6/10/93

Texas Instrument 99/4A and Myarc 9640 Computers

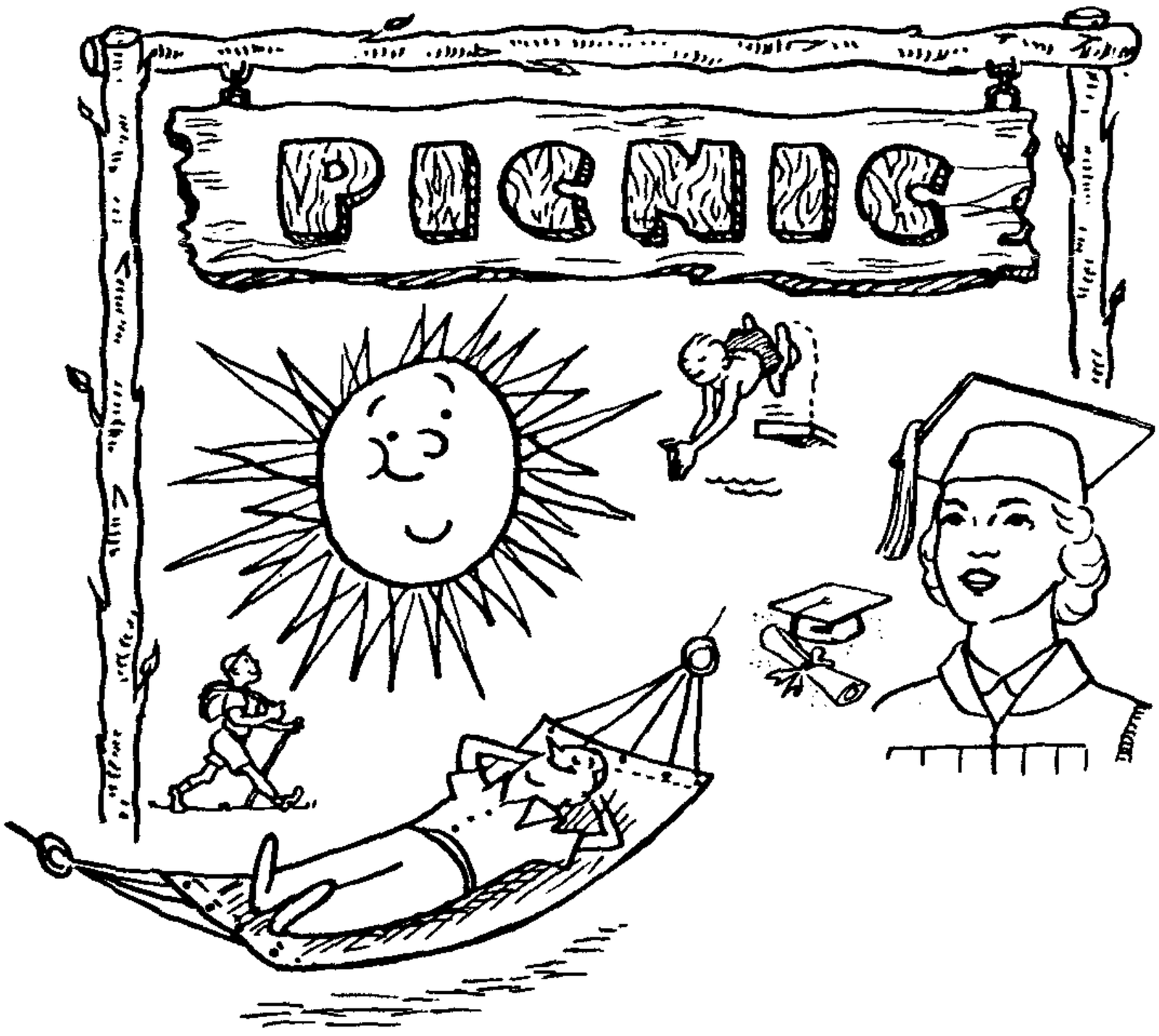
Spirit of 99

CENTRAL OHIO NINETY NINERS INC.

THE OFFICIAL NEWSLETTER OF THE CENTRAL OHIO NINETY-NINERS INC.

PUBLISHED MONTHLY IN COLUMBUS OHIO

"IN THE GOOD OLD SUMMER TIME"



$1.50 VOL. 11 NO. 6 JUN 1993

Central Ohio Ninety-Niners Incorporated (C.O.N.N.I.). Columbus Ohio 43212, USA. Spirit of 99 is published for Central Ohio Ninety Niners Inc. by C.O.N.N.I. members and is the official newsletter of C.O.N.N.I. User Group.

Editorial, address is:
181 HEISCHMAN AVE
WORTHINGION,OH 43085

Subscription rate (USA) $20.00 /1 year $25.00 /1 year outside continental U.S. Third class postage paid at Columbus, OH

CHANGE OF ADDRESS: NEW address to address listed above. We assume no responsibility for manuscripts,programs (tape or disk) not accompanied by return postage. Letters to the Editor are property of Spirit of 99. If published,we reserve the right to edit at our discretion.

OPINIONS EXPRESSED HEREIN ARE THE AUTHORS AND ARE BASED ON VALID DOCUMENTABLE RESEARCH,THEY DO NOT NECESSARILY REFLECT THE OPINIONS OF THE PUBLISHER.

We will not knowingly publish copyright material without the permission of the author and credit due.

All programs published herein are of public domain unless otherwise noted.

Other non-profit user groups may use material from this newsletter only if source and credit is given.

Central Ohio Ninety Niners Inc. is a non-profit organization comprised of MEMBERS who own or use the TI99/4A computer and it's related pro-ducts and have paid a yearly membership fee of $20 and whose main objectine is the exchange of Educational and Scientific information for the purpose of computer literacy.

C.O.N.N.I. meetings are held the 3rd sat-urday of each month at Chemical Abstract, 2540 Olentangy River Road Columbus, OH. Meeting time is 8:30 AM til 2:30PM, Meetings are open to the public. Membership dues ($20.00) are payable yearly to C.O.N.N.I. and cover the immediate family of the member. Please send check to our membership registerar and join C.O.N.N.I.

Please address it to:
Everett Wade
1/9 Erie Road
Columbus, OH 43214

CONTENTS

Check Sum.....P. 6
Convert Word Perfect.....P. 2
DV80 to Program.....P. 7
Father's Day.....P.15
Funnelweb V5.0/80 col....P. 8
Harrison Word Processor..P.15
Index.....P. 2
Lima Fair Review.....P. 3
Managing your Money-5....P.13
Memo Pad.....P.12
Print Using.....P. 9
Rave/Fast Term.....P 4.
TI-101 02.....P 5.

MORE ABOUT CONVERTING "WORD PERFECT 5.1" FILES TO TI DV80 FORMAT

by Charles Good
Lima Ohio Users Group

In my March 1993 review of Mike Wright's TI CYC I described a method of converting the MS-DOS CYC disk files to TI DV80 format usable by those who do not have access to Word Perfect 5.1. If you DO HAVE WP5.1 there are several ways to save a file from Wordperfect 5.1 to ASCII format so that the file can be converted with PC TRANSFER to DV80 on a TI disk.

--Use F5, save file, and select "Generic".
--Use F5, and select DOS to save the file.

--Format the file to your particular printer (the CYC is formatted for laser printer output). Then change the printer name to a disk file and PRINT the file to this disk file. Use the MS-DOS 5.0 text editor to remove the control code junk from the file and resave. The resulting file will be in ASCII format and when you use PC TRANSFER to convert the file to a TI disk the file will print very nicely exactly as it was formatted (in columns for example). This latest article was submitted by Harold Hoyt.

ONCE AGAIN THE FAIR
by Dick Beery

Saturday, May 15, 1993

Lima, that is! Dave Szippl, Charlie Good and their crew once again pulled it off! At the 12:30 officers' meeting, Charlie announced that fair attendance at that point was already considerably above that of the preceding year. It keeps on going!

Since C.O.N.N.I. did not host a table offering software this year, those of us who attended had greater opportunities for visiting the many exhibits and sale tables and for listening to the speakers. I enjoyed Jack Sughrue's presentation on "The Teaching T.I.: Our Computer as an Educational Tool." His point was that the 4A software already available can provide families with an inexpensive and in many ways superior set of learning tools primary-through-high school and perhaps beyond. He mentioned many standbys such as Early Reading Fun and others, and demonstrated a variety of programs for mathematics created by Tony Falco. Disks for these latter were on sale at his booth. Noting the vastly reduced prices for consoles using cassette and even for basic disk systems, he urged groups and individuals to identify those who could profit from such materials and help them find those suited to their needs.

I should have reminded him of the many fine educational programs developed by Jim Peterson, the Tigercub, in many areas of interest: mathematics, sprites, music, brain games, etc.

Times had been switched at the last minute, so that Barry Traver's seminar followed Jack's, rather than preceding it. This was announced over the speakers, but some people, including our own Jean Hall, did not hear the announcement, and were very disappointed at missing Jack's speech.

I did not attend Don O'Neil's talk, so cannot comment upon it. Harry Brashear was to have spoken at eleven on "FIRST DRAFT, XB3, memory expansion hardware and other products from Asgard". Instead, the developer of FIRST DRAFT himself, Art Gibson, took the podium. He focused entirely on FIRST DRAFT and almost exclusively on the editor. This is a bright and promising alternative to TI-Writer and its many clones and revisions (FW!). Some of the many features are: multiple columns, a 50-line keyboard buffer, a large and seemingly very capable dictionary/spell checker (yes, I know, you're a good speller, so am I, but had you ever thought about using it to catch TYPOS?), works well with a hard drive, file length up to 400k(basically as much as a double density disk will hold). Some caveats: you cannot use your old DV80 files with the Editor, without first using the conversion program (provided) to change them to the DF80 format used by FD. However, you can print your DV80 files directly from the Formatter, which is called FINAL COPY. Haven't had the time I want to play with this program yet, but I believe it will not be disappointing and will offer many features that until now have not been available to us. P.S. I forgot to mention the pull-down menus! AND the fact that from the Editor you can now format a new disk. Ever try to save a file to disk with insufficient space, only to discover that you don't have any more formatted disks? Try this/these program(s). You'll like it/them.

The officers' meeting was interesting, but did not really break any new ground.

Mike Maksimik's focus was on the streamer tape drive and program. When questioned about the release of the "ultimate" version (3.0) of his popular MidiMaster99, he could only say that it was not yet ready for release, but he did reiterate that he has been unable to produce a satisfactory version for the unassisted 4A. Users will need to purchase a Rambo or other such memory device to access version 3.0.

I understand that interesting seminars were given by Bruce Harrison (programming tools); Tim Bodenmiller (a promising young star on the programming scene)re: Game programming; one by Bud Mills, presumably on 4A MEMEX and related topics;

ONCE AGAIN THE FAIR

(continued)

one by Ken Gilliland (don't know the topic, but he seemed to have little that was new at the table); a lecture on Geneve software by Don Walden from Cecure Electronics; and one from S & T Software on "BBS and Terminal Emulation Software for the TI.

I am especially sorry to have missed the terminal emulation talk, but it conflicted timewise with Charlie Good's presentation on the Funnelweb v5 40 column editor, which I attended. Excellent presentation, as usual, by Charlie. He showed much of what is new in the Editor (which by the way was available from the Lima Users Group for one dollar a disk). I was particularly interested in learning how one could and could not access the foreign character sets.

Mike Wright's demonstration of the PC99 project, an emulator of the TI for use on a PC, went very well. Admittedly it is still in the development process, but I found it very interesting. Speed is still a problem, and it will probably require the use of at least a 40486 PC machine. They promise to code a Function/Shift/B key sequence for Fast Term diehards like me, if this is requested. (It uses, of course, a PC keyboard, which shares the limitations of the Rave keyboard in this respect.) They demonstrated both Funnelweb and TI Artist functioning on the PC and also a sprite simulator that was not at all bad. Colors may be changed and compared, and a utility has been provided that can show the contents of a T.I. file.

Some will ask: why would you want to run 4A programs on a PC? Not sure I can answer that at this time, but there appears to be a lot of interest. Maybe we should pay attention to this, as the developers appear to be riding a wave of the future. Similar emulations between unlike computers already exist for the Amiga and other machines, and you may have noticed that Apple MacIntosh reportedly now provides a new machine that runs traditional Mac software, plus PC MS-Dos programs, plus software for the old Apple II series. That was considered close to impossible, but they have apparently done it. So let's support this project and ride our own surfboards into the realm of the future.

Hats off once again to Dave, Charlie and all the others for a mammoth labor of love that provides us all with such rich opportunity and that is so well done. May not make you any less exhausted, guys, but you really ARE APPRECIATED.

RAVE KEYBOARD AND FAST TERM
by Jean Hall

Several months ago several members of the CONNI User Group purchased RAVE keyboards and wanted to know if they could still use Fast Term to log onto the local Spirit of 99 BBS. The answer from Chuck Grimes was a firm yes, with some minor key stroke changes.
Bob DeVilbiss and I looked in the manual and with some experimentation came up with the following key strokes and were able to log on to the BBS and do the things we had done just like we had done before we purchased our RAVE keyboards.

1-LOAD FAST TERM

2-DIAL PHONE NUMBER

3-LOG ON WITH YOUR NUMBER
  AND THEN GIVE YOUR
  PASSWORD

4-TO DOWNLOAD
  PRESS ALT N
  PRESS ENTER
  KEY IN DRIVE #
  KEY IN FILENAME

5-PRESS D

6-PRESS ZERO FOR CRC

7-PRESS CTRL F4

8-PRESS R TO RECEIVE

9-PRESS Y TO RECEIVE CRC

Other commands available:
CTRL 2 to continue with library listing
CTRL 6 to spool to printer
CTRL 2 dump to printer
CTRL 3 for parity
ALT D for half duplex to talk to another person via the modem.
Experiment and have fun with your new Rave keyboard and Fast Term.

DV80 TO PROGRAM CONVERSION

by Jim Peterson

John "Jeb" Hamilton of the Central Iowa User Group was the first to realize, several years ago, that a DV80 listing of a Basic or XBasic program could be converted to a DV163 file and then merged in and run as a program. I no longer have his program in my library, but this is a quick and dirty version of it -

```
100 DISPLAY AT(12,1)ERASE AL
L:"Input file? DSK";"":"Outp
ut file? DSK"
110 ACCEPT AT(12,16):A$ :: A
CCEPT AT(14,17):B$
120 OPEN #1:"DSK"&A$,INPUT :
: OPEN #2:"DSK"&B$,VARIABLE
163,OUTPUT :: LINPUT #1:M$
130 LINPUT #1:M$ :: IF LEN(M
$)>78 AND EOF(1)<>1 THEN LIN
PUT #1:M2$ :: M$=M$&M2$
140 X=POS(M$," ",1):: Y=VAL(
SEG$(M$,1,X-1))
150 PRINT #2:CHR$(INT(Y/256)
)&CHR$(Y-256*INT(Y/256))&"!"
&SEG$(M$,X+1,255)&CHR$(0)
160 IF EOF(1)<>1 THEN 130 EL
SE CLOSE #1 :: PRINT #2:CHR$
(255)&CHR$(255):: CLOSE #2
```

To try that out, key in this useless little program -

```
100 CALL CLEAR
110 FOR J=1 TO 10
120 PRINT J
130 NEXT J
140 END
```

List that to disk by LIST "DSK1.80". Then run the above converter program, answer the input prompt with 1.80 and the output prompt with 1.163. After it runs, enter NEW, then MERGE DSK1.163 and then LIST. This is what you should see

```
100 !CALL CLEAR
110 !FOR J=1 TO 10
120 !PRINT J
130 !NEXT J
140 !END
```

Type 100 and FCTN X to bring line 100 to the screen with the cursor on the "!". Type FCTN 1 to delete the "!" and repeat with FCTN X and FCTN 1 to delete all the others. Then enter RUN and it should do so!

All that the program does is delete the blank first line of the listing, convert each program line number to tokenized format, add a CHR$(0) end-of-line marker to each line, move the record to a DV163 file, and add the double CHR$(255) end-of-file marker.

The result is a merge format program composed of REM statements; when you delete the "!" REM indicator, these become program lines.

There is just one problem. A LISTed program is a DV80 file, consisting of records of 80 characters or less, but a program line in XBasic can be keyed in up to 140 characters long, and can be forced even longer (as I often do!) When such a line is LISTED, it is broken into 80-character records, which confuses the conversion program completely.

Line 130 of the conversion program attempts to resolve that problem. If a record is more than 78 characters long (because it could have been an 80-character line ending in a blank, which would become a 79-character record without the blank) it is taken to be most probably the first part of a long program line; another record is read in and tacked onto it.

However, this creates another problem, as you will find if you LIST the converter program and then try to convert it back to a program - line 140 will be tacked onto line 130 because line 130 is 79 characters long.

The best fix for this is to load the DV80 file into Funlweb and print out a hard copy; use a ruler to draw a vertical line after the 78th characters; mark any program line that ends on the 79th or 80th characters, delete those characters, save the listing, run it through the converter, merge it in and key those deleted characters back in - still much easier than keying in an entire listing.

After John Hamilton published his discovery, several authors wrote their own versions. It was suggested that programs could be written in text format, using the superior editing features of TI-Writer or Editor Assembler, and then converted to program format. Personally I was satisfied with the editing features of Basic and was not about to give up its syntax error-catching capability, so I never tried this method.

However, nowadays several hundred text files of newsletter articles are available on the Clearing House BBS, and other newsletters are being circulated on disk. Many of these articles contain program listings, and it would be much easier to extract and convert them than to print them out and key them in.

Later on, John Ford wrote a more complex converter called XLATE, which eliminates the need to delete all the "!" by converting the ASCII text file directly to tokenized merge format. It also checks for syntax errors and corrects them or reports them on-screen . If the LISTed program had regularly sequenced line numbers, it will check these to see whether records should be combined, which should greatly improve accuracy - I have not tested it enough to say how foolproof it is.

Blanks at the end of a DV record are dropped, so if the 80th character of a long program line is a blank, when the line is broken into two records and then recombined the blank will be missing. For instance, if the blank between FOR and J in FOR J=1 TO 10 happens to be the 80th character, it will recombine as FORJ=1 TO 10. This results in a SYNTAX ERROR referencing the line number, which is therefore easy to spot and correct. The same problem can cause the string "John Doe" to become "JohnDoe".

The above conversion programs are intended for listings in 80-column format. However, many of the listings within text articles have been reformatted to 28-column or 40-column width, or listed in those widths with Triton Super Extended Basic.

Fortunately, there is an alternative. Curtis Alan Provance has written a truly remarkable program in assembly, called TEXTLOADER, which will convert a DV80 file directly into a program in memory, and will handle the shorter line lengths, although with increased chance of error because the method of detecting new lines is far from foolproof.

I have not tested this program extensively, but have found only two major problems. The one is with records ending in a dropped blank, as described above; these are easy to correct. The other is with split referenced line numbers. For instance, if a line ends in GOSUB 120 :: GOTO 200 and splitting of records turns this into GOSUB 1 and 20 GOTO 200, you will find the line ending with GOSUB

1 and a new line 20 :: GOTO 200 at the beginning of the program. Comparison with the original listing makes this easy to correct.

TEXTLOADER loads into memory and remains there, so that you can load other text files by simply typing - CALL LINK("OLD"&"DSKn.filename"). The file loads and converts rapidly, displaying each line as it does so. Sometimes a line which has been corrupted will be reported as a syntax error and omitted, but sometimes it will be omitted without being reported, and sometimes it will not be detected until you try to run the program. Occasionally, especially when working with 28-character lines, you may get all sorts of invalid error messages. Apparently the program in memory differs from the screen display, and it may be impossible to debug in such cases.

Other features allow you to merge a converted text file into a program in memory rather than overwriting it, and to read and run a batch file of command type instructions, such as -
CALL FILES(1)
NEW
RUN "DSKn.bigprogram"

An improved general-purpose memory image program loader is also included.

XLATE is a public domain program, available on my TI-PD disk #1083. TEXTLOADER is a fairware program available on my TI-PD disk #1104.

F U N N E L W E B
VN 5.0/80
Appended solutions
DSHTI 3/93

Funnelweb 5.0 editor allows you to APPEND a DV/80 file onto the END of an existing file.. There IS A PROBLEM! If you saved the file the regular way, through Save File (SF), (call it DSKx.FILE) then the new lines saved PF and then a DSKx.FILE will be ADDED AFTER the TAB line. When you go to add the lines back in, my file stopped at the line with the TABS and would NOT load the rest. However, I could see everything using View the file, so I knew it was still there. Here are two solutions:

1. LOAD the file (LF) with the extensions '1 X DSKx.FILE' where the X is the line BEFORE the TAB. Then load the rest of the file to the END of the file in memory with "E X+2 Y DSKx.FILE" where X+2 is the line after tab and Y is the LAST LINE of the file that wouldn't load properly the first time. This worked and saved my bacon.

2. When you SAVE the file the first time, save it through PF using 'DSKx.FILE'. This will NOT save the TAB line so that the Append mode should work

At any rate, the TIWriter concept of MERGING lines from files into a file that you are editing proves its worth. Whoever designed the origional loader for TIWriter was a real thinker.

REPLACE STRING
in FW 5.0/80

Jim McLaren, from the Sunbury User Group phoned me regarding the use of REPLACESTRING on the latest version of FUNNELWEB EDITOR (5.0). He said that he couldn6t get the 3 numbers to work.

Most of us would use just REPLACESTRING without any numbers since we want the whole contents (all columns) to be searched. However, you can do a COLUMN search instead. For example, if you know that there will be numbers in the columns 1 to 5 then you search just those columns to make numeric changes. The ReplaceString (RS) would look like this:
1 5 /6/7/

This means, search for occurrences of the number between column 1 and 5 and replace it by 7.

However, if you only want the second and subsequent 6's replaced by 7's, then you would enter it this way:
1 1 5 /6/7/

The first '1' says to ignore the very first occurrence of 6 and go to the second and subsequent one.

I think Jim probably thought that the 3 number should have gone AFTER the columns and not BEFORE, which is the case. I hope that this clarifies it. It does work. I hope everyone remembers that when you use RS to make sure that you DO NOT HAVE WORDWRAP MODE turned on;: use the 'HOLLOW' cursor (ctrl 0) only.
Other wise everything will be reformatted. Some times you might want this, but most times NOT.

END

***** TI-101 *****

OUR 4/A UNIVERSITY

by Jack Sughrue
Box 459
E.Douglas MA 01516
originally appeared
BITYS,BITES & PIXELS
Lima, U.6.

#2 Holism
Happy New Century

Last time, Class, in our TI-101 classroom we introduced the historical perspective of public education in a few strong words. We stated that some of the wrongs with our schools today is the profiteering by the big book industry who would like all our children to be into some kind of large-scale, lock-stepping curriculum as devised by them. [Close to 100% of all the schools in America have curriculums established by publishers and screwed into place by administrative bureaucrats. They are not created by the teachers, the trained professionals who work directly with the children. Once in a while - such as the school in which a friend of mine teaches 3rd graders - a school is blessed with an intelligent, child--oriented principal who is not afraid to empower her teachers. But this scenario is truly rare in our country.]

Which brings me back to THE REVOLUTION in education I discussed during our last class. This is the revolution of holism in education. It is an international grass roots approach to learning. Though the spelling is H O L I S M, the meaning does not come from "holy" but from "whole." Why don't THEY spell it "wholism?" you may ask. Unfortunately, you may ask in vain.

But Whole Language is what is under discussion here. Whole Language is the most prominent movement in the revolution. It is a philosophy that asks how children learn and then seeks ways to provide those opportunities for the child. It is, in short, a research-based philosophy and an intellectual attitude and a creative style that considers the developmental learning stages of the children as a group and each child individually. But what is it, specifically?

Well, let's look at product results first, Class, although Whole Language Educators will be the first to say that process rather than product is the goal of W.L.:

In the standard achievement tests scores given world-wide the U.S. ranks 47th. On those same tests New Zealand is 1st. New Zealand has close to 100% of its teachers, K-12, using W.L. New Zealand has the highest rate of literacy of any country in the English-speaking world.

Now back to how W.L. works and what it is. In the U.S. we have had a long history of process methodology. Unfortunately, it has never been a part of mainstream education. Like jazz, as musically intricate as any form of music on the planet, has never become the mainstream of American culture. But there were many educators who understood how children think and how children learn. These people have taught and have written books and have done research. But, except for the unusual teacher or an extremely rare school staff, few people had access to these ideas and materials and methodologies. Such things as the Teacher-Writer Collaborative in New York, the Bay Area Project in California, and the Framingham Writing Project in Massachusetts spread the word through research, printed materials, workshops, teacher training programs, sweat, blood, and tears. But these were a few of the isolated programs and projects and groups that sought to integrate the curriculum by starting at Square One and helping the students learn from their own strengths in a positive "unending" environment which tied various aspects of learning into complex, relevant activities: thinking on a large scale, understanding analogies, making connections, discovering solutions.

To explain another way, Class:

Most of us grew up learning little isolated skills. We learned to Captitalize on the 9th week of school, let's say, in the 8th Grade.

Following that week, during which we'd be forced to learn the 60-odd capitalization rules for Friday's test, we'd leap into a couple days of hyphens and dashes, before going on to colons and semi-colons, and so on.

Isolated. Irrelevant. Boring. And not a good learning environment. We learned for the immediate tests and could not apply these "learned" skills to our daily (and real world) writings.

But such isolated, "testable" skills are a publisher's dream and an administrator's idea of Heaven. Because the kids can be tested on each of these isolated pieces, numbers can be attached to their names. These numbers can then be sorted into descending order and grades issued based on this garbage.

This has nothing to do with learning, with life-long skills, with internalizing and ownership. This has to do with outside forces trying to jam 19th Century methods down the throats of the people who will be running the 21st Century.

Bad stuff.

Take almost any English book you can get your hands on, and you will not find any writing activities (or few except in the most recent books and then as a way to thwart the movement away from texts). The books tell, tell, tell, tell how YOU are supposed to know this rule and that. The books test, test, test. They introduce the English materials in the most inane ways. For the most part, traditional English text books are sappy, to say the least, and anti-education to be really honest. And, except in a splashy, surface way haven't really changed since McGuffey's Readers of a century ago.

At the time of the Industrial Revolution the sum of human knowledge doubled about every 150 years; at the turn of this century it doubled about every 75 years; after World War II every 25 years; in 1990 every 9 months!

We still need to teach our kids skills, but we need to teach them DIFFERENT skills, better skills, more relevant skills, as "coverage" is impossible. [By the time a science book is researched and written and edited and printed and sold and distributed and finally used in a classroom it is already quite a few years out of date. And this is not just for info about our Solar System, for example, since the Voyager trips; it is about dinosaurs, which we know more about today than we did last year. Information progresses at a quantum rate, and this is true in every area of our real as well as academic lives.] Coverage is impossible, Class. Remember that. It's

going to be on your next test.

We need to teach our kids HOW to think. Informational regurgation is no longer relevant as we swing into the 21st Century. We need to teach our kids HOW to think, so they can be prepared for the future. And no matter how much we may long for the good ol' simple days of yore, they just ain't a'comin' back. We are - for better or worse - in the Electronic Age. And our kids, if they are going to compete with the rest of the world or if they are just simply going to keep America great, have got to become thinkers. They've got to become thinkers who can use the tools of the future NOW.

Einstein (Albert) was asked for his phone number by a reporter. He looked it up in the phone book, astounding the reporter. Einstein explained that it would be foolish to clutter up his brain with anything that could be looked up.

If Einstein felt he should not be cluttering up his brain with useless information, maybe we could all take heed.

Let's give our kids and everyone else's kids a headstart for the next century by supporting our overworked teachers (instead of bashing them) and joining forces with them to provide a new environment in schools and in our homes. Let's advocate FOR our kids and their teachers. On 60-MINUTES, recently, Andy Rooney said the real problem with education today is not the teachers and not the schools but that "there are too many dumb kids," and, worse, too many dumb parents who don't prize education, who don't value learning (thus, too many dumb kids). I believe, truly, that we can get rid of this dumbness (which Steve Allen calls "DUMBTH" in a wonderful book by that name about the state of American thinking) by turning off the electronic babysitters (TVs and Nintendos) and get the kids into electronic tutors (computers) and maybe even (gasp!) books!

And here we are at the point of these classes: our TIs and what they can do to reverse this terrible dumbing trend in our country.

We'll take this up in our next class by introducing you to some of our brave TI-World educational experts and what they have offered and how we can use their gifts.

Your homework is to dust off all your your educational cartridges (which includes TI-WRITER, of course, as well as TERMINAL EMULATOR and MINI-MEMORY (think about it), as well as DRAGON MIX, READING RALLY, SCHOLASTIC SPELLING, and BEGINNING GRAMMAR). You don't have to pass in any papers next session, but you must be prepared to present a 10-minute talk on at least two of your selected cartridges, being prepared to defend its educational relevance to the child of the future.

Be early for TI-101 next time and get a good seat up front. Adios.

END PART 2

*** CORRECTION ***

In the May issue of our newsletter there is an error in the article CHECKSUM on page 12. In the second column we were talking about steps required to insert the checksum numbers. The error is in the first step. The EX: SAVE "DSK1.A" <enter> (or any filename). Change this to read SAVE "DSK1.A",MERGE <enter>.

PRINT USING
by Mark Schafer
reprinted from
TIsHUG NEWS DIGEST
11/92

This is a new and different animal for me, as I normally do not write tutorials. It came up at a recent meeting that some people are having trouble with PRINT USING. Since I consider it to be no problem, I volunteered to write an article about it. As for the title, remember that USING can also be used in a DISPLAY USING statement. There is a lot of ground to cover, so let's get started with....

Beginner's Stuff

USING represents a method by which you can get data to print in a specific format instead of the default format. The syntaxes are:

```
PRINT USING format:print-list
DISPLAY [option-list:] USING
format[:print-list],
```

where format is a line number or a string expression. A line number would be the line where an IMAGE statement is found which would contain the string expression which is the format. A string expression can be as simple as a string literal enclosed in quotes or a complicated expression made up variables, function calls, whatever. Now let's compare the two methods of printing:

```
100 PRINT -56.7;109.2850
110 PRINT USING "####.# ###.
####":-56.7,109.285
```

RUN this program and it looks like this:

```
-56.7 109.285
-57.7 108.2850
```

USING allows you to better control the spacing as well as put trailing zeroes to the right of the decimal point.

Let's look at the format string. It is made up of fields with optional text. Fields are like fill-in-the-blanks. They mark the place where an unknown value will be printed. Text is printed the same way every time. Fields are made up of pound signs (#) with possibly a decimal point, a minus sign, or maybe some circumflexes (^). I will cover circumflexes in a later section.

Pound signs take the place of a digit or sign. So in line 110 above, I printed -56.7 with the field "####.#". -56.7 has only three characters to the left of the decimal point, so the initial character is left blank; the second one is where the minus sign went. Numbers are always aligned with the decimal point. If there is none, it is assumed to be at the end.

In order for me to get the numbers to be aligned with the decimal point, I had to change the field "####.#" to "###.# in line 110 . This is contrary to the statement in the above paragraph...(SPIRIT of 99 ..ED

If there is a minus sign in the field, it always goes at the beginning.
It indicates that you want the minus sign to appear immediately to the left of the number if it is negative. However, this is the same thing another pound sign would do, so you never really need to use a minus sign (except for a more advanced purpose to be discused later.) In other words, "-##.##" does the same thing as "###.##".

The decimal point indicates where you want it to be. The number of pound signs you put to the left of it dictates how many digits you want to the left; the number you put on the right indicated how many decimal places you want. The decimal is always printed even if there are no pound signs after it.

If there are fewer digits in the value to the left of the decimal point than there are in the field, spaces are used to fill it out. If there are too many, the computer will refuse to print your value and fill the field with asterisks (*). This means your value is too high and/or there is not enough places in your field. This includes the minus sign, if any.

If there are fewer digits in the value to the right of the decimal point than there are n the field, zeros are used to fill in the remainder. If there are too many, the computer will estimate the number to the number of places given. So.## can be used to estimate to the nearest 100th, .# to the nearest tenth, and if there are no decimal places, it will estimate to the nearest unit.

Below is a table of how various values will be printed with various fields. The left side represents the value; across the top are the fields.

```
          # ## ### #.# ##.## ###.###
2         2  2   2 2.0  2.00   2.000
-13       *  ** -13 *** ***** -13.000
9.671     *  10  10 9.7  9.67   9.671
125.678   *  ** 126 *** ***** 125.678
-.385     *  -0  -0 -.4  -.39   -.385
-3.05     *  -3  -3 *** -3.05  -3.050
```

Generally, Extended Basic handles format strings like this: It keeps printing text until it comes to a field (characterized by a pound sign.) Then it looks for the next value to be printed. If there is none, it terminates there and does not print anything else. However, you must specify at least one value and there must be at least one field in the format. Except, curiously,

DISPLAY USING

does not need any values, Why anyone would use DISPLAY USING

without any values is beyond me. Anyway, if it reaches the end of the string, and there is more values to be printed, it goes to the next line and starts over at the beginning of the string. So it is ok if the number of fields does not match the number of values.

Going back to syntax, if you want to use the format "I HAVE $###.##", you have three ways of doing it:

```
120 IMAGE I HAVE $###.##
130 PRINT USING 120:M

140 A$="I HAVE $###.##"
150 PRINT USING A$:M

160 PRINT USING "I HAVE $###
.##:M
```

Notice that if you user the first method using IMAGE, you do not need quotes. The only time you need quotes with an IMAGE statement would be when you have leading or trailing spaces. If you are going to use it repeatedly in a program, Image is the most efficient method. If you are only using it once, go with the third method. Practically the only time you need the second method would be when the format string is constructed so you may not know exactly what it looks like.

The IMAGE statement must be on a line by itself. The computer ignores it when it comes to it in a program the same as it does the DATA statement, so you can put it anywhere. Circumflexes are used to denote scientific notation which leads us to the next section....

Intermediate Stuff

You may want the number to come out in scientific notation (E format) even if the number normally would not. To do this you put four or five circumglexes at the end of the field. Four means you want two digits in the exponent; five means you want three. If you put less than four, they will be treated as text; if you put more than five, the first five will be used, and the rest will be treated as text.

There is a little something different about E format. It always reserves the first character for the sign, so you will need at least two #'s in the mantissa)or precede the field with a sign). Using the same numbers as above, the table can be amended thusly:

```
##     ####     #.####
2 2E+00 200E-02 .2000E+001
-13 -1E+01 -130E-01 -.1300E+002
9.671 1E+01 967E-02 .9671E+001
125.678 1E+02 126E+00 .1257E+003
-.385 -4E-01 -385E-03 -.3850E+000
-3.05 -3E+00 -305E-02 -.3050E+001
```

This format also estimates whenever it is not given enough places to express the exact value or tacks on zeroes when given too many. You will probably not need this format unless you are dealing with exceedingly low or high numbers.

Time to switch to another concept. USING can also be used to format text values. No bells or whistles here, though. Any field that can be used to format a number can also be used to format text. Text is just left-justified, and all characters within a field are treated the same. I can give you a table, but I assure you, it is quite boring:

```
### ###.### -###
BOB BOB BOB BOB
JOHN ### JOHN JOHN
25% OFF! ### ####### 25% OFF!
```

In general, it makes more sense to use only #'s when constructing a text field. You may want to do something like column 2 if you are printing a table of numbers, and you want to use the same format string for the column headers as in the table.

Of course you can mix numeric and text values for the same format as in something like:

```
170 IMAGE ##### HAS $###.##.
180 PRINT MARY HAS $123.40
```

This will print $123.40

Note the computer recognizes the period at the end as a period and not as a decimal point since the field already has one. Even if it did not, it would not make any difference since the decimal point would be printed at the end.

Now suppose you want part of a line formatted instead of a whole line. Do not worry; you can always put a semicolon at the end of a PRINT or PRINT USING statement, so you can stay on the same line for the next PRINT. You can also use this technique if you need to print a # as a text instead of a field character. However, you cannot put a comma at the end of a PRINT USING statement or the computer will think you left out a value.

Another problem you may have is suppose you want to concatenate two fields, say ## with ###. If you put them together, you get #####, and the computer will see that as one field. Well, you could try using "##-##". The computer will then recognize that as two fields, ## and -##. But if your second field can have three digits, this will not work. If this is the case, you will need to split up the format string into two PRINT USING statements. Perhaps like this:

```
190 PRINT USING "##":A;
200 PRINT USING "###":B
```

You cannot confuse the computer with pound signs, minus signs, circumflexes, decimal points, etc. It will always know where one field ends and another begins.

I will give you another way to handle that in....

Advanced Stuff

There is not very much in the way of advanced stuff because this is such a small subset of a much bigger entity. I always consider undocumented features to be advanced stuff. That is right, boys and girls, USING has an undocumented feature! By "etc." in the above paragraph, I meant the plus sign! It can also be used like the minus sign in a field. What it will do is float the sign in front of the number be it positive or negative. Let me illustrate.

```
+# +### +###.##
2 +2 +2 +200.00E-02
-13 ## -13 -130.00E-01
9.671 ## +10 +967.10E-02
125.678 ## +126 +125.68E+00
-.385 -0 -0 -385.00E-03
-3.05 -3 -3 -305.00E-02
```

If it is a negative, you get a minus sign: if it is positive or zero, you get a plus sign. There is no wordabout this in the Extended Basic manual.. This can be used to write format strings for equations, like this:

```
200 IMAGE ##x  +##x+##
210 PRINT USING 200:2,7,10
220 PRINT USING 200:-15,-11,1
230 PRINT USING 200:+9,33,-14
```

This will print:

```
2x 2 +7x+10
-15x 2-11x +1
9x 2+33x-14
```

Suppose you do not want it to print those leading spaces when there are no enough digits to fill the field. This is where you would use a variable for the format string. You would construct it so that each field would only have as many #'s as it needed to print the number (since they are all integers, that can easily be accomplished with LEN(STR$(X)) and concatenate the necessary text and use the variable as the format string. For instance, I have a program that uses a define function like this:

```
240 DEF MF$(X)=" $"&RPT$("#"
,LEN(STR$(INT(X))))".##"
```

That function generates a format string for money so that there will be no spaces between the dollar sign and the amount.. The reason I did it this way was to get trailing zeroes after the decimal point to look good. However, user-defined function calls take a long time in Extended Basic, so if you need speed or if you only need it once, put a formula like this directly where you need it.

There is another manual correction that must be made (although it may have been made already in some addendum I do not know about). The syntax for PRINT USING with files is wrong. I will not reprint it here because I do not like glorifying the incorrect. The correct syntax is as follows:

```
PRINT[#file-number,[REC                record-number,]USING
format:print-list
```

That translates specifically as something like:

```
250 PRINT #1,USING 200:A,B,C
260 PRINT #2,REC 15,USING 20
0:A$4,B$4,C$4
```

The file being referenced is the most likely the printer, but it could be a disk file. If it is, it must be a DISPLAY format file; you will get a FILE ERROR if is INTERNAL format. Obviously line 260 is not referring to the printer.

PRINT USING is very good for printing to a printer because the printer has so many more columns to print in. So there is a good chance that you may have a program in which a particular PRINT USING is only being used to print to the printer and never to the screen. If this is the case there is another solution to the concatenated field problem mentioned in the last section. You can print some unprint-two fields but they will be together on the print-out! Something like this would be typical:

```
270 A$/"##"&CHR$(0)&"###"
280 PRINT #1,USING $:A,B
```

CHR$(0) does not do anything on most printers, so the two fields will appear consecutive. Note that you could not do that in n IMAGE statement since you can only use characters and not expressions. The expression could have also been constructed within line 280 itself.

This should about exhaust the subject of USING except to remind you that you can also use DISPLAY USING if you need to format values somewhere else on the screen (by using the AT option) or if want to BEEP or clear the screen before doing it (REASE ALL).

END

MEMO PAD

by Bob August
from BUG News

This program is called MEMO PAD and is a daily reminder. You enter the date and a 28 character line memo. You can read the memos by pressing the N key for the next and then B key for back. You can also delete a memo by pressing the D key. The program is in Extended Basic and is dimensioned in line 130 for 100 memos. If you need more or less than 100, change both of the 100's to what ever you need. Notice that Chesum characters have been added to this program.

```
100 ! MEMO PAD IN TI !058
110 ! EXTENDED BASIC !118
120 ! BY R.W. AUGUST !092
130 DIM DATE$(100),MEMO$(100
):: N=0 !234
140 DISPLAY AT(3,7)ERASE ALL
:"<<< MEMO PAD >>>" !230
150 DISPLAY AT(7,1):"PRESS:"
: : :TAB(5);"(1) to Enter ne
w Memo": :TAB(5);"(2) to Dis
play/Del Memo" !233
160 DISPLAY AT(14,5):"(3) to
 Save Memos": :TAB(5);"(4) t
o Recall Memos": :TAB(5);"(5
) to Quit": :TAB(9);"Your ch
oice (1-5)" !036
170 CALL KEY(0,K,S):: IF S=0
 THEN 170 :: IF K<49 OR K>53
 THEN 170 :: CALL CLEAR :: O
N K-48 GOSUB 1000,2000,3000,
4000,5000 :: GOTO 140 !110
1000 ! Enter memo section !1
32
1010 DISPLAY AT(3,7):"<<< ME
MO PAD >>>": : : :"Enter 0 t
o quit.":"Use Period(.) for
Commaa(,)." !095
1020 N=N+1 :: DISPLAY AT(10,
1):"Memo #";STR$(N): :"Enter
 Date:": : : :"Enter Memo:":
 : :"" !143
1030 ACCEPT AT(13,1)SIZE(-28
):DATE$(N):: IF DATE$(N)="0"
 THEN DATE$(N)="End of List"
 :: N=N-1 :: RETURN !060
1040 ACCEPT AT(17,1)SIZE(-28
):MEMO$(N):: DISPLAY AT(20,1
):"Is the above correct Y/N"
 :: ACCEPT AT(20,26)SIZE(-1)
VALIDATE("YyNn"):YN$ !028
1050 IF YN$="Y" OR YN$="y" T
HEN DISPLAY AT(20,1):" " ::
GOTO 1020 !167
1060 IF YN$="N" OR YN$="n" T
HEN DISPLAY AT(20,1):" " ::
GOTO 1030 !156
2000 ! Displat/Delete Memo !
164
2010 DISPLAY AT(3,7):"<<< ME
MO PAD >>>": : :"Enter # to
Display/Delete": :"1 to";N :
: ACCEPT AT(10,1)VALIDATE(DI
GIT):EN !040
2020 DISPLAY AT(10,1):"#";ST
R$(EN):"Date:": :DATE$(EN):
:"Memo:": :MEMO$(EN)!1652030
 DISPLAY AT(23,1):"(N)ext (B
)ack (D)elete (Q)UIT" !098
2040 CALL KEY(0,K,S):: IF S=
0 THEN 2040 :: IF K=68 OR K=
100 THEN 2080 !074
2050 IF K=78 OR K=110 THEN E
N=EN+1 :: GOTO 2020 !241
2060 IF K=66 OR K=98 THEN EN
=EN-1 :: GOTO 2020 !205
2070 IF K<>81 AND K<>113 THE
N 2040 :: RETURN !232
2080 FOR D=EN TO N-1 :: DATE
$(D)=DATE$(D+1):: MEMO$(D)=M
EMO$(D+1):: NEXT D :: N=N-1
:: DATE$(N+1)="End of List"
:: MEMO$(N+11)=" " !102
2090 DISPLAT AT(23,1):"Memo
IS dELETED" :: FOR D=1 TO 20
0 :: NEXT D :: DISPLAY AT(8,
5)SIZE(3);N :: GOTO 2020 !01
5
3000 ! Save memos section !1
36
3010 DISPLAY AT(3,7):"<<< ME
MO PAD >>>": : : :"Save memo
s section." !204
3020 DISPLAY AT(12,1):"Enter
 Drive Number and File": :"n
ame: DSK1.MEMO/DATA" !252
3030 ACCEPT AT(14,7)SIZE(-15
):FN$ :: OPEN #1:FN$,OUTPUT,
VARIABLE 56 :: PRINT #1:N !1
88
3040 FOR I=1 TO N :: PRINT #
1:DATE$(I):MEMO$(I):: NEXT I
 :: CLOSE #1 :: RETURN !143
4000 ! Recall memos section
!076
4010 DISPLAY AT(3,7):"<<< ME
MO PAD >>>": : : :"Recall me
mos section." !146
4020 DISPLAY AT(12,1):"Enter
 Drive number and File": :"n
ame: DSK1.MEMO/DATA" !028
4030 ACCEPT AT(14,7)SIZE(-15
):FN$ :: OPEN #1:FN$,INPUT ,
VARIABLE 56 :: INPUT #1:N !0
77
4040 FOR I=1 TO N :: INPUT #
1:DATE$(I),MEMO$(I):: NEXT I
 :: CLOSE #1 :: RETURN !131
5000 ! Exit program  section
 !138
5010 DISPLAY AT(12,1):"HAVE
YOU SAVED YOUR MEMOS": :"Yes
 or No" :: FOR D=1 TO 50 ::
NEXT D !225
5020 CALL KEY(0,K,S):: IF S=
0 THEN 5020 :: IF K<>89 AND
K<>121 THEN RETURN ELSE CALL
 CLEAR :: END !121
```

MANAGING YOUR MONEY - 5

reprinted from LA 99ers Topics

Author's Note: This series of articles dealing with personal finance based solely upon my own opinions and my own experiences. I am not a trained financial advisor nor am I an investment counsellor. You are cautioned to always seek the advice of a finance professional before making any decision or taking any action that would involve what to you is a significant amount of money.

SAVINGS AND INVESTMENTS:

mutual funds are perhaps the fastest growing vehicle today. The reason for their popularity is the fact that they allow individual investors to put relatively small amounts of money into a pool of funds that is professionally managed, with little or no fees or commissions, compared to the alternative of investing through individual investment advisors or brokers. While *I am not going to go into the intracacies and* strategies of mutual fund investing, I will explain how one can get started with this comparitively new investment vehicle., and how one owns shares or units in a fund can determine the rate of return on their investment.

There are actually several hundred mutual funds to choose from today and they come in various "shapes and sizes", meaning their ranks span the risk scale from low-risk *conservatively managed funds*, to *high-risk, very aggressive* funds, and the investment goals scale from Money Market Funds to quicker yield sector funds. Which one(s) you decide to put your money into is usually based upon the level of risk you are willing to accept, your investment goals are the "track record" of the fund(s) you are considering.

RISK LEVEL:

The amount of risk you are willing to take is something you should determine before diving into mutual funds or any other investment vehicle. To determine your risk level you should consider:

1. How long you are willing (or able) to leave the investment tied up.

2. How devastating the total loss of your investment would be to your personal finances and to you and your family?

3. Whether you are investing for long-term growth or whether you need the investment return(s) for income?

4. How much money you can invest.

5. How much time (or willingness) you have to devote to managing your portfolio.

INVESTMENT GOALS:

Your investment goals play a big part in how much money you will gain or lose on your investments and most often how much quickly the gains or losses will occur. The theory being, if you are willing to take big risk (at losing money) then you should realize big returns for taking that risk. Thus, if you are a risk taker, your investment portfolio would be heavily weighted towards Growth and Income type mutual funds. This is becsuse these funds are geared towards maximum profit in minimum time. Your investment goals are what you hope to accomplish by seeking investing your money with a mutual fund rather than sticking with the passbook savings or Christmas Club account you may have used in the past.

CHOOSING A FUND(S):

If you have thee resources and the patience, you can track a particular funds' performance by reading the Stock Markes pages in your Sunday newspaper. Sunday papers are used because they usually have the most complete and accurate quotes for the previous week. Since the Market closes at 5pm on Friday, there is a whole day for the newspaper folks to get the information.

Tracking performance is a demanding task, but one that many investors use. It also has the disadvantage of not being able to tell you now, what the fund has done in the past. Past performance information can be had from the Fund's prospectus (if you want to trust it) or one of the "money" type magazines.

Another method, which is a little more costly, but probably *less risky, and certainly less demanding, is to subscribe to a* "reputable" Mutual Fund Investment Newsletter. How you determine what "reasonable" is, I will leave to you.

These newsletters are usually written by an "expert" or group of "experts" in the field of Mutual Fund Investing. Most often they employ a forecasting model that uses virtually the same method for determining Fund recommendations that you would use if you read the Sunday newspapers for a consecutive *time period, past history (performance). However, they also* consider many other variables that you and I might overlook or do not have access to, such as economic indicators, political climate, world prices etc.

Most newsletters create model portfolios that you follow in your investments. The models are based upon high, moderate and conservative investment risks. The higher the risk the higher the return. In investment jargon, high risk funds would be called "aggressive funds". That means that the fund manager(s) usually look for common stock investments in "un-favored" companies that they believe are going to blosson.

The newsletter is also a good place to begin your adventure into Mutual Funds since they list the names and often the descriptions of various funds that you might consider. Many newsletters also list the toll-free telephone numbers of the funds so that you may call and order a prospectus.

STARTING OUT:

Assuming that you have some method of determining which

funds to invest in, the first thing to do is find out how to get in contact with the fund administrator(s). Virtually all mutual funds have toll-free numbers for use in requesting a prospectus.

The obvious place to look for these numbers is NOT in the yellow pages as one might think, but rather in the trade publications or in the newsletter of a friend who is already involved with Mutual Fund investing. Once you have the telephone number and are ready to make the call, be prepared to request information on specific funds (by the Fund's name). Most Mutual Funds are "Families" of funds. This means that there are several different funds under one administrator. This is necessary to meet the varying demands of the investing public and to optimize the use of investor dollars by having a broad range of investment types. Not everyone has the same investment goals, nor is everyone willing or able to assume the same level of investment risk.

If you don't know the name of the fund you are interested in, since newspapers only provide cryptic abbreviations for them when they are published, ask for help from the person you talk to on the toll-free line. Don't be afraid to be a little humble and admit your novice status. A good Mutual Fund employee will ask you what your investment goals are or perhaps ask whether you would categorize yourself as a conservative, moderate or aggressive investor. Either way, they are trying to determine what type of fund you might be interested in, based upon your investment goals.

Once you receive the Fund's Prospectus in the mail (it's free), you can try to read it from cover-to-cover if you want to. Some are quite easy to read, some are like an accountant's nightmare and others are in-between the two extremes. No matter which type you receive, all will contain certain information that you should look for, since it directly affects you and the money you will be investing. The list that I am going to give you is based upon my concerns. Yours may be different or more demanding, so don't feel that I have listed ALL the questions you should be asking. However, at a minimum, I think you should look for the answers to the following questions:

1. How much (if anything) does the fund charge to join (the entry fee) and do they charge you to buy shares or units other words, is it a LOAD or NO-LOAD fund?

2. Is there a minimum time you must keep your money in the fund without paying a withdrawal penality (an exit fee)?

3. What are the administrative cost? No fund operates for free. You pay someone to manage your money for you.

4. What are the redemption procedures and what kind of a hassle is it to sell out?

5. Do they require a minimum dollar amount investment?

6. Do they offer telephone redemption and/or telephone switching between funds in the same family? How many year?

7. What kind of reporting procedure do they have for shareholders and how often is it provided?

While all funds assess a management fee, many today do not require a sign-up fee. In other words, they are NO-LOAD funds. Some funds however have a "hidden" load, one that is assessed when you try to sell your shares. This is an END-LOAD fund, as opposed to a FRONT-LOAD fund. Front-Load funds charge you to get in. End-Load funds charge you to get out, especially if you do so in the first 1-5 years. Read the Prospectus carefully to determine which type you are considering investing with.

A very important consideration in your decision-making process is the amount or percentage of the fund's administrative fees. Most funds today, whether they are load or non-load types, assess a "hidden" cost to you, in addition to regular administrative costs, that the fund uses to pay for things like their advertising and marketing. You pay for this because the law, under Securities and Exchange Commission regulation 12b-1 allows it. There isn't much you can do about it, but you can mitigate it by choosing the fund that has the lowest administrative cost otherwise.

An important part of the investment game is knowing when to dump your shares, when to sell out. Because you don't want to lose money, timing is critical. Thus, the fund that you are investigating should have telephone switching or roll-over privileges. This means that it should offer (and support) the ability to sell your shares in one fund, and either redeem them or roll them over into another fund, by placing a telephone call. If the fund provides for this only by mail, you must realize that you will still be on a sinking ship for the number of days it takes your letter to get to the Fund Administrator's office. Normally, you should be able to make a phone call, communicate your "sell or roll-over" order and then you are usually required to follow that up with a letter confirming the order. The important thing is for you to realize the inpact on your money because of the difference in procedures. Time is truly money in these cases. This is the end of the five part series of "MANAGING YOUR MONEY"

HARRISON WORD PROCESSOR
By Earl Raguse

(Thanks ROM UG)

I bought the Harrison Word Processor at the Chicago Faire. I had heard good reports about it, and it was only $10. I immediately printed the 40 page docs, and proceeded to feel intimidated. Hence, I never did actually load the program. Well, I finally did.

I was immediately ticked off, when it made me name a file even before I had written one. Then to make matters worse, I was told to put the text file disk in drive #1, where the WP disk was, I positively hate authors who tell me where I should put my disks. Why can't they ask me where I want to put them.? When I have more than one disk drive, I do not want to swap disks, that's why I have multiple drives. I wouldn't mind if this sort of thing were easily changed, but this program is written in Assembler.

The program saves files in DF80 format, and thus if you want DV80, you must swap disks again, load the WP menu again, and select Utilities which permits one to convert to ASCII (ie DV80). This time it asks you where you want the output file, hooray, no swapping disks, until it can't find your file because the WP disk is in the drive, darn, I have three disk drives, I wish it would let me use them. Once you overcome this however, the conversion works easily, but is an extra nuisance step that I can do without.

I never did figure out how to right justify text. I sure it must be able to do that, how else could it call itself a word processor? I always like to work in 40 column wide right justified format, because I like to use multicolumn (2 or 3) text for easier reading. This WP seems to like 80 column, and keeps track of where you are on an 80 column basis. Hence, you have to make a mental conversion. ie line 7 Pos 41 is line 8 character 1 etc. I do not know how one would do that if you were working in 28 or 34 column text.

The program is furnished with a prompt strip to put on your computer above the keyboard, but it does not use the keys I have learned for TIW/FW. It also seems to have incorporated more functions into the number keys. Like for instance FCTN 9 is Move Text, and CTRL 9 is Copy Text. I do not see that as much improvement over TIW/FW, even though it may save one keystroke. CRTL 4 is FIND TEXT, which I presume is the same as FS in TIW, and CTRL 5 is SRCH/REPLACE which is probably RS.

This word processor has Page control, that is one can move around by page number. Big Word Processors like Word Perfect do that, but I am not impressed. It might be nice if one wrote documents of more than one page all in one file. I never do that, I like to make each page its own file.

I would imagine that if one habitually worked in 80 column text, did not worry about right justification, wrote long files of many pages, and did not send text files to others on disk, this might be an acceptable Word Processor, but if you are like me, I say, AVOID it. Admittedly, I spent less than an hour with this program, so probably have not found the Good Features, but I was so ticked off by the drive swapping that I gave up on it.

Incidently, Bruce Harrison's policy is complete, no questions asked, refund if you are not completely happy with his programs. I think I might take him up on that, but on the other hand I liked his music disks, so well that I may just forego it.

FATHER'S DAY
By Earl Raguse

Father's Day is June 16th, don't forget. Here are some things a Dad is good for:

A dad is good for putting worms on a hook.

A dad is good for explaining to your mom why you can't be clean all the time.

A dad is good for carrying you when you are tired, and your mom won't stop shopping.

A dad is good for helping you with your homework for about two years after your mom gives up.

A dad is good for fixing your wagon, your bike, and your car racing set.

A dad is good for teaching you how to play catch.

A dad is good for letting you run the power mower, while your mother sits on the porch and prays.

A dad is good for explaining the meaning of words you are too embarassed to ask your mom about.

MEETING DATES FOR 1993

3RD SATURDAY

19 JUN 1993
17 JUL 1993
21 AUG 1993
18 SEP 1993
16 OCT 1993
20 NOV 1993
18 DEC 1993
15 JAN 1994

4TH WEDNESDAY

23 JUN 1993
28 JUL 1993
25 AUG 1993
22 SEP 1993

C.O.N.N.I. BOARD MEMBERS

Pres. - John Parkins	614/891-4965
Vice Pres. - Chuck Grimes	614/268-8821
Treas - Bill Sheppard	614/881-5742
Sec/Sat - Jim Peterson	614/235-3545
Sec/Wed - Dick Beery	614/459-3597
Membership - Everett Wade	614/262-6346
Librarian - Ken Marshall	614/875-1670
Disk - Dick Beery	614/459-3597
Cassette - Everett Wade	614/262-6346
Cartridge - Ken Marshall	614/875-1670
NL Exchange - Jean Hall	614-885-4223
TIABS BBS	614/851-0708
Spirit of 99 BBS	614/263-3412
Irwin Hott	614/263-5319
Dick Beery	614/459-3597
Co-Editors/Spirit of 99 NL	
Jean Hall	614/885-4223
Bob DeVilbiss	614/892-0566

C.O.N.N.I.
181 HEISCHMAN AVE
WORTHINGTON, OH 43085