



MID-ILLINOIS  
COMPUTER RESOURCE ORGANIZATION  
P. O. BOX 766  
BLOOMINGTON, ILLINOIS 61701-0766

MICRO/99 Newsletter  
Volume 4, Number 2  
February, 1986

MICRO/99 is a not-for-profit group dedicated to the sharing of information and public domain software for the Texas Instruments 99/4A home computer. Members have free access to our library of several hundred programs on cassette and diskette. Meetings are held at 7:00 p.m. on the third Thursday of each month at the Illinois Agriculture Association building, 1701 Towanda Avenue, Bloomington. Attendees sign in with the guard at employee entrance number 4 at the rear of the building. Visitors are especially welcome.

\*\*\* FEBRUARY 20 MEETING \*\*\*

At the February 20, 1986 meeting Sam Shank will demonstrate Padlock Plus. This is an encryption program that will scramble the data on some disk files, such as TI Writer text files. The program will decrypt the file only when supplied with the password you used to encrypt it originally. This program also comes with some graphics subroutines and a disk catalog program.

All members are encouraged to share any information gleaned from magazines, catalogs, bulletin boards, newsletters from other clubs, personal experience with products, etc. If you have a computer related question or problem, someone at the meeting may have an answer or suggestion for you. And, you are encouraged to bring and show any interesting program you found or wrote recently.

\*\*\*\* SMART REMARKS \*\*\*\*

My thanks to all those who contributed time and equipment for our booth at the College Hills Mall annual computer show last month. Those who were there had a great time. (I don't know when I've spent so much time at my computer without my wife complaining!) Furthermore, our efforts paid off with several new contacts with TI 99 owners, and lead to the best attended meeting we've had for some time. It was good to see some new faces and some that had been absent for awhile.

Many of those present paid 1986 membership dues of \$15 per family. This newsletter is our main expense, and we still need more paid memberships to continue it for another year. Most of you realize that we exchange newsletters with many other clubs around the U.S. and Canada. Many of you have checked out some of our newsletter files to read at home. I feel that our newsletter library is as important as our tape and disk program libraries. So our newsletter is more than just our internal communication vehicle. It's our contact with the rest of the TI 99 world. Please pay your 1986 dues so we can continue this member service.

Sid Smart, President

## \*\*\* NEWSLETTER REVIEW \*\*\*

In case some of you need convincing as to the value of our newsletter exchange, below is an outline of some of the articles in some of the newsletters we've received over the last month. They will be available at the meeting.

## LOS ANGELES, CALIFORNIA:

- Announcement of the first L.A. 99/4A expo, March 1-2, 1986.
- Article on HEMPRINT, a freeware program for printing files created by the Household Budget Management module.
- Review of MINI-WRITER, a word processor that requires only the Mini-Memory module and a cassette recorder.
- Article on programming in FORTH.
- BASIC programs which set screen colors, analyze memory contents, produce a color show and print a graphics design worksheet.
- Instructions on Gram Kracker modifications to Editor Assembler, TI Writer and Disk Manager II modules.
- Wiring diagram for adding a load interrupt switch to your speech synthesizer.

## PITTSBURGH, PENNSYLVANIA:

- Instructions for converting assembler programs to memory image form for faster loading and smaller size.
- An Extended Basic memory dump to a printer.
- Detailed instructions for mounting your keyboard in an external cabinet with a flexible cable.
- Instructions for mounting 32K of memory expansion inside your console with \$15 dollars worth of parts.
- Tutorial on assembly language programming.
- BASIC programs for doing a card trick and for printing graph paper.
- Review of the Pyramid of Doom adventure program.

## WORCESTER, MASSACHUSETTS:

- Code for a BASIC program - Star Fighter.
- Review of the LOGO programming language and some books on it.
- Article on the controversies about disk drive cleaning.
- BASIC program to convert text files to program files.

## ORANGE PARK, FLORIDA:

- Assembly language tutorial on counting and printing numbers.
- FORTH language tutorial on the same, with timing comparisons between languages.

## WILMINGTON, DELAWARE

- Announcement of an East Coast TI Fair on March 15.
- Code for a beginner's data base program in basic using cassette tape.
- BASIC program for adding printer output to other Basic programs.
- Article on using the MYARC RAMDISK with the CORCOMP disk controller.

## HARRISBURG, PENNSYLVANIA

- Note on embedded printer control codes with TI Writer.
- Instructions for changing TI Writer default colors.
- BASIC program for printing Display Variable 80 files.
- Article on the contents of disk sector zero.
- Review of Myarc 128K RAM disk.

## INDIANAPOLIS, INDIANA

- Instructions for putting TI Writer and Editor Assembler GROMs in one cartridge.
- Review of PRBASE, a data base program.

## DECATUR, ILLINOIS

- Review of QUICK CATALOGER, a resident type disk catalog program.
- BASIC programs for fancy text display on the screen, and for setting Epson printer options.

## MYARC 512K MEMORY CARD

---

by Brian McFeeters

Last month, I reported on MYARC's 128K memory card. Since then, I have received the necessary chips to bring the total memory to 512K. To go from 128K to 512K requires twelve 32K chips. Installation is very easy. After opening the card, you will see twelve empty IC sockets for plugging the chips into. Each chip has a dot for orientation. Just plug them the same as the chips already in the card.

With the extra chips, the available ramdisk memory is 400K (1598 sectors). 80K is available for print spooling. However, when using a sector copying program to load the ramdisk, it will reformat the ramdisk. Since the maximum I can format a disk is DS/SD, most copy programs will reformat the ramdisk to that format. Fortunately, Masscopy will allow you to specify the object disk's format. Since you can choose DS/DD, most of the available 400K can be used. DS/DD gives 1440 sectors or about 360K to use. If you carefully choose what is on the ramdisk, 360K goes a long way. I have loaded my ramdisk with TI-WRITER and MULTIPLAN files, several games and utilities including DM1000, and some modules to load thru the GRAM-KRACKER. It took about 90 seconds to load a DS/SD disk using Masscopy.

Another program that is on my ramdisk is TI-ARTIST. It comes with a program to install it on MYARC's 128K-512K card. However, although the instructions with TI-ARTIST say it can be loaded onto a 128K card, I could not get to load until I added the extra chips. It is especially nice having it on the ramdisk if you go back and forth between the drawing portion of the program and the enhancement section.

So far, I have had only two problems with the card. Twice the memory has been wiped out when the PE box was turned off even with the power supply plugged in. I haven't been able to explain it. Also, using DISK+AID (sector editor) to edit the ramdisk messes it up. Most programs wouldn't load after I used it. DISK+AID does work fine on a floppy disk with the card plugged in.

I ordered the chips and card thru Hunter Electronics. The cost of the chips were \$6.00 a piece for a total of \$72.00. The 128K card cost \$199.00. Their price for a 512K was \$299.00. So far, I have been very pleased with Hunter Electronics. Everything I have ordered was shipped promptly. Below is their address and phone number.

HUNTER ELECTRONICS  
604 S. Fairview Ave.  
Elmhurst, IL 60126  
(312)832-6558

## PE BOX FAN

---

by Brian McFeeters

Our January newsletter listed an address for a company that was selling quiet cooling fans to replace the noisy one in the PE box. I ordered one and received it in about one week. The fan comes with instructions which are complete enough to install it. [cont. next page]

The fan is fairly easy to install. I will list the basic steps for installation. First, all drives and expansion cards should be removed from the box (wait two minutes after turning power off). Next, about fifteen screws need to be removed from the back, sides and bottom. After that, you can slide the cover off from the chassis. Now, the fan can be removed. It is attached by four very small nuts. There are two wires running to the fan. Along the wires you will see a butt type connector. Carefully pull these apart. My connectors were very tight and difficult to separate. Now, you are ready to reassemble everything. The new fan requires you add a connector with a blade end so you can reconnect it to the existing PE box wire. Refer to the instructions included with the fan for more detail.

The whole process took almost 1 1/2 hours. It was worth it. You can barely hear the fan running. At first, you keep checking to see if the box is on because you don't hear the loud fan noise. It's too bad TI didn't use these fans.

Although our newsletter and MICROpendium stated the total cost was \$17.00, I was charged \$18.00, still a reasonable price to keep from going deaf. Below is the address for ordering:

STATCO, Inc.  
P.O. Box 145  
Townsend, MA  
01469-0145  
(617)433-2070

## TK-WRITER REVISION

-----

From the Jackson County 99'ers, comes this tip on modifying Tom Knight's Extended Basic load program for TI-Writer.

Apparently, when going from the Editor to the Formatter, the LOAD program reloads the assembly program, not checking to see if it is still in memory. The resulting wait can be avoided by making the following modifications to the LOAD program. In line 108 where it says "DSK1.TKWRITER", put in the program name you are using for TK-WRITER.

```
100 CALL CLEAR :: CALL INIT :: CALL PEEK(-2043,A,B):: IF A<>84 OR
    B<>75 THEN 108
102 CALL LOAD(16360,85,84,73,76,73,84,250,212,70,79,82,77,65,84,250,
    132,69,68,73,84,79,82,250,22)
104 CALL LOAD(8196,63,232):: GOTO 110
108 CALL LOAD("DSK1.TKWRITER")
```



# TI-ARTIST COMPANION #1

by Brian McFeeters

One of the best programs available for the TI-99/4A is TI-ARTIST. Now, there is a companion package that can be used to enhance the already great graphic capabilities of TI-ARTIST. Dave Rose is offering a five 5.25 disk package that adds 25 fonts, 30 pictures (instances), and 165 graphic characters. The additional graphics are loaded thru the ENHANCEMENT option. I recently purchased the package and have enjoyed using the various fonts and characters. The companion package is definitely worth the \$17.95 Dave is charging especially if you do much printing with TI-ARTIST. Below is the address for ordering:

DAVE ROSE  
2781 RESOR ROAD  
FAIRFIELD, OHIO 45014-5053

Below are some examples of the fonts and characters available.



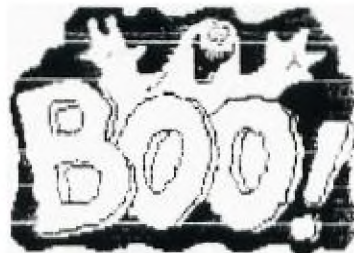
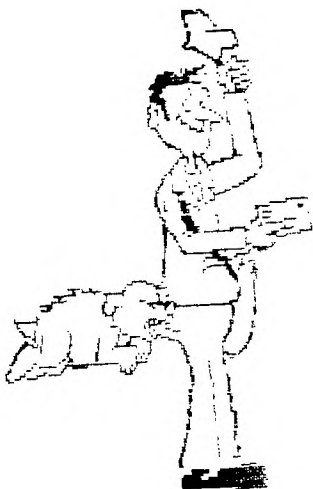
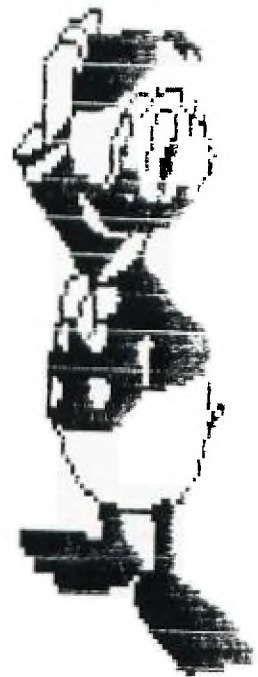
# TI ARTIST

Ti-99 4A

MICRO 99

COMPUTER

# DISK



Copyright 1985

TIGERCUB SOFTWARE  
156 Collingwood Ave.  
Columbus, OH 43213

NUTS & BOLTS DISK No. 2 is now ready, and I think it's better than the first one. It contains 188 utility subprograms in merge format, including many new character fonts and screen display routines as well as 2-dimensional array sorts, variable line numbers in GOSUB, GOTO and RESTORE, on-screen editing and much, much more. The price is \$19.95 postpaid, or you can order both Nuts & Bolts disks for \$37 ppd.

And I have put together 18 different collection disks each containing 5 or 6 of my catalog programs for just \$12 postpaid. The programs on each disk are all of the same category, and I have filled up the rest of the disk with public domain programs of the same category, as a bonus.

I want to make it very plain that I am NOT - repeat, NOT - selling public domain programs! My own programs on these disks are offered at a great discount and the public domain programs are just thrown in for free! Together with this issue of the Tips I am mailing to each user's group a copy of my catalog #6 with an added page describing these new offerings, and a rebate offer to user's groups.

My catalog will be sent to individuals for \$1, which is deductible from your first order. If you already have my catalog #6, the added page will be sent to you

free on request.

My full disk collections will now be available to bona-fide retailers at standard wholesale prices. Inquiries on your letterhead are invited.

And so, on to old business. Yes, I know that RESequencing a program does not resequence references to line numbers in REMs. I just forgot! In line 278 of the Menu Loader in Tips #27, the reference should be to lines 288 and 298, of course.

While programming the file reader in that menu loader, I ran into a peculiarity of the TI-99/4A that surprised most of the expert programmers whom I called for help. When you "read blind" you must read everything as a string, because attempting to read a string as numeric will crash the program. This is no problem with DISPLAY files - but when I tried it with INTERNAL files, I got the strangest garbage! My solution (not quite fool-proof) was to identify a record as numeric if it was 8 bytes long and contained an ASCII out of printable range, and then RESTORE the file, read back to that point and re-read it as numeric. Not very efficient!

The following routine will save a numeric input in an internal file, read it back out as a string, show you the way it was saved, and then attempt to translate it back to numeric. It works for positive and negative integers or non-integers of not less than -99, but not for less than that.

```
100 INPUT X :: OPEN #1:"DSK1
.TEST",INTERNAL,OUTPUT :: PR
INT #1:X :: CLOSE #1
110 OPEN #1:"DSK1.TEST",INTE
```

```
RNAL,INPUT :: INPUT #1:A$ ::
PRINT A$ :: CLOSE #1
120 FOR J=1 TO 8 :: PRINT AS
C(SEG$(A$,J,1)):: NEXT J
130 FOR J=1 TO 8 :: A(J)=ASC
(SEG$(A$,J,1)):: NEXT J
140 X=A(1)-63 :: IF X<73 THE
N 150
142 X=192-A(1):: N$="-" :: F
OR J=2 TO X+1 :: N$=N$&STR$(
256-A(J)):: NEXT J :: GOTO 1
60
150 FOR J=2 TO X+1 :: N$=N$&
STR$(A(J)):: NEXT J
160 IF A(J)<># THEN N$=N$&".
"&STR$(A(J))
170 J=J+1 :: IF A(J)<># THEN
N$=N$&STR$(A(J)):: GOTO 170
180 N=VAL(N$):: N$="" :: PRI
NT N :: GOTO 100
```

So, here is another Tigercub Challenge! Can you fix it? Let's HEAR from you this time!

Another problem that I ran into was in recovering from an I/O error. When ON ERROR is used to prevent crashing on such an error, the file is "ajar" - you can't close it and you can't open it. My solution was to simply RUN the program again - and this will show you how the pre-scan speeds that up. Since then, I have learned of three other ways. The method described in the Sydney (Australia) newsletter is a bit complicated, but Irwin Hott gave me a simple solution - just increment the file number! Works fine if you don't increment it into the number of another open file on the disk. Chuck Brines gave me an even better way - open and close anything else, even "PIO"! Example -

```
100 ON ERROR 110 :: OPEN #1:
"DSK1.TEST",OUTPUT :: PRINT
"CONTINUE PROGRAM" :: END
110 OPEN #1:"PIO" :: CLOSE #
1 :: PRINT "I/O ERROR":"CHEC
K DISK AND DRIVE":"THEN PRES
S ANY KEY" :: ON ERROR STOP
120 CALL KEY(#,K,S):: IF S=0
THEN 120 ELSE 100
```

There is a reason for that ON ERROR STOP, and it's why I don't use ON ERROR if I can avoid it. When an error occurs, the program goes to the line number specified by the last open ON ERROR statement, takes whatever action is directed by that line, and RETURNS as directed. If the error was not one that you expected to happen, the results can be very confusing!

For that reason, when you set out to modify a program, the first thing you should do is delete, temporarily, all the ON ERROR statements. The next thing you should do, if the program has a routine to turn off the pre-scan, is to disable that. Otherwise, you will be driven crazy by invalid SYNTAX ERROR messages and other strange happenings.

The third thing you should do is to make a list of all the lines that a GOTO or GOSUB goes to, so you don't delete or change them. And here is a program to do just that for you -

```
100 !GO-SEARCH by Jim Peters
on searches a MERGE format f
ile, finds all line numbers
containing a jump, sorts int
o "to" line number sequence,
110 !prints "to" line number
, statement (GO, GOTO or GOS
UB) and "from" line number
120 DIM C(200):: A=1 :: GO$(
1)="GO" :: GO$(2)="GOTO" ::
GO$(3)="GOSUB"
130 INPUT "FILENAME? DSK1.":
F$
140 OPEN #1:"DSK1."&F$,INPUT
,VARIABLE 163 :: OPEN #2:"P
IO"
150 LINPUT #1:A$
160 IF POS(A$,CHR$(133),1)=#
AND POS(A$,CHR$(134),1)=# A
ND POS(A$,CHR$(135),1)=# THE
N 210
170 LN=ASC(SEG$(A$,1,1))+256
+ASC(SEG$(A$,2,1)):: T=133 :
P=1
180 G=CHR$(T):: X=POS(A$,G$
```

```

,P)::: IF X=# THEN 200 :: LRE
F=ASC(SE6$(A$,X+2,1))*256+AS
C(SE6$(A$,X+3,1))!:: PRINT #
2:LN:60$(T-132);LREF :: P=X+
1 :: GOTO 180
190 C$=STR$(LREF)&". "&STR$(L
N)&STR$(T-132):: C(A)=VAL(C$
):: A=A+1 :: P=X+1 :: GOTO 1
80
200 IF 6%=CHR$(135)THEN 210
:: T=T+1 :: P=1 :: GOTO 180
210 IF EOF(1)THEN CLOSE #1 :
: GOTO 220 :: ELSE 150
220 A=A-1 :: CALL LONGSHELLN
(A,C(1))
230 FOR J=1 TO A :: A$=STR$(
C(J)):: X=POS(A$,".",1):: Y=
VAL(SE6$(A$,LEN(A$),1)):: A$
=SE6$(A$,1,LEN(A$)-1)
240 PRINT #2:SE6$(A$,1,X-1);
TAB(7):60$(Y);" FROM ";TAB(2
1):SE6$(A$,X+1,LEN(A$)):: NE
XT J
250 SUB LONGSHELLN(N,NN())
260 D=N
270 D=INT(D/3)+1 :: FOR I=1
TO N-D :: IF NN(I)<=NN(I+D)T
HEN 300 :: T=NN(I+D):: J=I
280 NN(J+D)=NN(J):: J=J-D ::
IF J<1 THEN 290 :: IF T<NN(
J)THEN 280
290 NN(J+D)=T
300 NEXT I
310 IF D>1 THEN 270
320 SUBEND

```

According to the User's Reference Guide that came with your computer, if you open a file without specifying INPUT, OUTPUT, UPDATE or APPEND, the computer will assume the UPDATE mode as the default and "UPDATE files may be both read and written. The usual processing is to read a record, change it in some way, and then write the altered record back out on the file." This is a very dangerous bit of misinformation! It is true only if you are using RELATIVE files with the REC clause. In any other case, the first record you write to the file will become the record FOLLOWING the last record you read, and it will also become the

LAST record in the file - any records beyond that point will be lost! The moral of the story - get in the habit of NEVER opening a file without specifying the mode. The only way to update a sequential file is to read it ALL into an array, update it, and then write it back to the file.

I reviewed hundreds of programs, in my PD library of about 2600, in order to select some of the best to fill up the collection disks. Often they needed only a few minor changes to greatly improve them. One frequent flaw was in interpreting the status of CALL KEY. The User's Reference Guide says that a status variable of -1 means that "the same key was pressed during the performance of CALL KEY as was pressed during the previous performance." This is misleading. It actually means that the same key is STILL BEING pressed. Try this -

```

100 DISPLAY AT(12,1)ERASE AL
L:"TYPE YOUR NAME" :: R=12 :
: C=3
110 CALL KEY(#,K,S):: IF S=#
THEN 110 :: DISPLAY AT(R,C)
:CHR$(K):: C=C+1 :: GOTO 110

```

Difficult to type without unwanted repetition of letters? Now try changing the S=# to S<1 ! IF S<1 (if S is less than 1) means that if no key is pressed (S=#) or if the same key is still being held down (S=-1) then CALL KEY again.

Another frequent flaw is INPUT "WANT TO PLAY AGAIN? " :0\$ :: IF 0\$<>"Y" THEN END - or, more professionally programmed, IF SE6\$(0\$,1,1)<>"Y" THEN.... which will accept either "Y" or "YES" as a reply. The problem is still that this

question is often asked at the end of a joystick game, for which the Alpha Lock will be unlocked - and a response of a lower case "y" then terminates the program! One solution is to precede the INPUT with a dummy CALL KEY(3,K,S), which will cause any subsequent upper case CALL KEY, INPUT, LINPUT or ACCEPT AT response to be read as lower case until you turn it off with CALL KEY(5,K,S).

Here's one that does nothing except look pretty.

```

100 DISPLAY AT(3,0)ERASE ALL
:"COLORSQUARES" :: DISPLAY A
T(8,1):"Select option 1, 2 o
r 3" ! by Jim Peterson, Tig
ercub Software
110 CALL KEY(#,K,ST):: IF ST
=# OR K<49 OR K>51 THEN 110
:: ON K-48 GOTO 150,120,130
120 FOR CH=38 TO 142 STEP 8
:: CALL CHAR(CH,RPT$("ASSA",
4)):: NEXT CH :: GOTO 150
130 FOR CH=38 TO 142 STEP 8
:: FOR L=1 TO 4 :: RANDOMIZE
:: X$=SE6$("###18243C425A667
E8199A5BDC3DBE7FF",INT(16*RN
D+1))*2-1,2)
140 B$=B$&X$ :: C$=X$&C$ ::
NEXT L :: CALL CHAR(CH,B$&C$
):: B$,C$=NUL$ :: NEXT CH
150 CALL CLEAR :: RANDOMIZE
:: FOR SET=#-(K>49)TO 14 ::
CALL COLOR(SET,SET+2+(K>49),
SET+2):: NEXT SET
160 Y=INT(4*RND+3):: R=INT(1
2*RND+1):: R2=25-R-Y :: C=IN
T(7*RND+7):: C2=32-C-Y :: IF
K=49 THEN X=INT(14*RND+1)*8
+22 ELSE X=INT(13*RND+1)*8+3
0
170 FOR T=R TO R+Y :: CALL H
CHAR(T,C,X,Y):: CALL HCHAR(T
,C2,X,Y):: NEXT T
180 FOR T=R2 TO R2+Y :: CALL
HCHAR(T,C,X,Y):: CALL HCHAR
(T,C2,X,Y):: NEXT T :: GOTO
160

```

The asterisk on the Gemini printer looks rather like a bug squashed side- ways, and it was confusing some folks in the condensed print of my

newsletter, so I improved it with this -

```

150 PRINT #2:CHR$(27);CHR$(4
2);CHR$(1);CHR$(42);CHR$(0);
CHR$(8);CHR$(34);CHR$(8);CHR
$(0);CHR$(62);CHR$(0);CHR$(B
);CHR$(34);CHR$(8);

```

And at the same time I improved the slashed zero -

```

140 PRINT #2:CHR$(27);CHR$(4
2);CHR$(1);CHR$(48);CHR$(0);
CHR$(64);CHR$(30);CHR$(96);C
HR$(17);CHR$(72);CHR$(5);CHR
$(66);CHR$(61);CHR$(0);

```

90 !THIS WON'T WORK, WILL IT ?

```

100 DISPLAY AT(9999,9999)ERA
SE ALL:SE6$("CAN'T DO THAT!"
,1,3)&SE6$("CAN'T DO THAT!"
,6,8)

```

If the Tigercub Math Puzzle in Tips #27 was a bit too tough, these changes will add a couple of easier levels.

```

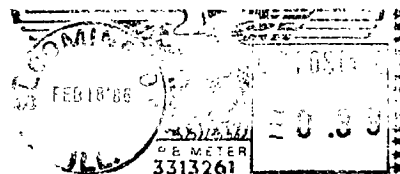
105 DISPLAY AT(6,1):"Level 1
, 2, 3 or 4?" :: ACCEPT AT(6
,21)VALIDATE("1234"):L$ :: L
=VAL(L$)
106 IF L<3 THEN M$="Insert +
, -, or * (multiply)" ELSE M
$="Insert +, -, * (multiply)
or / (divide)"
110 DISPLAY AT(5,1):M$: " bet
ween the digits": " to equal
the total": "Type 0 to give
up"
120 ! **DELETED LINE **
130 DISPLAY AT(12,1):" " ::
T,X=INT(9*RND+1):: M$=STR$(X
):: Z=M$&" "
140 FOR J=1 TO 4 :: Y(J)=INT
(9*RND+1):: @=3+ABS(L>2):: Z
=INT(@*RND+1):: DN Z 60SUB 2
40,250,260,270 :: Z%=Z%&STR$
(Y(J))&" " :: NEXT J
150 IF L/2<>INT(L/2)AND T<>1
NT(T)THEN 130 :: Z%=Z%&"="&S
TR$(T)

```

MEMORY FULL

Jim Peterson

MID ILLINOIS COMPUTER RESOURCE ORGANIZATION  
P.O. BOX 766  
Bloomington, IL 61701-0766



EDMONTON 99'ERS USER SOCIETY  
P.O. BOX 11983, EDMONTON  
ALBERTA, CANADA T5J-3L1

```
*****  
*          MMM   MMM   IIIIII   CCCCCC   RRRRRRRR   00000000   *  
*          MM M M MM     II     CC     RR     RR     00   00     *  
*          MM M M MM     II     CC     RRRRRRRR   00   00     *  
*          MM  M  MM     II     CC     RR     RR     00   00     *  
*          MM     MM     II     CC     RR     RR     00   00     *  
*          MM     MM     IIIIII   CCCCCC   RR     RR   00000000   *  
*  
*  
*          The MID ILLINOIS COMPUTER RESOURCE ORGANIZATION   *  
*****
```