



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

MICRO/99 Newsletter
Volume 4, Number 4
April, 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. Turn left at the sign for the main reception area and go down the stairs on the far side of it. Visitors are especially welcome, and may attend one meeting free of charge.

*** APRIL 17 MEETING ***

At the APRIL 17, 1986 meeting Brian Sydney, a member from Delevan, will present GENFILE, a general purpose filing system that he programmed in extended BASIC. Brian says he has used the program to keep files of paychecks, utility bills, winning lotto numbers, grocery lists and personal checks by category. I have read the documentation and experimented just a bit with the program, and it looks like it could be very useful. It not only maintains the data files, it automatically maintains a master index file which contains information on all the data files, and their backup copies. Brian has said he will make copies available at the meeting.

At our last meeting we had a request for a programming tutorial on the use of arrays. I wrote DICEPLOT, the program on the next page, for that purpose. It illustrates array usage, and also a graphics technique, the use of a subprogram, and some elementary probability theory. I'll discuss it at the meeting.

At all meetings 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 ****

I thought the last meeting was particularly interesting. Aubrey Johnson's telecommunications demo using local bulletin boards was very well received. I wish I were always so well prepared and well organized as he is. Nice job Aubrey!

Our hardware guru, Herb Beer impressed everyone again too. Others have mounted 32K of RAM in the console, but Herb is the only one I know who mounted it on the back of the receptacle in the module port. It made a compact plug-in unit that was passed around for all to see. This mounting also puts it right under the ventilation holes. Herb also had a module case with the chips from 3 modules mounted in it, selectable with a toggle switch. I understand Herb has been promoted into management at the phone company and would appreciate the donation of a suit or two! Congratulations Herb!

Sid Smart, President

The following program was written by George F. Steffan of the LA 99ers Computer Group. It appeared in the NOV85 issue of the Delaware Valley Users Group newsletter.

MULTI COLUMN PRINTING

For the past several months I have used two programs to list programs in our newsletter 28 characters wide as they appear on the screen and three columns wide so they do not waste space. I received a request for the method and at the same time, I saw a program to list programs on a wide printer. So I adapted my programs to be more versatile instead of single purpose.

VARYLIST will take a program listing and convert it to whatever line length you desire. There is one bug: if the listed line is an exact multiple of 80 characters in length, the next line will be appended to it. I can think of no simple solution to this and it is an infrequent occurrence, so it remains in the program. This program works on a program LISTed to disk. If your desired length is 80 or less, the disk file will be opened as VARIABLE 80 so that it may be edited with TI Writer. If you wish to list to a wide printer, the file will be opened with the correct length.

MULTIPRINT will take a text file and output it to the printer in multiple columns so that it may be read in normal newspaper fashion, one column after another. You determine the number of columns, but you must inform the program of the output device. This program has no provisions to enable the output text to be edited. Editing must be done before using it.

Before using MULTIPRINT you should prepare your text file. You should first use VARYLIST or the Formatter of TI-Writer to create a text file of the desired width. Then examine the file and delete any unneeded blank lines. Make sure that the number of lines is an exact multiple of the number of columns you will be using. Insert blank lines to reach this number. You may put these blank lines any place in the text, but they should be placed so as to form pleasing column breaks. If you have used the text formatter to print the file, you should use the Replace String command to change all Line Feeds (Control U, Shift J, Control U), Carriage Returns (Control U, Shift M, Control U) and New Page (Control U, Shift L, Control U) to spaces. Because the text is reformatted after these changes, be sure you are not in Word Wrap Mode when you do this. If you make the first line of your text longer than the line length you plan to tell the printer, it will print across the page as on this article. In this case, you must be sure that the first two lines of succeeding columns are blank. Then save the text file or print it to disk and run MULTIPRINT. The program is designed to accept 300 lines of text, enough for five columns of 60 lines each. If the number is increased too much, the computer will run out of memory.

The programs are listed herewith, each giving an example of itself.

```
100 REM VARYLIST -Geo. F. Steffan, LA 99ers Computer Group, OCT 1985
110 REM THIS PROGRAM WILL CONVERT ANY PROGRAM LISTED TO DISK INTO A LISTING OF ANY WIDTH YOU DESIRE
120 REM IT MAY BE A 28 COLUMN LISTING SIMULATING A SCREEN LIST
130 REM IF LISTED TO DISK AND OUTPUT WIDTH IS 80 OR LESS, OUTPUT MAY BE EDITED WITH TI-WRITER
140 REM IF A NUMBERED LINE IS EXACTLY 80, 160, OR 240 BYTES WHEN LISTED, THIS PROGRAM WILL COMBINE IT WITH THE F
```

```
OLLOWING LINE
150 DATA 3,DSK,WDS,RD
160 CALL CLEAR :: PRINT TAB(11);"VARYLIST"
170 PRINT :: LINPUT "NAME OF INPUT PROGRAM LIST? ":IP$
180 PRINT :: LINPUT "NAME OF OUTPUT FILE? ":OF$ :
: IF OF$=IF$ THEN PRINT "INPUT AND OUTPUT NAMES MUST BE DIFFERENT!" :: GOTO 170
190 PRINT :: INPUT "WIDTH OF OUTPUT FILE? ":OW :: OOW=0 W :: IF OW>79 THEN 220
200 READ N :: FOR I=1 TO N : READ DN$ :: IF SEG$(OF$,1,LEN(DN$))=DN$ THEN OOW,I=80
210 NEXT I
```

```
220 OPEN #1:IP$,DISPLAY ,VARIABLE 80,INPUT :: OPEN #2:OF$,DISPLAY ,VARIABLE OOW,OUTPUT
230 FOR I=1 TO 9999 :: LI$=""
240 IF EOF(1)THEN I=I+10000 :: GOTO 250 ELSE LINPUT #1:L2$ :: IF LEN(L2$)=0 THEN GOTO 240 ELSE L1$=L1$&L2$ :: IF LEN(L2$)=80 THEN GOTO 240
250 FOR O=1 TO LEN(L1$)STEP OOW :: PRINT #2:SEG$(L1$,O,OOW) :: J=J+1 :: NEXT O :: NEXT I
260 CLOSE #1 :: CLOSE #2 :: PRINT :I-10000;"NUMBERED LINES":J;"OUTPUT LINES" :: END
```

```

100 REM MULTIPRINT -Geo. F.
    Steffan, LA 99ers Computer
    Group, OCT. 1985
110 REM TI EXTENDED BASIC AND
    MEMORY EXPANSION
120 REM WILL PRINT MULTIPLE
    COLUMNS OF ANY TEXT FILE
130 DIM L$(300):: CALL CLEAR
    :: PRINT TAB(10);"MULTIPRIN
    T"
140 PRINT :: LINPUT "NAME OF
    INPUT FILE?          ":IF#
    :: INPUT "LENGTH OF INPUT LI
    NES?                  ":LL
150 PRINT :: LINPUT "NAME OF
    PRINTER?             ":P# ::
    INPUT "PRINTER LINE LENGTH?
    ":PL
160 PRINT "COLUMN SEPARATIO
    NS WILL BE CALCULATED." :: I
    NPUT "NUMBER OF COLUMNS? ":
    C
170 IF (2*(M+C-1)+C*LL)>PL T

```

```

HEN PRINT "WILL NOT FIT" ::
GOTO 160
180 OPEN #1:IF#,INPUT ,DISPL
AY ,VARIABLE :: FOR I=1 TO 3
00 :: IF EOF(1)THEN 210
190 LINPUT #1:L$(I):: IF ASC
(L$(I))>127 THEN L$(I)=" ::
    GOTO 210 ! DISREGARD TAB SE
    TTINGS
200 NEXT I
210 CLOSE #1 :: S=INT((PL-(C
*LL+2*M))/(C-1))+LL :: M=M+1
    :: OPEN #2:P#,DISPLAY ,VARI
    ABLE PL+1,OUTPUT
220 N=INT((I/1)/C):: FOR I=1
    TO N :: FOR J=0 TO C-1 :: F
    RINT #2:TAB(J*S+M);L$(I+J*N)
    ;:: IF I=1 AND LEN(L$(1))>LL
    THEN J=C
230 NEXT J :: NEXT I :: CLOS
E #2 :: END
240 STOP

```

BREAKDANCING

This program was written by Travis Ringold. We reprinted it from the SEPT/OCT issue of A9CUG CALL NEWSLETTER.

Press a number key and they change steps. Hold down one key, four keep dancing.

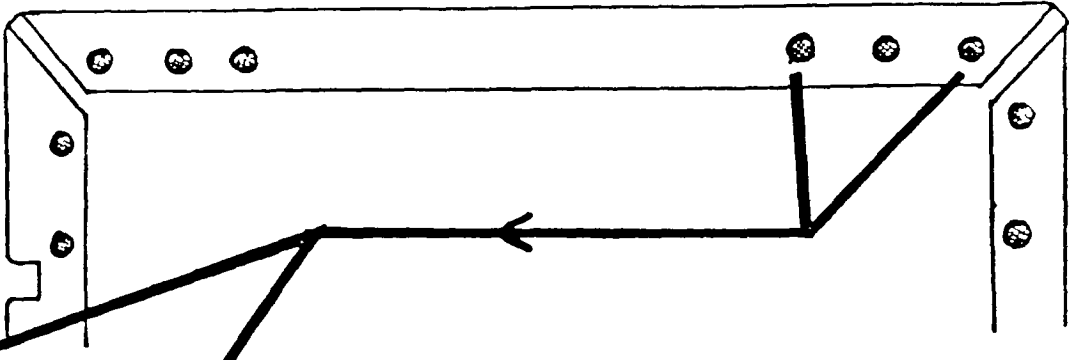
| | | |
|------------------------------|------------------------------|------------------------------|
| 10 RANDOMIZE | 160 GOSUB 180 | 8383854") |
| 20 GOSUB 250 | 170 GOTO 120 | 280 CALL CHAR(153,"0010FE383 |
| 30 PRINT "BREAKDANCING!!!" | 180 CALL VCHAR(12,10,BD-(INT | 84482") |
| 40 PRINT "HUMAN OR COMPUTER | <RND*2))+1) | 290 CALL CHAR(154,"8090FC3A3 |
| CONTROL???" | 190 CALL VCHAR(12,12,BD-(INT | 9484808") |
| 50 INPUT CON\$ | <RND*2))+1) | 300 CALL CHAR(155,"000000000 |
| 55 CALL CLEAR | 200 CALL VCHAR(12,16,BD) | 0847936") |
| 60 IF CON\$="HUMAN" THEN 120 | 210 CALL VCHAR(12,20,BD+(INT | 310 CALL CHAR(156,"00107CBA7 |
| 70 BD=INT(RND*5)+153 | (RND*2))+1) | C281808") |
| 80 CALL KEY(0,W,E) | 220 CALL VCHAR(12,22,BD+(INT | 320 CALL CHAR(157,"142424783 |
| 90 IF E=1 THEN 120 | <RND*2))+1) | 8") |
| 100 GOSUB 180 | 230 RETURN | 330 CALL CHAR(158,"41493E1C1 |
| 110 GOTO 70 | 240 GOTO 120 | C2214") |
| 120 CALL KEY(0,BD,N) | 250 REM CHARACTERS | 340 CALL CHAR(159,"40281E1D1 |
| 130 IF N=0 THEN 120 | 260 CALL CHAR(151,"000000000 | 41414") |
| 140 IF BD=32 THEN 70 | 4884438") | 350 RETURN |
| 150 BD=BD+102 | 270 CALL CHAR(152,"000082443 | |

The following page appeared in the MAR86 issue of the Manners Newsletter.

FLOPPY DISK IDENTIFICATION

By using the following chart, you can identify the manufacturer of most disks by how the disk jacket is sealed. This can be very helpful in determining the origin of generic disks.

IDENTIFICATION OF 5 1/4" DISKS



| SEAL | COMPANY | COMMENTS |
|------------------|--------------|---------------------------|
| ⊖ ⊕ ⊖ | MAXELL | |
| COMPLETE SEAL | MEMOREX | ALSO ALBINAR (BEST Co.) |
| ⊕ ⊕ | VERBATIM | |
| ● ● | NASHUA | 6 dots down each side |
| ■ ■ | BASF | |
| | ELEPHANT | |
| ———— | 3M | 2 BARS DOWN EACH SIDE |
| ■ ■ ■ ■ ■ ■ ■ ■ | WABASH | 6 SETS OF 8 SQS. PER SIDE |
| ■ ■ ■ ■ | FUJI | |
| ■ ■ ■ | CONTROL DATA | STORAGE MASTER |
| ———— | SYNCOM | 3 BARS DOWN EACH SIDE |
| ● ● | CERTRON | 8 DOTS DOWN EACH SIDE |
| ⊕ ⊕ | BONUS | |
| — — | | |
| | | |
| !!!!!!!!!!!!!!!! | | |
| | | |
| — — — — | | |
| ⊕ ⊕ | | |


```

160 DIM Q(20):: FOR X=1 TO 2
0 :: Q(X)=X :: NEXT X
170 M=0
180 FOR X=1 TO J-1 :: IF A$(Q(X),K)<A$(Q(X+1),K) THEN 21
0
190 M=-1
200 T=Q(X):: Q(X)=Q(X+1):: Q(X+1)=T
210 NEXT X
220 IF M THEN 170
230 FOR X=1 TO 20 :: FOR L=1 TO 4 :: PRINT A$(Q(X),L); "
";: NEXT L :: PRINT :: NEXT X :: GOTO 140

```

Did you ever need a routine that would accept either a string or a numeric value? Try this -

```

100 M=0 :: ON ERROR 110 :: ACCEPT M$ :: N=VAL(M$):: GOTO 120
110 ON ERROR STOP :: RETURN 120
120 ON (N=0)+2 GOTO 130,140
130 PRINT M$ :: GOTO 100
140 PRINT N :: GOTO 100

```

A useful tip from Stephen Shaw in England - if you have a long program which will run only in Basic, and which will load from disk with CALL FILES(1) but runs out of memory when you try to run it; and if you have the MiniMemory module -

Insert MiniMemory module, select Basic, enter CALL FILES(1), Enter NEW, enter OLD DSK1.(filename). When loaded, enter SAVE EXPNEM2. When SAVED, enter CALL LOAD(-31000,63,255), enter NEW, enter OLD EXPNEM2, and enter RUN. That is still a lot faster than loading a long program from tape!

Another reason for never using the default mode of so-called UPDATE when opening a file (without specifying INPUT or OUTPUT) is that you will get an I/O ERROR #1 if the file is write-protected.

Has anyone found a way to go from Extended Basic to Basic without losing the program in memory, or at least fouling it up? CALL LOAD(-32116,4) has been published in many newsletters as a way to do this, but has anyone actually made it work?

If you are printing out of TI-Writer Editor, finish your letter with CTRL U, SHIFT L, CTRL U and when it is printed the paper will automatically feed to the top of the next sheet.

To make a note to yourself while programming, just type ! and whatever you want to make note of, then LIST "PID":1, and then type ! and enter to delete the line.

TI-Writer puts an extra space after every period that is followed by a space. If you don't want this extra space after abbreviations such as "Mr." or "St.", use a caret sign ^ instead of a space after the period, Mr.^Jones. But TI-Writer puts only one space after ? or ! so if you want two, put a caret after the symbol !^

One of the very best tips for this month comes from Paul A. Meadows, in the September 85 newsletter of T.I.N.S. (Nova Scotia, Canada) -

How to print up to 132 characters in a line (condensed print, of course) out of TI-Writer! Just prepare your file as usual but in line 0001 put formatter commands such as .LM 10;RM 132; IN +5;FI;AD. The Fill and Adjust are necessary, the indent is up to you, as are the left and right margins - but notice that right margin set way over at 132? Now, instead of saving the

file with SF, type PF and then C DSK1.(filename) to print to the disk. This not only strips out the control C characters, it also erases the TI-Writer tab line that was applied to the last line of the file.

So now, with your printer opened and initialized for condensed print, go into the TI-Writer formatter mode and print your file!

I have made the following changes to my working copy of the Tigercub Menuloader. This sets up my Gemini printer to skip over the perforations and print full page width in elite print with a wide left margin for ring-binder punching. Other printers may need changes in these codes.

```

620 DISPLAY AT(12,1)ERASE ALL:
L:"PRINTER? PID" :: ACCEPT A
T(12,1)SIZE(-18):P$ :: GOSUB
B 895 :: PP=3
840 DISPLAY AT(24,1):"PRINTE
R NAME? PID" :: ACCEPT AT(24
,15)SIZE(-14):PP$ :: GOSUB B
95 :: PRINT @2:SEG$(D$,1,4)&
" - Diskname= "&N$
895 OPEN #3:P$,VARIABLE 132
:: PRINT #3:CHR$(27);"B";CHR
$(27);CHR$(27);"M";CHR$(18);C
HR$(27);"N";CHR$(6):: RETURN

```

I always keep a backup of everything, on the flipped side of another disk, and I often want to verify that the backup has everything that is on the master, and vice versa.

```

100 DISPLAY AT(3,6)ERASE ALL
:"TIGERCUB DOUBLECAT": : " To
compare the contents of": :
"a disk with a backup." !by
Jim Peterson
110 DISPLAY AT(12,1):"INSERT
MASTER DISK": : "PRESS ENTER
"
120 CALL KEY(0,K,S):: IF S=0
THEN 120
130 DATA DF,DV,IF,IV,P
140 RESTORE :: FOR I=1 TO 5
:: READ T$(I):: NEXT I
150 DIM F$(127):: OPEN #1:"D

```

```

SK1.",INPUT ,RELATIVE,INTERN
AL :: INPUT #1:A$,J,K :: F
$(0)=A$&" "&STR$(K)
160 X=X+1 :: INPUT #1:F$(X),
I,J,K :: IF F$(X)="" THEN 17
0 :: F$(X)=F$(X)&" "&T$(ABS(
I)): GOTO 160
170 X=X-1 :: CLOSE #1 :: DIS
PLAY AT(12,1)ERASE ALL:"REMO
VE MASTER DISK": : "INSERT BA
CKUP DISK": : "PRESS ENTER"
180 CALL KEY(0,K,S):: IF S=0
THEN 180
190 OPEN #1:"DSK1.",INPUT ,R
ELATIVE,INTERNAL :: INPUT #1
:A$,J,K :: DISPLAY AT(1,1)
ERASE ALL:F$(0):: DISPLAY A
T(1,15):A$&" "&STR$(K);
200 Y=Y+1 :: R=R+1 :: GOSUB
290 :: INPUT #1:A$,I,J,K ::
IF A$="" THEN 260 :: K$=A$&"
"&T$(ABS(I))
210 IF K$=F$(Y) THEN DISPLAY
AT(R+1,1):F$(Y):: DISPLAY A
T(R+1,15):K$:: GOTO 250
220 IF K$(F$(Y)) THEN DISPLAY
AT(R+1,15):K$:: Y=Y-1 :: GO
TO 250
230 DISPLAY AT(R+1,1):F$(Y);
:: R=R+1 :: GOSUB 290 :: Y=Y
+1
240 IF K$=F$(Y) THEN 210 ELSE
IF K$(F$(Y)) THEN 220 ELSE IF
Y<X THEN 230 ELSE DISPLAY A
T(R,15):K$;
250 GOTO 200
260 IF Y>X THEN 280
270 R=R+1 :: GOSUB 290 :: FO
R J=Y TO X :: DISPLAY AT(R,1
):F$(J):: R=R+1 :: GOSUB 290
:: NEXT J
280 DISPLAY AT(24,1):" P
RESS ANY KEY" :: CALL KEY(0,
K,S):: IF S=0 THEN 280 ELSE
CLOSE #1 :: END
290 IF R<23 THEN RETURN
300 DISPLAY AT(24,1):"PRESS
ANY KEY" :: DISPLAY AT(24,1)
:" " :: CALL KEY(0,K,S):: IF
S=0 THEN 300
310 CALL CLEAR :: R=1 :: RET
URN

```

And that is just about

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   *  
*****
```