hate switching between AZERTY and QWERTY layouts, then try the Canadian-French version. Don't, however, expect to type any German or Spanish.

Neither of these keyboards will place an umlaut on anything but the three vowels "eiu." On the other hand, if you choose the German keyboard, you can't type some of the French characters. Selecting the Swiss keyboard forces you to choose between Swiss-French and Swiss-German and also gives you a layout that is neither quite AZERTY nor quite QWERTY.

The Spanish and Latin-American keyboards, on the other hand, by making liberal use of "dead" accent keys, are able to produce most, if not all, of the characters you need for the main European languages and keep the familiar QWERTY layout besides.

If none of the DOS keyboards quite meet your needs or taste, another possibility is to use a keyboard configuration program. One of the best of these is produced by a Canadian company with the unlikely name of Mikholen Answerk (honest!). Called *Interpose*, the program can be obtained from the publisher as part of a $35-dollar "Sample Pack." *Interpose* alone is worth more than the cost of the package.

With it, you can define "dead" accent keys as Ctrl or Alt combinations. This method gives you a keyboard able to produce any of the foreign characters in the Extended ASCII set without affecting the default US keyboard. This means you don't have to switch keyboards when you need a character—like a backslash, a square bracket, an angle bracket, or a curly brace—that is unavailable on most DOS foreign keyboards.

Working with foreign-language text on MS-DOS computers is not nearly as easy or convenient as it should be, although we've come a long way compared to five or six years ago. Whether we progress much further depends on how loudly we foreign-language users protest the obstacles programmers throw in our path.

For an alternate keyboard-configuration program, contact Mikholen Answerk Computer Systems Inc. at 63 Glynn Avenue, Ottawa, Ontario K1K 1S7 CANADA (ph. 613-749-7214).

* * * * * * * * * * *

Ian M. Richmond is a member of the Department of French at the University of Western Ontario, London, Ontario, Canada N6A 3K7. He may be reached at 519-661-2163, Ext 5703 as well as BITNET: IMR@UWOVAX

Articles & Reviews Welcome

TEXT Technology **welcomes article submissions that pertain to wordprocessing, text-analysis, and research applications in professional writing situations, either corporate or academic. Also, hardware and software reviews are encouraged, but please contact Jim Schwartz before submitting them (call Jim at 419-586-2365 or send him a note at JSCHWARTZ@WSU.BITNET). Manuscripts should be submitted on MS-DOS 5¼" or 3½" floppy disks, through BITNET, or in hardcopy format.**

PERFECT TECHNIQUES

Guy Pace

Converting Documents from *TEX* to *WordPerfect*

My least favorite activity is to take some text file, strip out hard carriage returns, and reformat the file for my wordprocessor. When I first started using *WordStar*, in 19—*mumble mumble*—I would take plain text and convert it to a *WordStar* document, hard return by hard return.

The task was boring and repetitive, even when I used the then powerful capabilities of *WordStar* to strip files. Someone came along with a program that stripped *WordStar* files of formatting code and rebuilt text files for *WordStar*. That worked well enough. When *WordPerfect* and Microsoft *Word* and *XYWrite*, *Write*, *Volkswriter*, etc. came along, we needed more sophisticated software for converting one wordprocessor format to another and back. My personal favorite for document conversion is *Software Bridge*. It is relatively easy to use and performs beautifully.

I learned that converting text files is more difficult than just stringing together a bunch of search-and-replace commands.

However, conversion programs do not solve all the problems. In the university environment, or any other mainframe environment where *TEX* and its kin exist, we still deal with text files. Like *PerfectWriter* files, *TEX* files are straight ASCII files with formatting markup, similar to typesetting codes, mixed in with the text.

For instance, to center the word "the" on a line by itself in *TEX*, you would use the markup:

\centerline{the}

Of course, *WordPerfect* doesn't understand this at all. *Software Bridge* looks at *TEX* files as ASCII files. If you belong to the *TEX* User Group (TUG), you know there are programs available to convert *WordPerfect* 5.0 and 5.1 files to plain *TEX*, *LaTEX*, and *TEXT1*. There is nothing available I know of to change *TEX* to or from *WordPerfect*.

Recently, a professor dropped in with about 20 disks full of text files created in *TEX* which were downloaded from a mainframe. Her concern was to be able to take these files, her doctoral dissertation and supporting research, and edit them into research papers for appropriate journals. She was also concerned because it cost more to print from the mainframe than to preview on the PC.

My solution to the problem was to create a macro in *WordPerfect* 5.1 that converted most of the markup in the files to *WordPerfect* codes. I created a single macro to accomplish the job, but a series of macros might be easier to create and debug.

Here's how I approached the problem. I created a macro (Ctrl-F10) called CLEANUP. The initial macro moved to the top of the file, then executed a loop that searched for "\vs," which is markup for vertical skip, deleted to the end of the line, and lastly inserted two hard returns. For a starting point I saved the macro (Ctrl-F10) and then edited it (Ctrl-F10, CLEANUP, 2).

I learned that converting text files is more difficult than just stringing together a bunch of search-and-replace commands. I was more successful with a series of search commands, followed by formatting commands.

The routines I created repeated themselves until the job was done. The structured looping of the macro reminded me of BASIC programs I wrote a long time ago.

My macro program looked like this:

```
{;}This cleans up most of the markups in TEX and
TEX-style files.
{PROMPT}We're on out way!~
{DISPLAY OFF}

{Home}{Home}{Up}
{ON NOT FOUND}{GO}Resume~~
{LABEL}Loop~
   {Search}\vs{Search}{Home}{Left}{Del to
EOL}{Enter}{Enter}
{Go}Loop~
{LABEL}Resume~

{Home}{Home}{Up}
{ON NOT FOUND}{GO}Again~~
{LABEL}Loop1~
   {Search}\centerline{{Search}
   {Block}{LeftSearch}\{Search}{Left}{Del}y
   {Block}{Search}}{Search}{Center}y{Left}{Del}
{Go}Loop1~
{LABEL}Again~

{Home}{Home}{Up}
{ON NOT FOUND}{GO}DOIT~~
{LABEL}Loop2~
   {Search}\it{Search}
   {Block}{LeftSearch}\{Search}{Left}{Del}y
   {Block}{Search}}{Search}{Font}24{Left}{Del}
{Go}Loop2~
{LABEL}DOIT~

{Home}{Home}{Up}
{Replace}n\newpage{Search}{HPg}{Search}
{Home}{Home}{Up}
```

Loop searches through the document for "\vs." When Loop find "\vs," it goes to the left column, deletes the line, and inserts hard returns. Loop1 searches for markups that center a line. Loop1 finds the markups, erases them, blocks the target text, enters in the *WordPerfect* command, and kills the closing brace.

Loop2 looks for italics markups. When Loop2 finds the italics markups, it erases the markups, blocks the target text, italicizes it, and then kills the closing brace.

The program's last routine goes to the top of the file and executes a simple global replace of "\newpage" with a hard page command.

The "{Go}Loop~" commands provide program looping capability. The "{ON NOT FOUND}{GO}DOIT~~" construction controls program flow on "not found" error conditions. Please note the use of the tilde in the command syntax. The commands are accessible from the macro editor in *WordPerfect* by menu with Ctrl-PageUp. Macro command reference in the *WordPerfect* manual begins on page 845, in Appendix K.

As I mentioned earlier, it would be easier to test and debug a series of macros. Test each routine for accuracy before adding it to the macro "chain." With that approach, each of the above routines could be in a separate macro, with a "{CHAIN MACRO}Whatever~" command at the end of each. The first macro runs and then calls subsequent macros until the entire process is complete.

I did add more than one "{BELL}" to my macro so I could track its progress. The "{PROMPT}" command flashed its message too quickly. The audible bells were a better sign of progress. For a specific set of documents as with my professor friend, a custom macro takes the tedium out of converting files.

Now, to get *WordPerfect* to make the perfect cup of coffee in the morning. . . .

* * * * * * * * * *

Guy L. Pace lives in Pullman, WA, with his wife and family. He is an Information Services Evaluator at Washington State University, Cooperative Extension. A large part of his job involves helping people learn to better use computer hardware and software. His degree is in public relations, and he was a journalist (off and on) for 15 years. His teenagers think he is weird, so he must be normal.

TEXTechography

Arthur A. Molitierno

[Ed Note: Unless otherwise stated, all articles are from the USA. All programs and software are italicized. For all journals, the "series," "issue" and "number" are labeled as "n.," with the number, series, or issue of the journal following the abbreviation. In all cases, the volume precedes "n." Each issue's bibliography is also available on 3.5" (720k) and 5.25" (360k) diskettes in MS-DOS format to TEXT Technology *subscribers for $6.00 US, cash or check—no purchase orders, please.]*

Abbreviations:

Countries

CN ..... **Canada**
CZ ..... **Czechoslovakia**
FR ..... **France**
JP ..... **Japan**
SW ..... **Switzerland**
WG ..... **West Germany**
UK ..... **United Kingdom**

Terms

DTP ..... **Desktop Publishing**
np ..... **no page**
n. ..... **number**

Months

JA ..... **January**
FE ..... **February**
MR ..... **March**
AP ..... **April**
MY ..... **May**
JE ..... **June**
JL ..... **July**
AU ..... **August**
SE ..... **September**
OC ..... **October**
NO ..... **November**
DE ..... **December**

Acerson, K. "Review of WordPerfect *5.1: Complete Reference*." (McGraw-Hill, 1990) in *Computer Book Review* 8 n.10 (1 OC 1990): np.

Alonso, R. "Word Processing Comes to OS/2 and PM." *Personal Computing* 14 (27 AP 1990): 158. [review of *Describe* wordprocessor for *Presentation Manager*]

Anonymous. "Hypertext Standardization Workshop." *Journal of Research of the National Institute of Standards and Technology* 95 (MY-JE 1990): 345-48.

Anonymous. "Software for Workstations: Preparing for Future Graphics." *IEEE Computer Graphics and Applications* 10 (JL 1990): 11-82.

Anonymous. "*Powerpoint* 2.0 Uses *Windows* 3.0 for PC Presentation Graphics." *Computer* 23 (JL 1990): 109.

Anonymous. "Graphical Word Processors." *Computer* 28 (MY 1990): 87-90. [*Ami Professional*, *WordPerfect* 5.1, *Win Word*]

Anonymous. "Desktop Publishing: Dressing Up a Document." *Personal Computing* 14 (AU 1990): 142. [buyer's guide to software]

Bach, M. "Optimizing Hardcopy." *Mikrocomputer Zeitschrift* (WG) n.4 (AP 1990): 160-165. [in German, use of Turbo-Pascal screen-copying programs]

Bell, J. "DTP for Mere Mortals." *Personal Computing* 14 n.4 (27 AP 1990): 156. [reviews *Publisher II*, software which integrates desktop publishing and wordprocessing]

Berk, E. "A Worthy Guide to Hypertext: *Guide* 3.0." *Personal Computing* 14 n.8 (AU 1990): 129. [reviews Owl International's *Guide* 3.0, a hypertext tool which supports programming language, importing and exporting texts and graphics]

Birkholz, I. "Quicker Success Using *Word* 5.0." *Chip* (WG) n.3 (MR 1990): 374-80. [in German]

Birkholz, I. "Quicker Success with *Word* 5.0." *Chip* (WG) n.5 (MY 1990): 268-276. [in German, examines graphics supported by *Word* 5.0]

Bobker, S. "Canned Art: Clip Art for the Mac." *MacUser* 6 n.9 (1 SE 1990): 40.

Bond, G. "*Legacy*: Processing Words under *Windows*." *BYTE* 15 (SE 1990): 121-22.

Bornstein, H. "Goof-proof Writing." *MacUser* 6 (SE 1990): 242-44. [reviews spell-checkers, thesauri, and grammar-checkers for Macintosh]

Bradbury, J. "Finishing Touches." *MacUser* 6 (JL 1990): 216-19. [overviews index features of *Word* 4.0, *Nisus* 2.03, *FullWrite Professional* 1.1, and *WordPerfect* for the Macintosh]

Brambert, D. et al. "Page by Page: The Latest (And Hottest) Page Makeup Software." *Electronic Publishing & Printing* 5 n.5 (JE-JL 1990): 20-28. [reviews *XPress* 3.0, *PageMaker* 4.0, *DesignStudio*, and *PageComp*]

Bryan, M. "New *MultiMate* Keeps Its Basic Form." *Personal Computing* 14 n.5 (25 MY 1990): 164. [reviews *MultiMate* 4.0]

Burgsdorff, B. Kuhn-von. "Painting in Pixels." *Personal Computer* (WG) n.4 (AP 1990): 126-27. [in German, overview of *Sled* 2.0, graphics program]

Campbell, G. "Words into Print...." *PC World* 8 n.9 (1 SE 1990): 225-226. [tips for wordprocessing]

Carlisle, B. H. "Word Processor Caters to Engineers." *MachineDesign* 62 (10 MY 1990): 26. [*WordPerfect* 5.1]

Cisler, S. "*Marco Polo*: Managing Electronic Documents in a Group." *Online* 14 n.5 (SE 1990): 101-04. [reviews features of *Marco Polo*, software for Macintosh for creating and managing an electronic document center with up to 255 users and documents up to 32,000 pages]

Cook, R. "A Bold New Look for Word Processing." *Datamation* 36 (13 SE 1990): 35-36.

Cwiakala, M. "Using Hypermedia to Enhance CAD." *Mechanical Engineering* 112 (SE 1990): 44-47.

Danuloff, C. "19 Ways to Customize *Word*." *MacUser* 6 (SE 1990): 302-305. [customizing menus in Microsoft's *Word* 4.0 for Macintosh, includes tutorial]

Davies, R. "Database Publishing on a Desktop." *Canadian Journal of Information Science* (CN) 15 n.1 (AP 1990): 37-48. [overview of various aspects of producing camera-ready copy for database publishing]

Dickinson, J. "Tomorrow's Word Processors Today: the Graphical Advantage." *PC Magazine* 9 (JL 1990): 95-115. [review of *Word* for Windows, *Ami*, *Ami Professional*, *Legend*, and *Describe*]

Dolbear, G. "The Expandable Page." *Chemical Engineering* 97 (MY 1990): 157-8. [wordprocessing for engineers and scientists]

Eberlein, E. "Write and Save." *Personal Computer* (WG) n.4 (AP 1990): 76-79. [in German, overview of wordprocessors: *Textomat PC*, *Startexter Deluxe*, *PC-Text* 2.2]

Eckhouse, R. "The Infinite Font Cartridge." *Computer* 23 (JL 1990): 104.

Eckstein, H. "Winning the Desktop Publishing Game." *Catalog Age* 7 n.4 (AP 1990): 81-82. [considers cost of choosing desktop publishing]

Eglowstein, H., et al. "Word Processors That Build Character." *BYTE* 15 (SE 1990): 132-36. [overview of 15 WYSIWYG programs for Macintosh and PCs]

Eglowstein, H. "*PageMaker* Revs Up under *Windows*." *BYTE* 15 (JL 1990): 122.

Eglowstein, H. "*PageMaker* Never Looked Better." *BYTE* 15 (JE 1990): 117. [*PageMaker* 4.0]

Eickman, I. "Graphics in Text." *Mikrocomputer Zeitschrift* (WG) n.4 (AP 1990): 78-83. [in German, *Turbo-Pascal* program for color graphics]

Exline, A. "Computer Graphics." *IEEE Potentials* 9 n.2 (AP 1990): 43-45.

Field, C. E. "Flights of Fancy...Desktop-published Aerogrammes...." *InCider* 8 n.9 (1 SE 1990): 68-70.

Fourier, A. and S. Pfeitter. "Word Processing in Order—Eight Programs." *Mikrocomputer Zeitschrift* (WG) n.5 (MY 1990): 68-78. [in German, overview of *WordPerfect* 5.1, *Euroscript*, Lotus *Manuscript*, *Star Writer* 5.0, *Wordstar* 2000, *Wordstar* 5.5, *Ami Professional*, *Word* 5.0]

Fourier, A. "Ready Formulae." *Mikrocomputer Zeitschrift* (WG) n.5 (MY 1990): 64-66. [in German, examines *Scientex Publisher* and *Large Windows*, programs designed for chemical and mathematical formulae]

Gillikin, D. P. "Document Delivery from Full-text Online Files: A Pilot Project." *Online* 14 n.3 (MY 1990): 27-30, 32. [library acquisition of journal articles through electronic retrieval]

Gillman, P. "Making Labels." *C & L Applications* (UK) 3 n.9 (MY 1990): 4-6. [using *WordPerfect* 5.1 for address labels]

Gruberman, Ken. "Fast Forms and Informed Designer Fast Forms...." *MacUser* 6 n.9 (1 SE 1990): 73-75.

Gruman, G. and S. Timacheff. "Next Machine Creates Publishing Platform." *InfoWorld* 12 n.27 (2 JL 1990): 54-58. [review of *Frame Maker* Version 2.0, a documentation software marketed with Next computers]

Gruman, G. et al. "Publish or Perish: *InfoWorld* Looks at Six Low-End DTP Packages for the Nondesigner." *InfoWorld* 12 n.25 (18 JE 1990): 63-87. [reviews *Avagio* 2.0, *Express Publisher*, *Finesse*, *GEM Desktop Publisher*, *PFS: First Publisher*, and *Publish It*]

Heid, J. "Aldus *PageMaker* 4.0 Desktop Publishing Program." *Macworld* 7 n.9 (1 SE 1990): 225-28.

Heilbron, M. "Microsoft's *Win Word* Makes *Windows* Shine." *Computing Canada* (CN) 16 n.8 (12 AP 1990): 27. [review of *Win Word* with online hypertext feature]

Hepe, J. "Frame Guidelines." *Chip* (WG) n.5 (MY 1990): 132-42. [in German, overview of 24 DTP programs]

Hiatt, J. T. "What's New in Desktop Publishing and Graphics." *Computers in Accounting* 6 n.6 (AU 1990): 14-43 [overviews 14 desktop publishing programs, with company newsletters in mind]

Honan, P. "IBM LaserPrinter E: A Printer for All Seasons." *Personal Computing* 14 n.7 (JL 1990): 122. [reviews product suited for small offices]

Honan, P. "Desktop Publishing Power on a Budget: *Avagio* Publishing System." *Personal Computing* 14 n.8 (AU 1990): 127-128 [considers *Avagio* best suited for one-page newsletters]

Jantz, R. "The Art of Persuasion...High-impact Documents and Presentations." *PC World* 8 n.9 (1 SE 1990): 237-238.

Jones, M. "*Arts & Letters* Graphics Editor." *PC Magazine* 9 n.15 (11 SE 1990): 254-259.

Kawanda, T. "From Kana to Kanji: Word Processing in Japan." *IEEE Spectrum* 27 (AU 1990): 46-48.

Kendall, R. "*Letter-Perfect*: A Less-intimidating *WordPerfect* 5.1." *PC Magazine* 9 (16 OC 1990): 38. [description of WordPerfect Corporation's *Letter-Perfect*]

Kerlow, I. V. "Micrografx *Designer*." *PC Magazine* 9 n.15 (11 SE 1990): 277-292.

Kerlow, I. V. "Adobe *Illustrator*." *PC Magazine* 9 n.15 (11 SE 1990): 250-254.

Lauriston, R. "Fast Relief for Hypertext Headaches: *Folio Views* 2.0." *PC World* 8 n.9 (1 SE 1990): 141.

Lavin, P. "Word Processors." *Which Computer?* (UK) 13 n.5 (MY 1990): 115-18. [overview of *Wordstar*, *WordPerfect*, *DisplayWrite*, and *Word*]

Lavroff, N. "Screen Players." *MacUser* 6 (SE 1990): 248-50. [reviews specialized wordprocessing software for screen and script writing for Macintosh]

Lewis, D. "*Publish It! Easy Publish It!*...." *MacUser* 6 n.9 (1 SE 1990): 61-63.

Littman, D. and Tom Moran. "The Return of *Illustrator*." *Macworld* 7 n.9 (1 SE 1990): 129.

MacGregor, C. "*Wordstar* Gives Its Loyal Fans a New Look." *Computing Canada* (CN) 16 n.12 (7 JE 1990): 38. [reviews *Wordstar* 6.0]

Marshall, G. "Review of *Manager's Guide to Desktop Electronic Publishing*." (Prentice-Hall, 1990) in *Computer Book Review* 8 n.9 (1 SE 1990): np.

Mendelson, E. "Adobe *Streamline*." *PC Magazine* 9 n.15 (11 SE 1990): 274-75.

Mendelson, E. "*QuickTrace*." *PC Magazine* 9 n.15 (11 SE 1990): 275.

Mendelson, E. "*Ami, Ami Professional*." 9 *PC Magazine* (JL 1990): 98-100. [review of Lotus Development Corporation's two graphics wordprocessing packages]

Mendelson, E. "Still to Come: More GUI Word Processors." *PC Magazine* 9 (JL 1990): 128. [overviews graphical user interfaced wordprocessors]

Morris, R. "Image Processing on the Macintosh." *Computer* 23 (AU 1990): 103-07. [*IPLab, Image, Ultimage*]

Mullin, J. K. and D. J. Margoliash. "A Tale of Three Spelling Checkers." *Software: Practice & Experience* 20 (JE 1990): 625-30.

Naffiah, N. "Multimedia Applications." *Computer Communications* (UK) 13 n.4 (MY 1990): 243-49. [overview of trends in multimedia applications in publishing and imaging, including hypermedia]

Nielsen, J. "Review of *Hypertext and Hypermedia*." (Academic Press, 1990) in *Computer Book Review* 8 n.8 (1 AU 1990): np.

O'Connell, S. E. "Employee Communication and the Computer." *HRMagazine* 35 n.7 (JL 1990): 24-25. [use of *Publish It!* for small to mid-sized companies]

O'Malley, C. "Fujitsu's Dual-Personality Laser." *Personal Computing* 14 n.5 (25 MY 1990): 176. [reviews Fujitsu's RX7100PS laser printer which connects to either a DOS or Macintosh computer]

Quinn, R. "In-house Publishing Improves." *American City & County* 105 (AP 1990): 10.

Quraishi, J. "*PageMaker*: Aldus' Latest Version...." *MacUser* 6 n.9 (1 SE 1990): 46-47.

Ray, K. and J. R. Driscoll. "New Directions for Microcomputer-Based Hypertext Systems." *Database* 13 n.4 (AU 1990): 60-64. [reviews features of *SPIRIT*, Syntactic and Probabilistic Indexation and Retrieval of Text, a menu-driven information retrieval system using statistical and linguistic analysis]

Reinhardt, A. "Making Points Powerfully Through *Windows*." *BYTE* 15 (JL 1990): 118. [desktop publishing, *PowerPoint* for *Windows* 3.0]

Rosenbaum, D. J. "Getting Technical." *MacUser* 6 (SE 1990): 250-52. [overviews how various programs for Macintosh handle long, complicated documents, includes indexing and outlining]

Rosenberg, S. "New Adventures in Graphic Design." *BYTE* 15 (AU 1990): 198-99. [Micrografx *Designer* 3.0]

Roth, S. "*Publish It Easy* 1.10 Page-makeup Software." *Macworld* 7 n.9 (1 SE 1990): 228-232.

Roth, S. "*PageMaker* vs. *Quark Xpress*." *Folio: The Magazine for Management* 19 n.8 (1 AU 1990): 69-73. [reviews *Quark Xpress* 3.0 and *PageMaker* 4.0]

Samborn-Kaliczak, A. "Radius Pivot Monitor Easily Switches Modes." *InfoWorld* 12 n.17 (23 AP 1990): 86-88. [reviews monitor which rotates 90 degrees for portrait and landscape designs; screen image also rotates for portrait or landscape mode]

Scisco, P. "*WordPerfect* 5.1." *Compute* 12 (AP 1990): 85-86. [software review]

Simone, L. "*CorelDraw*." *PC Magazine* 9 n.15 (11 SE 1990): 261-270.

Slaney, M. "Interactive Signal Processing Documents." *IEEE ASSP Magazine* 7 n.2 (AP 1990): 8-20. [use of *Mathematica* for interactive mathematical and wordprocessing tasks in scientific documentation]

Slater, S. W. "Computer Support for the Co-authoring of Documents: '...as Much a Psychological Research Area as a Technical One'." *Information Media & Technology* (UK) 23 n.3 (MY 1990): 109-11. [use of hypertext for co-authoring]

Smart, H. "*Word* of Honour." *What Micro* (UK) (AP 1990): 52-54. [compares features of *Winword* to *Ami*]

Smart, H. "Not So Simple." *What Micro* (UK) (JE 1990): 66-8, 70, 73-4. [review of new desktop publishing program, *Deskpress*, which uses *GEM* or runs as stand-alone]

Smith, B. "Around the World in Text Display." *BYTE* 15 n.5 (MY 1990): 262-66, 268. [solutions to some textual problems created by various alphabets: Cyrillic, Hebrew, Greek, Arabic, etc.]

Stark, C. L. "The Working Word." *PC Magazine* 9 (AU 1990): 461-62. [describes a macro program for *WordPerfect* 5.0]

Stark, C. L. "The Working Word: Why WYSIWYG Won't Work Under *Windows*." *PC Magazine* 9 (10 AP 1990): 329-31. [describes shortcomings of *Windows* screen fonts, e.g. eye strain, for full-time writers]

Stark, C. L. "The Working Word." *PC Magazine* 9 (12 JE 1990): 379-80. [overviews *Companion*, utility, for *WordPerfect* users]

Stark, C. L. "High-powered Word Processing." *PC Magazine* 9 (24 AP 1990): 329-30. [overviews *Word* for *Windows*]

Todd, G. "Review of DisplayWrite *5 Made Easy*." (McGraw-Hill, 1990) in *Computer Book Review* 8 n.10 (1 OC 1990): np.

Treplin, D. "Print Optimiser." *Chip* (WG) n.5 (MY 1990): 100-1. [in German, test of *Bitmaster*, graphics program for converting 9 needle graphics to 24 needle graphics]

Udell, J. "*Ventura*'s *Window* of Opportunity." *BYTE* 15 (AU 1990): 118. [*Ventura Publisher, Windows* ed.]

Varesi, A. "Almost Professional." *Chip* (WG) n.5 (MY 1990): 288-92. [in German, using *Turbo-Pascal* for wordprocessing, graphics, and desktop publishing]

Walter, M. "The AP *Leaf Picture Desk*: Newspapers about to Embrace Digital Photography." *Seybold Report on Publishing Systems* 19 n.12 (12 MR 1990): 13-18. [overview of Associated Press' use of digitalized photographic system and *Leaf Picture Desk* software]

Wang, W. E. "Review of *First Book of Personal Computing*." (Sams, 1990) in *Computer Book Review* 8 n.9 (1 SE 1990): np.

Wasson, G. "Photographs and Clip Art on CD-ROM Clip-art." *MacUser* 6 n.9 (1 SE 1990): 98-101.

Wasson, G. "Six PostScript Tricks for Word Processors." *MacUser* 6 (AU 1990): 230-33. [includes features such as graying, reversed-text paragraphs, and shadowing text]

Wasson, G. "*Metamorphosis* Gives Type Designers and Artists Access to Adobe Type 1 Outlines...." *MacUser* 6 n.9 (1 SE 1990): 80-92.

Wasson, G. "*FontStudio* Letraset's Latest Addition to Its Studio Line...." *MacUser* 6 n.9 (1 SE 1990): 76-80.

Welsch, E. K. "New Technologies: Desktop Publishing and *WordPerfect* 5.1." *Computers in Libraries* 10 n.4 (AP 1990): 29-31. [aspects of *WordPerfect* 5.1 of use to librarians]

Wilson, M. "Mass-11 *Draw*." *PC Magazine* 9 n.15 (11 SE 1990): 270-277.

Wood, E. H. "Concept Finder: Search Capabilities Across Multiple Files." *Information Today* 5 n.8 (1 SE 1990): 14. [software review]

Wood, L. "Word Processing in *Windows*." *BYTE* 15 n.4 (AP 1990): 157-60. [overview of *Ami Professional* 1.0, *Word* for *Windows* 1.0, and *Legend* 2.0]

Yager, T. "A Frame of Reference." *BYTE* 15 (JL 1990): 114. [*FrameMaker* 2.0, Macintosh version]

* * * * * * * * * *

Arthur A. Molitierno is an Editor of *TEXT Technology*. You may contact him at Wright State University—Lake Campus, 7600 State Route 703, Celina, Ohio 45822-2921 USA (phone: 419-586-2365).

New Subscriber Information

To academic and corporate writers and teachers of writing, *TEXT Technology* brings analyses of microcomputer hardware and software, discussions of programming techniques (both in languages and in applications), book reviews, updates of significant events in computing around the world, bibliographic citations, and much more.

TEXT Technology, created by the editor of the *Research in Word Processing Newsletter*, also will become an information clearinghouse for subscribers' opinions and queries about personal computing during the '90s—and beyond.

Subscription rates for one year (6 bi-monthly issues—16 pages) of *TEXT Technology* are as follows:

US	$20
Canada	$27
Foreign	$35

All prices are in US funds

Please fill out the form below and send it with your check or purchase order to

Subscriptions Department
TEXT Technology
Wright State University—Lake Campus
7600 State Rte. 703
Celina, Ohio, USA 45822-2921

Name _____

College or Corporation _____

Street _____

City _____

State or Province _____

Zip Code or Postal Code _____

BITNET _____

Affiliation: **ACH** ____ **ALC** ____ **ALLC** ____ 03/01/91