

THE FORT'S USER GROUP

```

*****
| President           Secretary
| D.Bradtaueller-483-0896  Denny Przybyla - 432-1228
|
| Treasurer           Editor/Publisher
| Bud Darr - 693-3341    Pat Murphy - 485-2623
|
| Vice-president     Librarian
| Bud Darr - 693-3341    Tom Carson - 485-7086
|
*****

```

May 1988

PRESIDENT'S COMMENTS - BY DAVE BRADTMUELLER

I am sorry that I missed the last meeting but I was in Denver, Colorado. From what I was told it went well. Two things, first, I was told that the second drive did not work. I have run out of ideas. If anybody has any suggestions let me know. Second, I still have not had any of the members approach me about doing any demos. If nobody contacts me before the next meeting we will have a very short meeting.

Don't forget about the User Group Conference in Lima Ohio on May 21. As you know we have a table reserved. I need people to man this table. Let me know if you can by this meeting. Later on in this newsletter you will find more info on this Conference.

We have the 2nd Saturday of the month reserved for the next 3 months from 9:00 to 11:30, so plan ahead. The next meeting will be on May 14, at 9:30 am. The doors open at 9:00 so try to be there as early as possible. See you all there.

MINUTES

Meeting of April 09, 1988
Submitted by Dennis B. Przybyla, Secretary

Vice-President Bud Darr called the meeting to order at 9:34 am. There were eleven people present including nine members and two family members. The Secretary's minutes of the March meeting and the Treasurer's report for April were published in the April Newsletter. The formal presentation of these reports at the meeting were omitted but submitted as published. There were no additions or corrections to the reports of the Secretary and Treasurer and the reports were approved as published. Bud presided over this meeting in Dave Bradtmueller's absence. Dave was out of town on company business.

Tom Carson reported that the United Data Base primary disk has arrived. As yet, he has not completed his review of the material. He did indicate, though, that he will now have to index the articles that appeared in our Newsletters for 1987. The information will be entered in PR Base format.

Bad news! Our disk controller card was sent to TI for an exchange. The replacement disk controller card has been received and installed in our system. The result? The system still acts the same. It will not recognize the second disk drive. Help! Does anyone have any suggestions?

There was limited discussion on our user group's attendance at the LIMA TI MULTI USER GROUP CONFERENCE. It was felt that all required planning for this event could be handled at our May 14 meeting. The Conference will be held at the OSU Lima Branch Campus from 11 am to 6 pm, Saturday, May 21. The complete schedule of events will be printed elsewhere in this Newsletter. There is no charge for admission. Hopefully, a good number of us will be able to attend this Conference. It should be an interesting and worthwhile event.

Some members were interested in the progress of the club's project on "Improved Video". This project was the replacement of a VDP Load Resistor in the TI console from 560 ohms to 330 ohms which would provide about a 40% improvement in the picture quality. Delbert Turner volunteered to replace the load resistor in his console first before we attempt it on the club's console. Delbert was absent at this meeting. Hopefully, he can bring us up to date on his progress at the May meeting.

Blair MacDermid suggested that the club purchase COMPUTERFACTS by Sams, a technical service data manual for your TI99/4A. Pat Murphy informed the group that he had this manual and it was available for use by Blair and whoever else had need of it.

Tom Carson reminded the group of the variety of reading material available in our library and for members to make use of it. Bud Darr also related that he has past issues of the Computer Shopper that he would like to give away. Pat Murphy requested of the members to submit material for the Newsletter. Charles Delabarre announced that one of their TI expansion systems is now for sale. If you are interested, contact Charlie.

Blair MacDermid announced that the Forth User Group meeting will be held on Tuesday, April 12, 7:00 pm, room 138 Neff Hall, on the IPFW campus. This was a personal invitation by Blair. He will be giving a presentation on the Forth language!

Following the general discussion period, Pat Murphy gave a demonstration on the utility program EZ-KEYS. The program was written by Harry Wilhelm and produced by Asgard Software. Pat also demonstrated the program CALCULATOR, which was recorded from an Extended Basic program listing printed in the MICROpendium magazine. These are both very neat programs, especially EZ-KEYS for those who do a lot of extended basic programming, or record (type) basic programs.

The door prize winner was Bud Darr! Yes, he won again. He received two blank diskettes. Our 1988 membership now stands at nineteen.

The next meeting will be held on Saturday morning, May 14, at the Shawnee Branch Library. The meeting start time will be at 9:30, with the doors opening at 9:00 am.

Topics covered in The TI Forum column of the Computer Shopper magazine, May issue, were PC Pursuit Problems, Autodialer From MDOS, Careful With That Mini Memory Battery, Charles Earl Releases TELCO - a terminal emulator for both the TI-99/4A and Myarc 9640, Ron's Part, In The Mailbox, New Software, Coming Attractions, and Winding Down. Absent in this issue was the one-half page advertisement on the Myarc GENEVE 9640.

The MICROpendium's featured articles in the March issue are Regena on Basic (signed numbers, more or less), c99 (onward through the fog), Fileread (another program that displays a variety of file types directly to the screen), Text-to-Speech (using the GRAM Kracker to access this TELL feature), Geneve 9640 (Solutions to some problems, Autodialing from MDOS, Using the Horizon RAMdisk), Myarc's new disk controller (for hard and floppy disks with the TI or 9640), PEB meets hard disk (installing a hard disk in the PEB), Reviews (Telco, String Master, Epson LX-900 Printer), Newsbytes (more BBSeS, a book on controlling printers, and an upgrade for Asgard's Legends), and User Notes (XBASIC with DV/80 files, recycling printer ribbons, and making those numbers stand out).

Membership/Treasury
by Bud Darr

One month has past with no funds paid out. This means the club balance remains as the same balance reported for April 1, 1988. Of course I know we will have a hefty bill due to the postage on a couple months news letters.

	DEBITS	
OLD BALANCE.....		\$414.17
NO FUNDS PAID OUT.....		\$0.00
<hr/>		
BALANCE.....		\$414.17
	CREDITS	
NO FUNDS TAKEN IN.....		\$0.00
<hr/>		
NEW BALANCE.....		\$414.17

I guess I owe the club members present at the April meeting, a great big THANKS for the help conducting the meeting. I just hope that the few plant & office closings in the area do not mess up our membership roll.

>>>EDITOR'S NOTES<<<
by P. Murphy

The first article comes from TIGERCUB Software in Columbus, Ohio. The TIGERCUB has just the best of things to sell you. This article is just one of the many interesting things from the TIGERCUB. If anyone is interested in the very useful items from them...contact TIGERCUB SOFTWARE, 156 Collingwood Ave., Columbus, OH 43213. Now, here is the article...very good...thanx Jim Peterson!

ARRAYS AND SORTS
by Jim Peterson

The concept of arrays, and especially of multi-dimensional arrays, is very difficult for many people to grasp. The following is the best explanation that I know of.

A variable name is a box in which you store something. When you write A\$="X" you are telling the computer to "go to the box labeled A\$ and put the character "X" in it". Or, more accurately, "go to the box labeled A\$, throw away anything you find in it, and put "X" in it."

A simple array such as A\$(3) is a row, labeled A\$, of at least 3 boxes, labeled (1), (2), (3), and maybe more. When you tell the computer that A\$(3)="X" you are again telling it to go to the row of boxes labeled A\$, find the box labeled (3), and put "X" in it.

A 2-dimensional array such as $A\$(3,3)$ is a row, labeled $A\%$, of at least 3 filing cabinets, labeled (1, and (2, and (3, and each having at least 3 drawers labeled 1) and 2) and 3). So, you can use $A\$(3,3)="X"$ to tell the computer to find the row of filing cabinets labeled $A\%$, go to the one labeled (3, and open the drawer labeled 3) and put "X" in it.

And in a 3-dimensional array, $A\$(3,3,3)="X"$ tells the computer to find the $A\%$ row of cabinets, find the one labeled (3 and find the drawer labeled ,3, and find the folder in that drawer labeled 3) and put.....

Finally, you can write $A\$(2,2,2,2,2,2,2,2,2)="X"$ to tell the computer to find row $A\%$; cabinet (2 ; drawer ,2 ; folder ,2 ; paper 2, in the folder; line 2, on the paper; word 2, on the line; and letter 2) of the word!

Yes, TI Extended Basic can handle 7-dimensional arrays, but it is not very practical. Try running this - 100 DIM $A(3,3,3,3,3,3,3)$ - and you will get MEMORY FULL IN LINE 100. Arrays with several dimensions are very wasteful of memory. I don't think I have ever seen a program that used more than a 4-dimensional array, and very rarely more than 3 dimensions.

Now then - $A\$(J)="X"$ means "go to the box labeled "J", find the number in it, then go to the row of boxes labeled $A\%$ and find the box in that row which is labeled with that number....."

And even something as horrible-looking as $A\$(Y(J),Z(A,B))="X"$ just tells the computer to -

1. go to box J and find the number in it;
2. go to row of boxes Y and find the number in box number J of that row;
3. go to box A and find the number in it;
4. go to box B and find the number in it;
5. go to the row of filing cabinets labeled Z, find the one labeled with number A, open the drawer labeled with number B and find the number in it;
6. go to the row of filing cabinets labeled $A\%$, find the one labeled with the number you found in $Y(J)$, open the drawer labeled with the number you found in $Z(A,B)$ and;
7. put the "X" in it!

Remember that, in a multi-dimensional array, only the last dimension holds the value; the others are just pointers to its location. $A\$(2,3)=A\$(3,3)$ throws out whatever is in the 3rd drawer of the 2nd cabinet of the $A\%$ row, and replaces it with whatever is in the 3rd drawer of the 3rd cabinet of that row, but the contents of the 3rd drawer of the 3rd cabinet are unchanged.

Also remember that box X or box $X(1)$ or cabinet drawer $X(1,1)$ or whatever, contain a 0 until you put something else in; box $X\%$ or $X\$(1)$ or drawer $X\$(1,1)$ contain nothing at all until you put a string value into them. When you put something in the box, you throw away whatever was previously in the box. And to empty a box without putting anything in, you put a 0 in a numeric box or "" into a string box.

Enough, on that subject. Now, when you have all your data crammed into an array, the next thing you will probably need to do is to sort it into alphabetic or numeric sequence.

Sorting is one of the hardest jobs that you can give to a computer, and one of the things that a computer is the slowest at doing. Your TI can figure your bank balance in a split second, but might take half an hour to sort your mailing list.

Here's why. You can sort a bridge hand of 13 cards into sequence in 13 moves or less, by simply pulling out each card and slipping it back into its proper place. But, suppose those 13 cards were in 13 boxes, and you had to sort them without removing them from the boxes, except that you could hold one card in your hand? Even if you could figure out the best way, it would take you far more than 13 moves.

That is the problem that the computer has. You have just learned that the computer stores all those values in labeled boxes, or file drawers, and therefore must sort them by shuffling them from one box to another, emptying a box to shuffle into by holding one value in a temporary box while its value is compared with the others to find its proper place.

Of course, you could just set up a new row of empty boxes, and then search through the old boxes for the lowest value and move that to the first box in your new row, etc. - but that would double the amount of memory that the job would require. This would be no problem for a small array, but the computer can sort small arrays fast enough by the one-row method - it is the largest arrays that are too slow by the one-row method and would need too much memory by the two-row method.

Many ingenious routines have been written to accomplish these one-row sorts. I have written a program called "Sort Watcher" which enables you to actually watch various sorts taking place on the screen. It will also tell you the number of swaps and comparisons that were made.

This program demonstrates that the time required for a sort increases greatly as the size of the array increases. Sorting an array of 20 does not take just twice as long as sorting an array of 10 - it may take 4 times as long. For this reason, some of the faster and more complex sorting routines divide an array into smaller segments to be individually sorted and then merged.

After an array has been sorted, my program will also let you change any value in any part of the array, and then let you watch the array being resorted. From this, you will learn that a sorting routine which is very fast for a completely random array may be very slow for an array which is already almost in sequence!

In fact, to add just one additional value to a sorted array, the fastest method is the simple "shoehorn" - just set up an empty box at the end of the row, and move each value down by one box until you come to the proper place for the new value.

A sorting routine can be either numeric or alphabetic depending on whether the variable names used are numeric or string. A numeric sort will be in strict numeric sequence and an alphabetic sort will be in ASCII sequence. That means that if all your strings are composed of upper case alphabetic characters, or all are lower case alphabetic characters, you will get an alphabetic sort - but if they are mixed, all of the upper case strings will come before any of the lower case strings, because the upper case ASCII's are 65-90 and the lower case are 97-122. And if you have lower case words with capitalized initial letters...!

For the same reason, if you perform an alphabet sort of strings containing numeric digits, you will not get a numeric sequence - 10000 will come before 2 because 1 has a lower ASCII code than 2. It would be extremely difficult to devise a sorting routine which could sort numeric digits numerically within strings. However, if all the numbers are the same length, such as ZIP codes, the ASCII sort will be numeric.

Sorting a multi-dimension array becomes a very complex task. If you swap values around without also swapping all the related values, you will end up with complete garbage. Swapping all the related values takes time, and a dimensioned temporary variable name is also required.

Another way around this is to combine the data from an array into simple strings, or set it up originally as simple strings, and then perform a simple sort based on a specified segment of the string. For instance, you could use TI-Writer with tab settings to create a mailing list having first name at tab 1, second name at tab 15, address at tab 25, city at tab 45, state at tab 55 and zip code at tab 65. Then you could sort into last-name alphabetic sequence by sorting on SEG\$(M\$(J),10,255), or into zip code sequence by sorting on VAL(SEG\$(M\$(J),70,5)).

When using TI-Writer to set up such a file, be very sure to save it by PF with the C option, not by SF, and don't leave any blank lines at the end or elsewhere.

Alternatively, elements of data can be crammed into a string separated by control codes, and sorted by position of the code - FOR J=1 TO 5 :: READ A\$:: M\$=M\$&CHR\$(J)A\$:: NEXT J and then sort on element X SEG\$(M\$(J),POS(M\$(J),CHR\$(X),1),255)

***** PROGRAM WITH PRE-SCAN *****
by PATRICK MURPHY

This month's program is a universal file reader written by Robert Carmany. It is supposed to read any file (almost any) that is not in program format. It is supposed to read D/V 80, D/F 80, INT/V & INT/F files. I personally have only used it to read D/V 80 files, but assume it will do equally as well with the other types.

The first version is not PRE-SCANNED, but is written with enough fore sight to have the program initially run almost as fast as if it were PRE-SCANNED!! Nice programming!! The first listing is the original, the second is PRE-SCANNED. Even after PRE-SCANNING, the program could be made shorter yet with XBASHER! Xbasher will be available at the MULTI-USER GROUP CONFERENCE in Lima, Ohio!

I have given a copy to our librarian (as usual). Thanx Robert Carmany! (Note "underline" = 3, i.e., _=3).

```

110 ! *****
120 ! * UNIVERSAL *
130 ! * FILE READER *
140 ! * BY *
150 ! * BOB CARMANY *
160 ! * VERSION 4.0 *
170 ! * XB *
180 ! *****
190 ON BREAK NEXT
200 CALL CLEAR :: CALL SCREEN(5):: DISPLAY AT(2,8):"FILE READER:" VERSION 4.0:" "&RPT$("-",24):: DISPLAY AT(14,1):"
&RPT$("-",24)
210 DISPLAY AT(10,4)BEEP:"INSTRUCTIONS (Y/N)? N" :: ACCEPT AT(10,24)VALIDATE("Yy Nn")SIZE(-1):CHOICE$ :: DISPLAY
AT(10,4):RPT$("-",24):: IF CHOICE$="N" OR CHOICE $="n" THEN 220 ELSE 60SUB 560
220 DISPLAY AT(8,1)BEEP:"Device Filename:" :: ACCEPT AT(9,2)SIZE(15):FILE$ :: IF FILE$="" THEN 220 :: CALL ERASE
230 DISPLAY AT(16,1):"Device.Filename" :: DISPLAY AT(17,1):"&FILE$"
240 DISPLAY AT(8,1)BEEP:"File Descriptors (Choose 1)" :: DISPLAY AT(9,1):"1) DISPLAY ,VARIABLE:"2) DISPLAY ,FIXED"
250 DISPLAY AT(11,1):"3) INTERNAL ,VARIABLE:"4) INTERNAL ,FIXED" :: DISPLAY AT(13,20):"1" :: ACCEPT
AT(13,20)VALIDATE("1234")SIZE(-1):A :: CALL ERASE
260 DISPLAY AT(8,1)BEEP:"Record Length" :: DISPLAY AT(9,6):"80" :: ACCEPT AT(9,6)VALIDATE(DIGIT)SIZE(-3):L :: CALL ERASE
270 ON A GOTO 280,290,300,310
280 ON ERROR 620 :: DISPLAY AT(20,1):"DISPLAY ,VARIABLE" :: DISPLAY AT(20,19):L :: OPEN #1:FILE$,INPUT ,DISPLAY ,VARIABLE
L :: 60TO 320
290 ON ERROR 620 :: DISPLAY AT(20,1):"DISPLAY ,FIXED" :: DISPLAY AT(20,16):L :: OPEN #1:FILE$,INPUT ,DISPLAY ,FIXED L ::
GOTO 320
300 ON ERROR 620 :: DISPLAY AT(20,1):"INTERNAL ,VARIABLE" :: DISPLAY AT(20,20):L :: OPEN #1:FILE$,INPUT ,INTERNAL,VARIABLE
L :: 60TO 320
310 ON ERROR 620 :: DISPLAY AT(20,1):"INTERNAL ,FIXED" :: DISPLAY AT(20,17):L :: OPEN #1:FILE$,INPUT ,INTERNAL,FIXED L ::
60TO 320
320 DISPLAY AT(8,1):"Parameter Flag ":"1 For DISPLAY Files:"2 For INTERNAL Files" :: DISPLAY AT(10,22):"1" :: ACCEPT
AT(10,22)VALIDATE("12")SIZE(-1):Z :: CALL ERASE
330 F=Z
340 DISPLAY AT(8,1)BEEP:"Printout of file? (Y/N) N" :: ACCEPT AT(8,25)VALIDATE(" YyNn")SIZE(-1):P$

```

```

350 IF P$="N" OR P$="n" THEN 450
360 DISPLAY AT(8,1)BEEP:"Printer devicename
ACCEPT AT(10,2)SIZE(-3):PRINT$
370 DISPLAY AT(23,1):"Printer > "&PRINT$ :: CALL ERASE
380 ON ERROR 620 :: OPEN #2:PRINT$,OUTPUT,DISPLAY
390 IF EOF(1)THEN 510
400 ON F GOTO 410,420
410 LINPUT #1:A$ :: GOTO 430
420 INPUT #1:A$
430 CALL KEY(0,K,S):: IF K=13 THEN 540 :: IF S<> THEN 430 ELSE PRINT A$ :: PRINT #2:A$
440 GOTO 390
450 IF EOF(1)THEN 520
460 ON F GOTO 470,480
470 LINPUT #1:A$ :: GOTO 490
480 INPUT #1:A$
490 CALL KEY(0,K,S):: IF K=14 THEN 540 :: IF S<> THEN 490 ELSE PRINT A$
500 GOTO 450
510 CLOSE #2
520 CLOSE #1
525 FOR DE=1 TO 2000 :: NEXT DE
530 DISPLAY AT(12,1)ERASE ALL:"Read Another File? (Y/N) N" :: ACCEPT AT(12,26)SIZE(-1)VALIDATE("YyNn"):Z$ :: IF Z$="Y" OR
Z$="y" THEN 200 ELSE
540 END
550 DISPLAY AT(2,1)ERASE ALL:"TO USE THIS PROGRAM, SIMPLY":"FOLLOW THE INPUT PROMPTS":"AS THEY APPEAR ON THE SCREEN"
560 DISPLAY AT(9,1):"THE 'RECORD LENGTH' PROMPT":"IS THE RECORD LENGTH THAT":"APPEARS AT THE END OF THE":"FILE
DESCRIPTION":"(IE. D/V 80)"
570 DISPLAY AT(16,1):"YOU MAY STOP THE SCREEN":"SCROLLING BY PRESSING ANY":"KEY OR <ENTER> TO ABORT THE":"FILE
PRESENTATION":
580 FOR DE=1 TO 3000 :: NEXT DE
590 CALL CLEAR :: DISPLAY AT(1,8):"FILE READER":" ",24):: DISPLAY AT(14,1):" &RPT$("-",24):: RETURN
610 CALL CLEAR :: CALL SCREEN(7):: DISPLAY AT(12,1):"YOU HAVE JUST ENCOUNTERED ":"FATAL, I/O FILE ERROR,":"PLEASE RE-ENTER
YOUR FILE"
620 DISPLAY AT(15,1):"PARAMETERS" :: FOR DE=1 TO 1000 :: NEXT DE
630 SUB ERASE :: DISPLAY AT(8,1):RPT$(" ",162)
640 SUBEND

```

Now for the PRE-SCANNED version...

```

1 @=0 :: [=1 :: ]=2 :: =3 :: \=4 :: GOTO 190 :: A$,CHOICE$,FILE$,P$,PRINT$,Z$ :: @,A,DE,F,K,L,S,Z,[,\,],_ :: CALL CLEAR
:: CALL KEY :: CALL SCREEN :: !@P-
190 ON BREAK NEXT
200 CALL CLEAR :: CALL SCREEN(5):: DISPLAY AT(1,8):"FILE READER":" VERSION 4.0:" " &RPT$("-",24):: DISPLAY AT(14,1):"
&RPT$("-",24)
210 DISPLAY AT(10,\)BEEP:"INSTRUCTIONS (Y/N)? N" :: ACCEPT AT(10,24)VALIDATE("Yy Nn")SIZE(-):CHOICE$ :: DISPLAY
AT(10,\):RPT$(" ",24):: IF CHOICE$="N" OR CHOICE $="n" THEN 220 ELSE 60SUB 560
220 DISPLAY AT(8,1)BEEP:"Device Filename:">" :: ACCEPT AT(9,1)SIZE(15):FILE$ :: IF FILE$="" THEN 220 :: CALL ERASE
230 DISPLAY AT(16,1):"Device.Filename" :: DISPLAY AT(17,1):"&FILE$
240 DISPLAY AT(8,1)BEEP:"File Descriptors (Choose 1)" :: DISPLAY AT(9,1):"1) DISPLAY ,VARIABLE":"2) DISPLAY ,FIXED"
250 DISPLAY AT(11,1):"3) INTERNAL ,VARIABLE":"4) INTERNAL ,FIXED" :: DISPLAY AT(13,20):"1" :: ACCEPT
AT(13,20)VALIDATE("1234")SIZE(-):A :: CALL ERASE
260 DISPLAY AT(8,1)BEEP:"Record Length" :: DISPLAY AT(9,6):"80" :: ACCEPT AT(9,6)VALIDATE(DIGIT)SIZE(-):L :: CALL ERASE
270 ON A GOTO 280,290,300,310
280 ON ERROR 620 :: DISPLAY AT(20,1):"DISPLAY ,VARIABLE" :: DISPLAY AT(20,19):L :: OPEN #L:FILE$,INPUT ,DISPLAY ,VARIABLE
L :: GOTO 320
290 ON ERROR 620 :: DISPLAY AT(20,1):"DISPLAY ,FIXED" :: DISPLAY AT(20,16):L :: OPEN #L:FILE$,INPUT ,DISPLAY ,FIXED L ::
GOTO 320
300 ON ERROR 620 :: DISPLAY AT(20,1):"INTERNAL ,VARIABLE" :: DISPLAY AT(20,20):L :: OPEN #L:FILE$,INPUT ,INTERNAL,VARIABLE
L :: GOTO 320
310 ON ERROR 620 :: DISPLAY AT(20,1):"INTERNAL ,FIXED" :: DISPLAY AT(20,17):L :: OPEN #L:FILE$,INPUT ,INTERNAL,FIXED L ::
GOTO 320
320 DISPLAY AT(8,1):"Parameter Flag ":"1 For DISPLAY Files":"2 For INTERNAL Files" :: DISPLAY AT(10,22):"1" :: ACCEPT
AT(10,22)VALIDATE("12")SIZE(-):Z :: CALL ERASE
330 F=Z
340 DISPLAY AT(8,1)BEEP:"Printout of file? (Y/N) N" :: ACCEPT AT(8,25)VALIDATE(" YyNn")SIZE(-):P$
350 IF P$="N" OR P$="n" THEN 450
360 DISPLAY AT(8,1)BEEP:"Printer devicename " :: DISPLAY AT(10,1):">PIO" :: ACCEPT AT(10,1)SIZE(-):PRINT$
370 DISPLAY AT(23,1):"Printer > "&PRINT$ :: CALL ERASE
380 ON ERROR 620 :: OPEN #1:PRINT$,OUTPUT,DISPLAY
390 IF EOF(1)THEN 510
400 ON F GOTO 410,420
410 LINPUT #L:A$ :: GOTO 430
420 INPUT #L:A$
430 CALL KEY(@,K,S):: IF K=13 THEN 540 :: IF S<>@ THEN 430 ELSE PRINT A$ :: PRINT #1:A$
440 GOTO 390
450 IF EOF(1)THEN 520
460 ON F GOTO 470,480
470 LINPUT #L:A$ :: GOTO 490
480 INPUT #L:A$
490 CALL KEY(@,K,S):: IF K=14 THEN 540 :: IF S<>@ THEN 490 ELSE PRINT A$

```


Lastly...here is some information on PLUS! from Charles Good (a fairware copy from Charles is in our library). this program has some nice qualities. Actually it is a series of small programs that perform a variety of tasks!! Very good!! Here is a brief synopsis of the disk (paraphrased) from Charles Good. Thanx...

PLUS! by Jack Sughrue

The main part of PLUS! has to do with enhancements you can use in TI-Writer to ornament your text as in headers and hi-lites. Here are some of the utility programs in XB on the PLUS! Disk...

- CAT catalogs a disk and provides one keypress load and run.
- INSTALABEL prints any six lines of text on a label.
- INSTAPRINT prints D/V 80 files to a printer.
- INSTADUMP is a fast XB screen dump.
- INSTAMAIL asks for a name, address, city, state and prints in tabular format.
- YEARLY/CAL prints four months per sheet.
- DESK/CAL prints one month per sheet.
- BANNER makes banners with large letters.
- 3/COL prints a disk directory in 3 columns.
- IG!PAY prints text in pig latin.
- SETUP configures printers.
- tinyteeny is a small word processor...prints very small!
- PLUS!VIEW presents features of PLUS!.
- MULTICOLUMN prints text in two column format.
- GOTHIC allows neat gothic print.
- SMALLIFY squeezes single line XB programs into autiline format.
- MAX-RLE...this is an old one that needs no explanation.

** LIBRARY NEWS **
 by Tom Carson
 Returns next month!

 W.I.S.H. LIST

- WISH GAC0686...A terminal emulator program that would transfer a complete disk rather than one file at a time.
- WISH TM0686...Would like a ribbon cable connector (female) for a 36 pin .100" card edge connector.
- WISH GAC0786...A program that would download different fonts to a dot matrix printer i.e., script, gothic, roman, etc.
- WISH JY0886...A program that converts CALL LOAD statements into assembly language source code.
- WISH BCD0986...Would like a used expanded system (RS232 Optional), at least one disk drive and 32K memory expansion.
- WISH GCC1286..Would like a cassette cable and educational programs or modules (primary level).

- ANSWER GAC0686...Use FREWARE program "MASS TRANSFER" available from Stuart Olson, 25322 W. Mayside Place, Lake Villa, IL 60046. Program now in user group library.
- ANSWER TM0686...Connectors you want are available from PILGRIM'S PRIDE, 5 Williams Lane, Hatboro, PA 19040
- ANSWER JY0886...The program to convert CALL LOAD to ASM. LANGUAGE source or object code has been written by Tom Freeman of LA99ers and is in our library.
- ANSWER GAC0886...Program called OLDENG prints any TI-WRITER file in old english letters - also Character Sets and Graphics Design III provides 6 full character sets - can be purchased from TEXAMENTS.

x
 FOR SALE: EXPANDED TI-99/4A SYSTEM - All or Part
 Charles DeLaBarre
 744-4120 home...461-1851 work

LIMA TI MULTI USER GROUP CONFERENCE SATURDAY MAY 21, 1988

UPDATED MAILING April 14, 1988

The Lima Ohio user group is pleased to announce the following list of user groups and suppliers of TI software, hardware, and other goodies that have made plans to be at the conference. We still have others that are interested, but have not made their reservations. GET THEM IN AS SOON AS POSSIBLE.

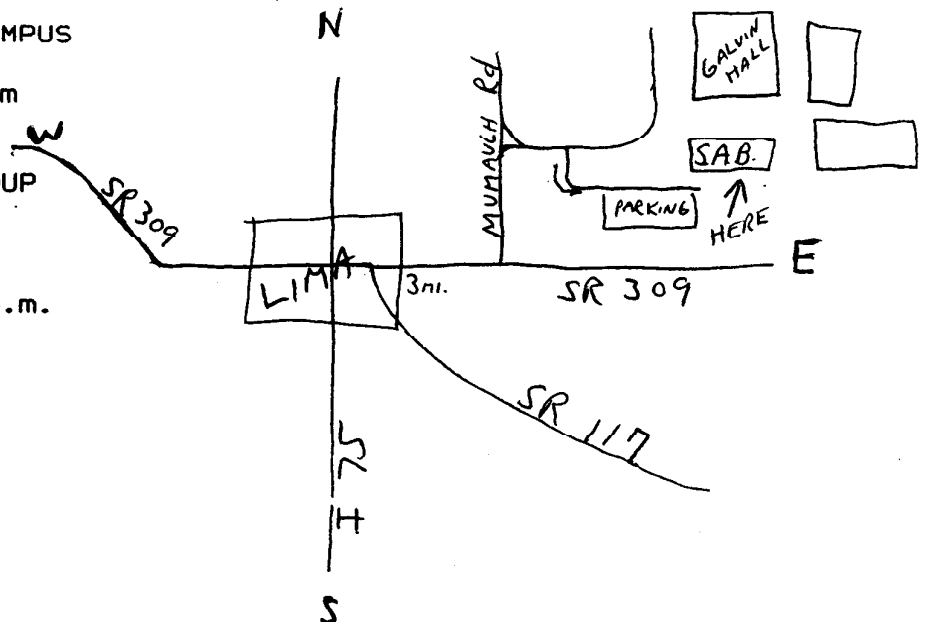
LIMA TI 99/4A USER GROUP
 CIN-DAY USER GROUP
 OH-MI TI USER GROUP
 NEW HORIZONS USER GROUP
 GREAT LAKES COMPUTER GROUP
 MICRO-SERVICE - Printers and repair service
 HORIZON COMPUTER LIMITED - Bud Mills Horizon Ram Disk and console modifications

THE FORT USER GROUP
 MR. JACK SUGHRUE - Author of many articles on the TI 99/4A computer
 TIGER CUB SOFTWARE - Software and TIPS FROM TIGERCUB
 GREAT LAKES SOFTWARE - Software
 MR. IRWIN HOTT - demo on how the blind use the TI 99/4A
 CARNATION CITY 99ers USER GROUP
 C.O.N.N.I. USER GROUP
 RAVE 99 - catalogs
 GLIDDEN ELECTONICS - catalogs
 L.L.CONNER ENTERPRISE - Hardware and software for TI 99/4A
 ASGARD SOFTWARE at Jack Sughrue's table
 GENIAL COMPUTERWARE by J.PETER HODDIE at Jack Sughrue's table
 BARRY TRAVER'S GENIAL TRAVELER at Jack Sughrue's table

We still have tables and conference room time available for those of you that would like to participate in this event. This event is FREE. There is NO CHARGE FOR EXIBITION and NO ADMISSION CHARGE. If you plan on coming, there is still time to pre-register.

PLACE - OSU LIMA BRANCH CAMPUS
 DATE - MAY 21, 1988
 TIME - 11:00 am to 6:00 pm

LIMA AREA TI 99/4A USER GROUP
 c/o Dave Szippi (president)
 4 Poulston Place
 Lima, Ohio 45805
 (419) 228-7109 after 6:00 p.m.



CUT OFF AND RETURN TO PAT MURPHY AT THE NEXT MEETING

Wish I Sure Had...

W.I.S.H. (print or type please)

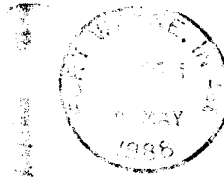
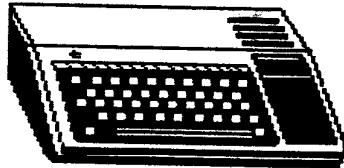
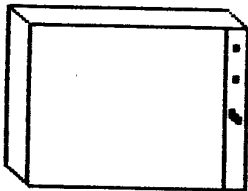
Answer to WISH # _____ (# = name and date rec'd...i.e., GAC0586)

NAME And DATE

<input type="checkbox"/> \$15.00 Renewal	NAME: _____
<input type="checkbox"/> \$15.00 New member	ADDRESS: _____
<input type="checkbox"/> \$ 7.50 Subscribing Membership	CITY: _____ STATE: _____
OFFICE USE	PHONE: (____) _____ - _____ ZIP CODE: _____
<input type="checkbox"/> REC. <input type="checkbox"/> UNUM	INTERESTS: _____
<input type="checkbox"/> PAY <input type="checkbox"/> DATE	

te

The Fort's User Group
5319 Twilight Lane
Ft. Wayne, IN 46835



MIAMI CO. - 99/4A USER GRP
P.O. BOX 1194
PERU IND. 46970

