

PRESIDENT'S NOTES...by Jay Seaberg

No warmth, no sun, no snow, no fun...November! It's almost time to kick off the holiday season again. Your computer can be an integral part of this time of year.

We have programs for mailing lists (so you don't forget Cousin Ernie again this year), computer Christmas cards with both graphics and music, software to design and print your own cards, and bookkeeping software to keep track of how much you spend. All of these programs and more are available at every meeting or through Jess Jolly or myself.

More of our members have been taking advantage of the software library. They have discovered the wide variety of software available to them on request.

We have an interesting agenda lined up for the next meeting, which will be November 14, 6:30 pm at the Library. There will be several printers demonstrated, including a color printer, 40 and 80-column thermal printers, and a monochrome dot-matrix printer. We will have the software available to do both text and graphics printing. These demo's may be just the ticket to help you decide what to ask Santa for.

One of our members attended the TI-Fest in Chicago, and he will give a report of what he saw. The 99/4A is definitely not dead.

Please try to attend the next meeting. We have shortened the business portion to allow more time for demonstrations, library access, and general interest discussions.

As usual, this month's newsletter is chock full of information of interest to our members. We have an excellent selection of software listings from our corresponding groups. Take the time to enter these programs, you may be surprised at how useful they can be.

THIS IS YOUR GROUP, USE IT!

INSIDE...

TIPS FROM THE TIGERCUB by Jim Peterson.

A SIMPLE MAIL LIST by George F. Steffen, LA 99'ers Users Group.

MULTI-COLUMN PRINTING by George F. Steffen, LA 99'ers Users Group

KEYBOARD CONVERSION by Tom Freeman, LA 99'ers Users Group newsletter.

SPREADSHEET PROGRAM by Greg Sears, Edmonton 99'er Computer Society.


```

ING 250:NN+3
470 DISPLAY AT(X+6,1):" C
hoice?" :: ACCEPT AT(X+6,16)
SIZE(-3)VALIDATE(DIGIT):K
480 IF FLAG=1 THEN 500
490 IF K=NN+2 THEN 840 ELSE
IF K=NN+3 THEN CLOSE #1 :: N
N=0 :: GOTO 190
500 IF K<>NN AND K<>NN+1 THE
N 590
510 IF K=NN THEN CALL CLEAR
:: CLOSE #1 :: END
520 DISPLAY AT(X+5,12)SIZE(1
2):" #?" :: ACCEPT AT(X+5,15
)SIZE(2)VALIDATE(DIGIT):KD :
: IF KD<1 OR KD>NN THEN 520
530 IF V(KD,1)>0 THEN 550
540 FOR J=1 TO 10 :: DISPLAY
AT(11,1):" " : " PROTECTED -
CANNOT DELETE": " " :: DISPL
AY AT(12,1):" " :: NEXT J ::
GOTO 570
550 DISPLAY AT(X+6,1)SIZE(27
)BEEP:" Verify - Delete ";P6
$(KD);"? " :: DISPLAY AT(X+6,
28)SIZE(1):"Y" :: ACCEPT AT(
X+6,28)SIZE(-1)VALIDATE("YM
"):Q# :: IF Q#<>"Y" THEN 570
560 DELETE D#&P6$(KD)
570 CLOSE #1
580 CALL VCHAR(1,3,32,672)::
NN=0 :: X=0 :: FLAG=0 :: GO
TO 260
590 IF K<1 OR K>127 OR LEN(P
6$(K))=0 THEN 430
600 IF ABS(V(K,1))=5 OR ABS(
V(K,1))=4 AND V(K,2)=254 THE
M 640
610 DISPLAY AT(12,1)ERASE AL
L:"Print to ? S": "(P)rinte
r?":(S)creen?" :: ACCEPT AT
(12,12)SIZE(-1)VALIDATE("PS
"):Q# :: IF Q#="S" THEN PP=0
:: GOTO 630
620 DISPLAY AT(12,1)ERASE AL
L:"PRINTER? P10" :: ACCEPT A
T(12,10)SIZE(-10):P# :: OPEN
#3:P# :: PP=3
630 CALL CLEAR :: CALL SCREE
N(16):: ON ABS(V(K,1))GOTO 6
80,690,750,760
640 CLOSE #1 :: IF SEG$(P6$(
K),LEN(P6$(K)),1)="" THEN D
ISPLAY AT(12,1)ERASE ALL:"RE
TURN TO BASIC AND LOAD BY:"
TYPING OLD ";D#&P6$(K):: STO
P
650 CALL PEEK(-31952,A,B)::
CALL PEEK(A#256+B-65534,A,B)
:: C=A#256+B-65534 :: A#D#&

```

```

P6$(K):: CALL LOAD(C,LEN(A#
)
660 FOR I=1 TO LEN(A#):: CAL
L LOAD(C+1,ASC(SEG$(A#,1,1))
):: NEXT I :: CALL LOAD(C+I,
0)
670 CALL VCHAR(1,3,32,672)::
CALL SCREEN(0):: FOR S=0 TO
14 :: CALL COLOR(S,2,1):: N
EXT S :: DISPLAY AT(12,2):"L
OADING ";A# :: GOTO 900
680 OPEN #2:D#&P6$(K),INPUT
,FIXED :: GOTO 700
690 OPEN #2:D#&P6$(K),INPUT
700 LINUT #2:M# :: PRINT #P
P:M# :: IF EOF(2)THEN 730
710 CALL KEY(0,K,S):: IF S=0
THEN 700
720 CALL KEY(0,K2,S2):: IF S
2<1 THEN 720 ELSE 700
730 CLOSE #1 :: CLOSE #2 ::
PRINT " >>>press any key<<
" :: IF Q#="P" THEN CLOSE #
3
740 CALL KEY(0,K,ST):: IF ST
<1 THEN 740 ELSE 580
750 OPEN #2:D#&P6$(K),INPUT
,INTERNAL,FIXED :: J=0 :: GO
TO 770
760 OPEN #2:D#&P6$(K),INPUT
,INTERNAL :: J=0
770 IF EOF(2)=1 THEN 730 ::
J=J+1 :: INPUT #2:M# :: IF L
EN(M#)=0 THEN 790
780 PRINT #PP:M# :: GOTO 820
790 FOR Y=1 TO 0 :: @0=ASC(S
EG$(M#,Y,1)): IF @0<32 OR @
0>127 THEN 810
800 NEXT Y :: GOTO 780
810 RESTORE #2 :: FOR X=1 TO
J-1 :: INPUT #2:M# :: NEXT
X :: INPUT #2:M :: PRINT #PP
:M
820 CALL KEY(0,K,S):: IF S=0
THEN 770
830 CALL KEY(0,K2,S2):: IF S
2<1 THEN 830 ELSE 770
840 DISPLAY AT(24,1):"PRINTE
R NAME? P10" :: ACCEPT AT(24
,15)SIZE(-14):PP# :: OPEN #2
:PP# :: PRINT #2:SEG$(D#,1,4
)&" - Diskname= "&N#
850 PRINT #2:RPT#("=",28):"A
vailabl=";350-VT;"Used=";VT
:RPT#("=",28)
860 PRINT #2:"FILENAME SIZE
TYPE":RPT#(" ",28)
870 FOR P=1 TO NN-1 :: PRINT
#2:P6$(P);TAB(15);V(P,3);TA
B(20);T$(ABS(V(P,1)));TAB(25

```

```

);V(P,2):: NEXT P :: CLOSE #
2
880 DISPLAY AT(12,3)ERASE AL
L:"(P) to print again": " (R
) to rescan": " (Q) to quit"
890 ACCEPT AT(15,4)VALIDATE(
"PQR")SIZE(-1)BEEP:Q# :: IF
Q#="P" THEN 840 :: CLOSE #1
:: NN=0 :: IF Q#="R" THEN 19
0 ELSE END
900 RUN "DSKX.1234567890"

```

This version turns off the Quit key, restarts itself rather than crashing on an I/O error, and has pre-scan for faster start-up. It displays disk name, sectors available and sectors presumably used - it also totals up actual sectors used and sounds a warning if any sectors are not accounted for.

It lists up to 127 programs and files by number, filename, number of sectors, program or file type, file record length, and write-protection. It will stop for menu selection on any keypress or at the end of each screen, continuing on Enter. It will load and run any program that can run from Extended Basic, displaying its filename while loading. If the filename ends in an asterisk, it will warn you to return to Basic. It will delete any unprotected program or file, after first requiring verification by filename, or will inform you if the file is protected. It will read any readable file, including internal numeric, and list it to screen or printer. It will dump a catalog of the disk to your printer, and it will offer the option of quitting or rescanning the disk or another disk. And it's free, I don't even want a freeware donation - but I would appreciate if you would take a look at my catalog and see if,

somewhere among those 140 programs, there might be something you would be willing to pay \$3 for? The Menu Loader is included as a bonus on every disk I sell!

```

100 CALL CLEAR :: RANDOMIZE
:: DISPLAY AT(3,4):"TIGERCUB
MATH PUZZLE"
110 DISPLAY AT(6,1):"Insert
+, -, * (multiply) DR / (div
ide) between the digits
to equal the total": "Type
0 to give up"
120 DISPLAY AT(12,1):"Level
1 or 2?" :: ACCEPT AT(12,15)
VALIDATE("12"):L#
130 T,X=INT(9#RND+1):: M#ST
R$(X):: Z#M#&" "
140 FOR J=1 TO 4 :: Y(J)=INT
(9#RND+1):: Z=INT(4#RND+1)::
ON Z 60SUB 240,250,260,270
:: Z#Z#STR$(Y(J))&" " :: N
EXT J
150 IF L#="1" AND T<>INT(T)T
HEN 130 :: Z#Z#&"#&STR$(T)
160 DISPLAY AT(12,1):Z# :: D
ISPLAY AT(18,1):" " :: DISPL
AY AT(20,1):" " :: DISPLAY A
T(22,1):" "
170 P=2 :: FOR J=1 TO 4 :: A
CCEPT AT(12,P)VALIDATE("Q+=#
/")SIZE(1):S#
180 IF S#="Q" THEN 200 ELSE
IF S#="+" THEN X=X+Y(J)ELSE
IF S#="-" THEN X=X-Y(J)ELSE
IF S#="*" THEN X=X*Y(J)ELSE
X=X/Y(J)
190 P=P+2 :: NEXT J :: IF X=
T THEN 230 :: DISPLAY AT(18,
1):"WRONG!"
200 DISPLAY AT(20,1):"ANSWER
IS ";M#
210 DISPLAY AT(22,1):"PRESS
ANY KEY"
220 CALL KEY(0,K,ST):: IF ST
<1 THEN 220 :: GOTO 130
230 DISPLAY AT(18,1):"RIGHT!
" :: GOTO 210
240 M#M#&"#&STR$(Y(J)):: T
=T+Y(J):: RETURN
250 M#M#&"-&STR$(Y(J)):: T
=T-Y(J):: RETURN
260 M#M#&"*&STR$(Y(J)):: T
=T*Y(J):: RETURN
270 M#M#&"/&STR$(Y(J)):: T
=T/Y(J):: RETURN

```

Enjoy!

Jim Peterson

A SIMPLE MAIL LIST by George F. Steffen

This program is the result of a request for a very simple mail list program. It was designed to use the TI Writer for entry of data. To avoid the necessity of switching modules, the program has been written entirely in TI Console Basic.

The program includes commands to make it work on an Epson Printer. FL\$ sets Form Length with the next character being the length. CT\$(1) sets characters to normal width, emphasized and double strike. CT\$(2) sets characters to compressed. RST\$ returns the characters and form length to normal. No provision has been made for changing the line feed, so it will remain at the normal six lines per inch.

The program has provision for entering label height and width and single or double line feeds. Of course, with one inch labels and double line feeds, you are restricted to three lines per label. The program will discard any portion of a line which will not fit on the size label you have entered.

With this program, you can print continuous labels from any number of different files. It will also print lists from these files. It will give you, on screen, the number of names in each list and the total. If you are listing, these figures will be placed at the end of the list.

The data for the labels or lists should be prepared using TI Writer. You should cancel WordWrap (Control O till you get a hollow cursor). The first line in each file--the file title--will not be printed. If you omit this title, the program will skip the first line of your first entry. Entries are separated by New Page (Control 9) commands. Because page length on the printer has been set to label height, this puts the next entry on the next label. If you wish to add any printer commands to your data file(s), you may place them between the title line and the New Page entry which follows it. You should remember that this will cause the command to be effective in both label and list printing.

The program includes a stop label printing feature so that entries may remain on the list, but labels will not be produced for them. This is handy for those whose dues are not current, for example, but whom you expect to catch up. If the first line after the New Page is "END", the program will stop printing labels, but will continue if it is printing lists.

For easy entry of data, I recommend you first enter common data--perhaps city, state and zip code or membership status--for one entry and then duplicate it as needed with the Copy command. Leaving blank lines for the distinctive information also helps to speed entry. You can then go through and complete the information as necessary.

Below is a sample of a list for use by this program. "PC" indicates New Page and Carriage Return.

Membership Test List

PC
Jane Doe
123 E. Main St.
Anytown, CA 98765
PC
John Q. Public
P. O. Box 75
Los Angeles, CA 90001
PC
END END END END
Members who have not renewed
PC
Richard Roe
3298 Elm St.
Old Town, NY 12345
PC

NOTE! If you wish to use commas, leading spaces or trailing spaces within the addresses, the statement in Line 660 must be changed to LINPUT instead of INPUT and the program run in Extended Basic rather than Basic.

It will take practice to learn to position the labels so that the text will be entirely within one label. I suggest drawing lines on some ordinary computer printer paper to represent the label separations and try centering the text between the lines. You should turn off the printer while lining up the labels and then turn it on before starting to print.

The program is available in the LA 99er Computer Group Library under the name LISTLABEL. I can not give a number because our chief librarian is on vacation.

```
100 REM LISTLABEL
110 REM SEE ACCOMPANYING TI
WRITER FILE WHICH GIVES INST
RUCTIONS.
120 DATA PRINT LABELS,PRINT
LISTS,END SESSION
130 FL$=CHR$(27)&"C"
140 CT$(1)=CHR$(27)&"E"&CHR$(
27)&"G"
150 CT$(2)=CHR$(15)
160 RST$=CHR$(18)&CHR$(20)&C
HR$(27)&"F"&CHR$(27)&"H"&FL$
&CHR$(66)
170 FOR I=1 TO 3
180 READ M$(I)
190 NEXT I
200 CALL CLEAR
210 PRINT TAB(8);"MAIL PRINT
ER"
220 FOR I=1 TO 3
230 PRINT :STR$(I);" " ;M$(
I)
240 NEXT I
250 PRINT
260 INPUT "CHOICE? " ;S
270 IF (S<1)+(S>3)THEN 260
280 ON S GOTO 290,440,990
290 PRINT
300 INPUT "LABEL HEIGHT (IN)
? " ;L
310 L=L*6
320 IF L=INT(L)THEN 350
330 PRINT "ERROR IN LABEL HE
IGHT."
340 GOTO 300
350 PRINT
360 INPUT "LABEL WIDTH (IN)?
" ;C
370 C=C*10
380 IF C<80 THEN 410
390 PRINT "ERROR IN LABEL WI
DTH."
400 GOTO 360
410 PRINT : "1. SINGLE LINEF
EED";"2. DOUBLE LINEFEED"
420 INPUT "CHOICE? " ;D
430 IF (D<1)+(D>2)THEN 420 E
LSE 460
440 L=66
450 C=132
460 PRINT
470 INPUT "FILE NAME FOR YOU
R PRINTER? " ;PS
480 OPEN #2:PS,DISPLAY ,VARI
ABLE C,OUTPUT
490 PRINT #2:FL$;CHR$(L);CT$(
S);
500 PRINT : "FILE(S) TO BE PR
INTED?";"NULL ENTRY WILL TER
MINATE"; :
510 I=1
520 INPUT STR$(I)&" " ;F$(I
)
530 IF F$(I)="" THEN 570
540 PRINT
550 I=I+1
560 GOTO 520
570 PRINT : :TAB(11);"PRINTI
NG"; :
580 L=0
590 FOR J=1 TO I-1
600 PRINT STR$(J);" " ;F$(J
);
610 E=0
620 K=0
630 OPEN #1:F$(J),DISPLAY ,V
ARIABLE 80,INPUT
640 INPUT #1:L$
650 IF EOF(1)THEN 740
660 INPUT #1:L$
670 IF ASC(L$)>127 THEN 740
680 IF ASC(L$)<>12 THEN 720
690 ON S GOSUB 860,880
700 K=K+1
710 GOTO 650
720 ON S GOSUB 900,970
730 IF E=0 THEN 650
740 K=K-1
750 L=L+K
760 IF S=1 THEN 780
770 PRINT #2: :TAB(7);K;"ENT
RIES";
780 PRINT K;"ENTRIES": :
790 CLOSE #1
800 NEXT J
810 PRINT :TAB(6);"PRINTING
COMPLETED";L;"TOTAL ENTRIES"
: :
820 IF S=1 THEN 840
830 PRINT #2: : :TAB(7);"Tot
al Entries";L;RST$
840 CLOSE #2
850 GOTO 220
860 PRINT #2:L$
870 RETURN
880 PRINT #2
890 RETURN
900 IF SEG$(L$,1,3)<>"END" T
HEN 930
910 E=1
920 RETURN
930 PRINT #2:L$
940 IF D=1 THEN 960
950 PRINT #2
960 RETURN
970 PRINT #2:L$;" " ;
980 RETURN
990 END
```

MULTI COLUMN PRINTING and VARIABLE LENGTH LISTING by George F. Steffen

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 length of the input text and the length 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. Both programs are available in the club library.

```

100 REM VARYLIST
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 FOLLOWING LINE
150 DATA 3,DSK,WDS,RO
160 CALL CLEAR :: PRINT TAB(11);"VARYLIST"
170 PRINT :: LINPUT "NAME OF INPUT PROGRAM LIST? ":IPS
180 PRINT :: LINPUT "NAME OF OUTPUT FILE          ":OFS :: IF OFS=IPS THEN PRINT "INPUT AND OUTPUT NAMES
    MUST BE DIFFERENT!" :: GOTO 170
190 PRINT :: INPUT "WIDTH OF OUTPUT FILE? ":OW :: ODW=OW :: IF OW>79 THEN 220
200 READ N :: FOR I=1 TO N :: READ DNS :: IF SEGS(OFS,1,LEN(DNS))=DNS THEN ODW,I=80
210 NEXT I
220 OPEN #1:IPS,DISPLAY ,VARIABLE 80,INPUT :: OPEN #2:OFS,DISPLAY ,VARIABLE ODW,OUTPUT
230 FOR I=0 TO 9999 :: L1$=""
240 IF EOF(1)THEN I=I+10000 :: GOTO 250 ELSE LINPUT #1:L2$ :: 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 OW :: PRINT #2:SEGS(L1$,O,OW):: J=J+1 :: NEXT O :: NEXT I
260 CLOSE #1 :: CLOSE #2 :: PRINT :I-10000;"NUMBERED LINES":J;"OUTPUT LINES" :: END
    
```


LOADER PROGRAM TO LOAD SPRED SHEET PROGRAM WRITTEN BY GREG SEARS. THIS PROGRAM IS

FREEMWARE

BUT GREG ASKS THAT USERS MAKE A SMALL CONTRIBUTION TO EDMONTON 99'ER COMPUTER USERS SOCIETY,

P/O BOX 11983, EDMONTON, ALBERTA, CANADA

T5J-3L1

```

:TAB(4); "Load Greg's Spread Sheet"
-----
210 B$="Choice?" :: CALL B(B$,C):: IF C=80 THEN 250 :: IF C=83 THEN 230
-----
220 IF C=76 THEN CALL E(14,16):: RUN "DSK1.GS/SPREAD" :: ELSE 210
-----
230 CALL E(3,23):: A,B=1 :: D=0 :: RESTORE :: B=B+1 :: F OR B=3 TO 24 :: READ A$ :: C ALL D(A$,A,B):: IF B=23 THEN B$="Next..." :: D=D+1 :: CALL B(B$,C):: CALL E(3,23):: B=2 :: IF D=3 THEN 160
-----
240 NEXT B
-----
250 B$="Printer Ready?(Y/N)" :: CALL B(B$,C):: IF C=78 THEN 200 ELSE IF C<>89 THEN 250 :: CALL E(16,18)
-----
260 RESTORE 290 :: OPEN #1:"RS232.BA=4800"
-----
270 PRINT #1:TAB(31); "GRES'S SPREAD SHEET":TAB(31);RPT$( "A",19):: FOR E=1 TO 63 :: READ A$ :: PRINT #1:TAB(40-LEN(A$)/2);A$ :: NEXT E
-----
280 PRINT #1:CHR$(12):: CLOSE #1 :: GOTO 32767 0
-----
290 DATA "INSTRUCTIONS", "=====", "The [Inspect Disk] Section", "====="
-----
300 DATA "Catalog for Int/Ar 80 FILES."
-----
310 DATA "The [Load Data] Section", "=====", "ENTER filename to load FILE."
-----
320 DATA "The [Enter/Edit] Section", "=====", "First entry Promps a re...", "No. of Rows No. of Coils.", "Title Name Row Heading"
-----
330 DATA "Name Rows Name Columns", "In spread sheet section...", "You can change Row & Col. names."
-----
340 DATA "ENTER a value for the Cursor", "then FCTN-Arrow Keys move the Cursor up, down, left, or right."
    
```

```

350 DATA Press C[calculate R] return
-----
360 DATA "The [CALCULATE] Section", "=====", "Is a FOR-NEXT loop in Basic, Promps are 'Defaulted' to ease your operations in this Mode."
-----
370 DATA Answer the promps to o...," Operation(+,-,*,/,^)"
-----
380 DATA "Operate on Rows(Y/N)", "Operate on Cols(Y/N)", "Single Pair Oper.(Y/N)"
-----
390 DATA "ENTER Constant (v alue)", "Put result in Row No.", "Put result in Col.N o."
-----
400 DATA "From Row No.... (FOR)", "To Row No..... (NEXT)"
-----
410 DATA "From Col. No... (FOR)", "To Col. No..... (NEXT)"
-----
420 DATA "Use Fctn X to fill all Promps", "or ENTER to fill single Prompt.", "( Fctn= Calc Fctn=Return )"
-----
430 DATA "The [Save Data] Section", "=====", "filename-from Load 'Defaulted'", " "
-----
440 DATA " -can be changed.", "The Title can be changed."
-----
450 DATA "The [Print Data] Section", "=====", "Printer set Gemini-10X 'PID', Print with double space Rows."
-----
460 DATA "The [Clear Memory] Section", "=====", "Erases the FILE in Computer...", "Notice- save your data first."
-----
470 DATA " Press C[clear R]e turn ", "Program Notes", "====="
-----
480 DATA Program I/O ERROR protected, Dimensioned to 20-Rows 13-Col., Press Q To Quit Program.
-----
490 SUB A :: CALL CHAR(64, '
    
```

```

EB19DA1A19DB17E"):: FOR A=0 TO 14 :: CALL COLOR(A,16,5):: NEXT A :: SUBEND
-----
500 SUB B(A$,A):: DISPLAY AT(24,14-LEN(A$)/2)BEEP:A$
-----
510 CALL KEY(3,A,B):: CALL HCHAR(24,20,32):: CALL C(50):: IF B=0 THEN 520 ELSE CALL HCHAR(24,20,A):: CALL C(100):: CALL HCHAR(24,3,32,28):: SUBEXIT
-----
520 CALL HCHAR(24,20,95):: CALL C(50):: IF B=0 THEN 510
-----
530 SUBEND
-----
540 SUB C(A):: FOR B=1 TO A :: NEXT B :: SUBEND
-----
550 SUB D(A$,A,B):: C=INT((32-LEN(A$))/2):: FOR D=A TO L EN(A$): E=ASC(SEG$(A$,D,A)) :: CALL HCHAR(B,C+D,E):: NEXT D :: SUBEND
-----
560 SUB E(A,B):: FOR A=A TO B :: CALL HCHAR(A,1,32,31):: NEXT A :: SUBEND
-----
570 SUB F :: RESTORE 580 :: FOR A=97 TO 122 :: READ A$ :: CALL CHAR(A,A$):: NEXT A :: SUBEXIT
-----
580 DATA 0000007008384874,0040407844444478,000000384440438,0004043C4444443C
-----
590 DATA 00000038447C403C,0018242070202020,000004384438047C,0040407844444444
-----
600 DATA 0010003010101038,0008001808084830,0040404850704844,0030101010101038
-----
610 DATA 0000007854545454,0000005824242424,0000003844444438,0000007844784040
-----
620 DATA 0000003844544834,000000584404040,0000003C40380478,0010381010101408
-----
630 DATA 0000004848484824,000004444282810,0000004454545424,0000004428102844
-----
640 DATA 0000004424181060,000007C0810207C
-----
650 SUBEND
    
```

```

100 ! * LOAD SPREAD SHEET !
-----
110 ! * By Greg Sears !
-----
120 ! * Version 1.1 9/22/85
-----
130 ON BREAK NEXT :: CALL IN IT :: CALL LOAD(-31806,16)
-----
140 CALL SCREEN(5):: CALL CLEAR :: CALL A :: CALL F :: ON N WARNING NEXT
-----
150 A$="Greg's Spread Sheet" :: A,B=1 :: CALL D(A$,A,B) :: B=2 :: A$="*****" :: CALL D(A$,A,B)
-----
160 A$="FREE/WARE Program" :: B=4 :: CALL D(A$,A,B):: A$="By Greg Sears 1985" :: B=6 :: CALL D(A$,A,B):: A$="For T.I. User's Group" :: B=8 :: CALL D(A$,A,B)
-----
170 A$="Options" :: B=10 :: CALL D(A$,A,B):: B=11 :: A$="*****" :: CALL D(A$,A,B)
-----
180 DISPLAY AT(14,4)BEEP:"If you like this Program":TAB(B); "Send. $5.00 to...":TAB(4); "Edmonton T.I. Users Group":TAB(10); "P.O.Box 11983":TAB(8); "Edmonton Alberta."
-----
190 FOR DELAY=1 TO 1500 :: NEXT DELAY
-----
200 DISPLAY AT(14,4):"Print The Instructions":TAB(4); "Screen The Instructions":
    
```


SPREAD SHEET PROGRAM

WRITTEN BY GREG SEARS

THIS PROGRAM IS CERTIFIED AS

FREEMWARE

BUT GREG ASKS THAT USERS
MAKE A SMALL DONATION TO THE

EDMONTON 99'ER COMPUTER
USERS SOCIETY,

P/O BOX 11983,
EDMONTON, ALBERTA, CANADA
T5J-3L1

.....THANKYOU

100 ! * GREG SPREAD SHEET *

110 ! * By Greg Sears *

120 ! *Version 1.1 9/22/85*

130 ON BREAK NEXT :: CALL CL
EAR :: CALL PEEK(8198,A):: I
F A<170 THEN CALL INIT :: C
ALL LOAD(-31806,16)

140 CALL LOAD(8196,63,248)::
CALL LOAD(16376,67,85,82,83
,79,82,48,8):: CALL LOAD(122
88,0,0,0,0,0,0,252)

150 CALL LOAD(12296,2,0,3,24
0,2,1,48,0,2,2,0,8,4,32,32,3
6,4,91):: CALL LINK("CURSOR"
):: B=13 :: C=20 :: ON WARNIN
G NEXT :: GOTO 190

160 DIM A\$(20,13),D(20,13),B
\$(4),E(11)

170 D(,()),A\$(,()),E(,),B\$(,)
F,G,H,I,J,K,C%,L,M,D%,N,O,P,
Q,R,S,E%,T,F%,B%,C,U,H%,I%,
V,W,X,Y,Z,AA,BA,CA,DA,EA,J\$(
,),FA,GA,HA,IA

180 CALL CHAR :: CALL PEEK ::
CALL LOAD :: GOSUB :: CALL
CLEAR :: CALL COLOR :: CALL
ERR :: CALL SOUND :: CALL H
CHAR :: CALL VCHAR :: CALL K
EY :: CALL SCREEN

190 !3P-

200 M=16 :: BA=5 :: GOSUB 15
30 :: G\$="Greg's Spread-shee
t" :: GOSUB 1540 :: H\$="Titl
e" *%I% :: DISPLAY AT(4,14-L

EN(H\$)/2):H\$:: ON ERROR 155
0

210 DISPLAY AT(6,12):"MENU":
TAB(12);RPT\$(=" ",4) :TAB(6)
;"1) Inspect Disk" :TAB(6);
;"2) Load Data" :TAB(6);"3)
Enter/Edit Data" :TAB(6);"4
) Calculate Mode"

220 DISPLAY AT(17,6):"5) Sav
e Data" :TAB(6);"6)-Print D
ata" :TAB(6);"7) Clear Memo
ry" : :TAB(9);"Press Choice

230 CALL SOUND(200,550,0)

240 CALL KEY(3,U,BA):: IF BA
=0 THEN 240 :: IF U=81 THEN
1580 ELSE U=U-48

250 IF U<1 OR U>7 THEN 230

260 DISPLAY AT(24,5):U:: ON
U GOSUB 280,380,410,710,134
0,1380,1480

270 GOTO 200

280 T=1 :: L=6 :: J\$(1)="Dis
/Fix" :: J\$(2)="Dis/Var" ::
J\$(3)="Int/Fix" :: J\$(4)="In
t/Var" :: J\$(5)="Program" ::
BA=13 :: GOSUB 1530 :: T=IN
T(T)

290 OPEN #1:"DSK"&STR\$(T)&".
",INPUT_RELATIVE,INTERNAL ::
INPUT #1:F%,0,0,P :: DISPL
AY AT(2,2):"DSK"&STR\$(T):"-D
ISKNAME: ";F%:" Available="
;P;"Used=";0-P

300 DISPLAY AT(4,1):RPT\$("-"
,28):"FILENAME SIZE TYPE
P":RPT\$("-",28):: FOR S
=1 TO 127 :: L=L+1

310 INPUT #1:F%,T,O,P :: IF
LEN(F%)=0 THEN 360

320 IF ABS(T)<>4 THEN 350

330 IF L=22 THEN G\$="Press C
ontinue" :: GOSUB 1510 :: I
F P<>67 THEN 330 :: L=7

340 IF T>0 THEN DISPLAY AT(L
,1):F%;TAB(12);O:TAB(17);J\$(
ABS(T));P:"" ELSE DISPLAY AT
(L,1):F%;TAB(12);O:TAB(17);J
\$(ABS(T));P:TAB(28);"Y:""

350 NEXT S

360 CLOSE #1

370 G\$="Press L)oad R)eturn"
:: GOSUB 1510 :: IF P=82 TH
EN P,BA=0 :: RETURN :: ELSE
IF P<>76 THEN 370

380 M=2 :: BA=12 :: GOSUB 15

30 :: G\$="Load Data" :: GOSU
B 1540 :: DISPLAY AT(23,2)BE
EP:"File Name?":D\$:: ACCEP
T AT(23,13)SIZE(-10):D\$:: I
F D\$="" THEN 200

390 G\$="Loading Data" :: GOS
UB 1540 :: OPEN #1:"DSK1."&D
\$,INTERNAL,INPUT_VARIABLE ::
M=M+1 :: INPUT #1:I%,C,B ::
FOR AA=0 TO C :: FOR F=0 TO
O B :: INPUT #1:A\$(AA,F),D(A
A,F):: NEXT F :: NEXT AA

400 CLOSE #1 :: DA,Z,F=1 ::
IA=0 :: GA=12 :: HA=21 :: H=
H+1 :: M=16 :: RETURN

410 BA=13 :: GOSUB 1530 :: I
F DA=1 THEN 470 :: G\$="Enter
/Edit Data" :: GOSUB 1540 ::
DISPLAY AT(4,6):"Making a
New File."

420 DISPLAY AT(6,2)BEEP:"Num
ber of Rows: ";C:TAB(2);
"Number of Columns":B :: AC
CEPT AT(6,2)SIZE(-2)VALIDAT
E(DIGIT):C :: ACCEPT AT(8,21
)SIZE(-2)VALIDATE(DIGIT):B

430 DISPLAY AT(16,3):"To cha
nge the number of: " Rows
or Cols. Press"ENTER"::
:" Report Titles" :: ACCEP
T AT(22,15)SIZE(14)BEEP:I%
:: IF I%="" THEN 420 :: IF I%
="RETURN" THEN 200 :: DA=1
:: GOSUB 1530 :: GOSUB 1640

440 GOSUB 1540 :: DISPLAY AT
(4,2)BEEP:"Row Headings:" ::
ACCEPT AT(4,14)SIZE(8):A\$(10,
0)

450 FOR AA=1 TO C :: DISPLAY
AT(AA+4,2):"Row":AA:TAB(9):
"Name:" :: ACCEPT AT(AA+4,14
)SIZE(8):A\$(AA,0):: NEXT AA
:: Z=1 :: CALL CLEAR :: GOSU
B 1540

460 FOR F=1 TO B :: DISPLAY
AT(F+4,2):"Column";F:TAB(12)
:"Name:" :: ACCEPT AT(F+4,17
)SIZE(8):A\$(0,F):: NEXT F ::
F=1 :: GA=12 :: HA=21

470 DISPLAY AT(1,6)ERASE ALL
:"Enter/Edit Data":TAB(6);RP
T\$("-",15):: DISPLAY AT(3,1)
:USING 540:A\$(0,0),A\$(0,F),A
\$(0,F+1)

480 FOR AA=Z TO Z+9 :: DISPL
AY AT((AA-IA)*2+3,1):USING 5
50:A\$(AA,0),D(AA,F),D(AA,F+1
):: NEXT AA :: AA=Z

490 CALL HCHAR((AA-IA)*2+3,G
A,32):: CALL HCHAR((AA-IA)*2
+3,HA,32):: F=F+Y :: DISPLAY
AT(24,4):" " :: ACCEPT AT((
AA-IA)*2+3,GA-1)SIZE(-8)BEEP

VALIDATE(NUMERIC):D(AA,F)::
F=F-Y

500 DISPLAY AT(24,4):"Press
C)alculate R)eturn" :: GOSUB
1640 :: IF BA=0 THEN 500

510 IF P=8 THEN 560 :: IF P=
9 THEN 600 :: IF P=10 THEN 6
30 :: IF P=11 THEN 660 :: IF
P=6 THEN 470

520 IF P=82 OR P=114 THEN 20
0 :: IF P=67 OR P=99 THEN 80
SUB 710 ELSE 500

530 GOTO 500

540 IMAGE "#####

#####"

550 IMAGE "#####

#####"

560 IF GA=2 OR AA=0 THEN 500

570 CALL HCHAR((AA-IA)*2+3,G
A,32):: CALL HCHAR((AA-IA)*2
+3,HA,32):: IF GA=21 THEN GA
=12 :: HA=21 :: Y=0 :: GOTO
490

580 IF GA=12 AND F>1 THEN F=
F-1 :: Y=0 :: GOTO 470

590 IF GA=12 AND F=1 THEN F=
0 :: GA=2 :: HA=11 :: Y=0 ::
DISPLAY AT(24,4):" " :: GOTO
690

600 IF AA=0 THEN 500 :: IF G
A=21 AND F+1=B THEN 500 :: I
F GA=21 AND F+1<B THEN F=F+1
:: Y=1 :: GOTO 470

610 CALL HCHAR((AA-IA)*2+3,G
A,32):: CALL HCHAR((AA-IA)*2
+3,HA,32):: IF GA=2 THEN F=1
:: GA=12 :: HA=21 :: GOTO 4
90

620 IF GA=12 THEN GA=21 :: H
A=30 :: Y=1 :: GOTO 490

630 IF AA=C OR GA=2 THEN 500

640 CALL HCHAR((AA-IA)*2+3,G
A,32):: CALL HCHAR((AA-IA)*2
+3,HA,32):: AA=AA+1 :: IF AA
=11 THEN Z=11 :: IA=10 :: 60
TO 470

650 GOTO 490

660 IF AA=0 OR GA=2 THEN 500
:: DISPLAY AT(24,4):" "

670 CALL HCHAR((AA-IA)*2+3,G
A,32):: CALL HCHAR((AA-IA)*2
+3,HA,32):: AA=AA-1 :: IF AA
=0 THEN 690 :: IF AA=10 THEN
Z=1 :: IA=0 :: GOTO 470

```

680 GOTO 490
-----
690 CALL HCHAR((AA-IA)*2+3,B
A,91):: CALL HCHAR((AA-IA)*2
+3,HA,93):: F=F+Y :: ACCEPT
AT((AA-IA)*2+3,BA-1)SIZE(-8)
BEEP:A$(AA,F):: F=F-Y :: IF
F=0 THEN F=1 :: GA=12 :: HA=
21
-----
700 RETURN
-----
710 BA=13 :: GOSUB 1530 :: I
F DAC1 THEN 200 :: DISPLAY A
T(1,6):"CALCULATE NODE":TAB(
6);RPT$(" ",14)
-----
720 DISPLAY AT(3,2):"Operati
on (+,-,*,/,^)?":TAB(26);B$(
1):: TAB(2):"Operate on Row
s(Y/N)?":TAB(26);B$(2):: :TA
B(2):"Operate on Columns(Y/N
)?":B$(3);
-----
730 DISPLAY AT(9,2):"Single
Pair Oper. (Y/N)?":TAB(26);B$(
4):: TAB(2):"ENTER Constan
t":TAB(17);E(4):: TAB(2):"
Put result in Row No.":TAB(
24);E(5);
-----
740 DISPLAY AT(15,2):"Put re
sult in Col. No.":E(6):: :TA
B(2):"From Row No. ....
":E(7):: TAB(2):"To Row No.
.....":E(8);
-----
750 DISPLAY AT(21,2):"From C
olumn No. ....":E(9):: :TA
B(2):"To Column No. ....
":E(10):: Q=1
-----
760 DISPLAY AT(24,1):"" :: I
F Q<5 THEN ACCEPT AT(Q*2+1,2
6)VALIDATE("+,-,*,/,^,Y,N")S
IZE(-1)BEEP:B$(Q):: GOTO 790
-----
770 IF Q=5 THEN ACCEPT AT(11
,18)SIZE(-5)VALIDATE(NUMERIC
)BEEP:E(4):: GOTO 790
-----
780 IF Q>5 THEN ACCEPT AT(Q*
2+1,25)VALIDATE(DIGIT)SIZE(-
2)BEEP:E(Q-1)
-----
790 G$=" (Fctn6=Calc. Fctn5=
Return)" :: GOSUB 1510 :: IF
P=10 AND Q<11 THEN Q=Q+1 ::
GOTO 760 :: IF P=10 AND Q>1
1 THEN 790
-----
800 IF P=11 AND Q>1 THEN Q=Q
-1 :: GOTO 760
-----
810 IF P=12 THEN 820 :: IF P
=14 THEN IA=0 :: Z,F,DA=1 ::
GOTO 200 ELSE GOTO 790
-----
820 AA=E(7):: F=E(9):: DISPL
AY AT(24,14)BEEP:" Calculati
ng " :: IF B$(1)="" THEN 85
0 ELSE IF B$(1)="-" THEN 960
-----
830 IF B$(1)="" THEN 1070 ::
: IF B$(1)="/" THEN 1180 ::
IF B$(1)="" THEN 1290 ELSE
790
-----
840 GOSUB 1370 :: IA=0 :: Z,
F,DA=1 :: GOSUB 410 :: RETUR
N
-----
850 IF B$(3)="Y" THEN 910 ::
IF B$(4)="Y" THEN 870 :: IF
E(4)<>0 THEN 900
-----
860 FOR AA=E(7) TO E(8):: FOR
F=E(9) TO E(10):: D(E(5),F)=
D(E(5),F)+D(AA,F):: NEXT F :
: NEXT AA :: GOTO 840
-----
870 IF E(4)<>0 THEN 890
-----
880 FOR F=E(9) TO E(10):: D(E
(5),F)=D(E(7),F)+D(E(8),F)::
NEXT F :: GOTO 840
-----
890 FOR F=E(9) TO E(10):: D(E
(5),F)=D(E(7),F)+E(4):: NEXT
F :: GOTO 840
-----
900 FOR AA=E(7) TO E(8):: FOR
F=E(9) TO E(10):: D(AA,F)=D(
AA,F)+E(4):: NEXT F :: NEXT
AA :: GOTO 840
-----
910 IF B$(4)="Y" THEN 930 ::
IF E(4)<>0 THEN 900
-----
920 FOR F=E(9) TO E(10):: FOR
AA=E(7) TO E(8):: D(AA,E(6))
=D(AA,E(6))+D(AA,F):: NEXT A
A :: NEXT F :: GOTO 840
-----
930 IF E(4)<>0 THEN 950
-----
940 FOR AA=E(7) TO E(8):: D(A
A,E(6))=D(AA,E(9))+D(AA,E(10
)):: NEXT AA :: GOTO 840
-----
950 FOR AA=E(7) TO E(8):: D(A
A,E(6))=D(AA,E(9))+E(4):: NE
XT AA :: GOTO 840
-----
960 IF B$(3)="Y" THEN 1020 :
: IF B$(4)="Y" THEN 980 :: I
F E(4)<>0 THEN 1010
-----
970 FOR AA=E(7) TO E(8):: FOR
F=E(9) TO E(10):: D(E(5),F)=
D(E(5),F)-D(AA,F):: NEXT F :
: NEXT AA :: GOTO 840
-----
980 IF E(4)<>0 THEN 1000
-----
990 FOR F=E(9) TO E(10):: D(E
(5),F)=D(E(7),F)-D(E(8),F)::
NEXT F :: GOTO 840
-----
1000 FOR F=E(9) TO E(10):: D(
E(5),F)=D(E(7),F)-E(4):: NEX
T F :: GOTO 840
-----
1010 FOR AA=E(7) TO E(8):: FO
R F=E(9) TO E(10):: D(AA,F)=D
(AA,F)-E(4):: NEXT F :: NEXT
AA :: GOTO 840
-----
1020 IF B$(4)="Y" THEN 1040
:: IF E(4)<>0 THEN 1010
-----
1030 FOR F=E(9) TO E(10):: FO
R AA=E(7) TO E(8):: D(AA,E(6)
)=D(AA,E(6))-D(AA,F):: NEXT
AA :: NEXT F :: GOTO 840
-----
1040 IF E(4)<>0 THEN 1060
-----
1050 FOR AA=E(7) TO E(8):: D(
AA,E(6))=D(AA,E(9))-D(AA,E(1
0)):: NEXT AA :: GOTO 840
-----
1060 FOR AA=E(7) TO E(8):: D(
AA,E(6))=D(AA,E(9))-E(4):: N
EXT AA :: GOTO 840
-----
1070 IF B$(3)="Y" THEN 1130
:: IF B$(4)="Y" THEN 1090 ::
IF E(4)<>0 THEN 1120
-----
1080 FOR AA=E(7) TO E(8):: FO
R F=E(9) TO E(10):: D(E(5),F)
=D(E(5),F)*D(AA,F):: NEXT F
:: NEXT AA :: GOTO 840
-----
1090 IF E(4)<>0 THEN 1110
-----
1100 FOR F=E(9) TO E(10):: D(
E(5),F)=D(E(7),F)*D(E(8),F):
: NEXT F :: GOTO 840
-----
1110 FOR F=E(9) TO E(10):: D(
E(5),F)=D(E(7),F)*E(4):: NEX
T F :: GOTO 840
-----
1120 FOR AA=E(7) TO E(8):: FO
R F=E(9) TO E(10):: D(AA,F)=D
(AA,F)*E(4):: NEXT F :: NEXT
AA :: GOTO 840
-----
1130 IF B$(4)="Y" THEN 1150
:: IF E(4)<>0 THEN 1120
-----
1140 FOR F=E(9) TO E(10):: FO
R AA=E(7) TO E(8):: D(AA,E(6)
)=D(AA,E(6))*D(AA,F):: NEXT
AA :: NEXT F :: GOTO 840
-----
1150 IF E(4)<>0 THEN 1170
-----
1160 FOR AA=E(7) TO E(8):: D(
AA,E(6))=D(AA,E(9))*D(AA,E(1
0)):: NEXT AA :: GOTO 840
-----
1170 FOR AA=E(7) TO E(8):: D(
AA,E(6))=D(AA,E(9))*E(4):: N
EXT AA :: GOTO 840
-----
1180 IF B$(3)="Y" THEN 1240
:: IF B$(4)="Y" THEN 1200 ::
IF E(4)<>4 THEN 1230
-----
1190 FOR AA=E(7) TO E(8):: FO
R F=E(9) TO E(10):: D(E(5),F)
=D(E(5),F)/D(AA,F):: NEXT F
:: NEXT AA :: GOTO 840
-----
1200 IF E(4)<>0 THEN 1220
-----
1210 FOR F=E(9) TO E(10):: D(
E(7),F)=D(E(7),F)/D(E(8),F):
: NEXT F :: GOTO 840
-----
1220 FOR F=E(9) TO E(10):: D(
E(5),F)=D(E(7),F)/E(4):: NEX
T F :: GOTO 840
-----
1230 FOR AA=E(7) TO E(8):: FO
R F=E(9) TO E(10):: D(AA,F)=D
(AA,F)/E(4):: NEXT F :: NEXT
AA :: GOTO 840
-----
1240 IF B$(4)="Y" THEN 1260
:: IF E(4)<>0 THEN 1230
-----
1250 FOR F=E(9) TO E(10):: FO
R AA=E(7) TO E(8):: D(AA,E(6)
)=D(AA,E(6))/D(AA,F):: NEXT
AA :: NEXT F :: GOTO 840
-----
1260 IF E(4)<>0 THEN 1280
-----
1270 FOR AA=E(7) TO E(8):: D(
AA,E(6))=D(AA,E(9))/D(AA,E(1
0)):: NEXT AA :: GOTO 840
-----
1280 FOR AA=E(7) TO E(8):: D(
AA,E(6))=D(AA,E(9))/E(4):: N
EXT AA :: GOTO 840
-----
1290 IF B$(4)="Y" THEN 1310
-----
1300 FOR AA=E(7) TO E(8):: FO
R F=E(9) TO E(10):: D(AA,F)=D
(AA,F)*E(4):: NEXT F :: NEXT
AA :: GOTO 840
-----
1310 IF B$(3)="Y" THEN 1330
-----
1320 FOR F=E(9) TO E(10):: D(
E(5),F)=D(E(7),F)^E(4):: NEX
T F :: GOTO 840
-----
1330 FOR AA=E(7) TO E(8):: D(
AA,E(6))=D(AA,E(9))^E(4):: N
EXT AA :: GOTO 840
-----
1340 M=2 :: BA=12 :: GOSUB 1
530 :: IF DAC1 THEN 200 :: G
$="SAVE DATA" :: GOSUB 1
-----
540 :: DISPLAY AT(18,5)BEEP:
"Change Title Name?":TAB(1
4-LEN(I$)/2);I$: " File may
be over-written"
-----
1350 ACCEPT AT(20,14-LEN(I$)
/2)SIZE(-14):I$: " DISPLAY A
T(23,2)BEEP:"FILE NAME?":D$
:: ACCEPT AT(23,13)SIZE(-10
):D$: " IF D$="" THEN 200 ::
OPEN #1:"DSK1."&D$,INTERNAL
,OUTPUT,VARIABLE :: W=W+1
-----
1360 PRINT #1:I$,C,B :: FOR
AA=0 TO C :: FOR F=0 TO B ::
PRINT #1:A$(AA,F),D(AA,F)::
NEXT F :: NEXT AA :: CLOSE
#1 :: M=M+1 :: F,Z=1 :: IA=0
:: GA=12 :: HA=21 :: M=16
-----
1370 G$="OPERATION COMPLETE"
:: GOSUB 1540 :: FOR K=1 TO
250 :: NEXT K :: RETURN

```

```

1380 BA=14 :: GOSUB 1530 ::
IF DA<1 THEN 200 :: G$="Print
Data" :: GOSUB 1540 :: G$="
Printer Ready?(Y/N)" :: GOS
UB 1510
-----
1390 GOSUB 1520 :: IF P=78 T
HEN 200 ELSE IF P<>89 THEN 1
390
-----
1400 OPEN #2:"PIQ",VARIABLE
132 :: X=X+1 :: GOSUB 1650
-----
1410 G$="DOUBLE SPACE?(Y/N)"
:: GOSUB 1510 :: IF P=78 TH
EN 1420 ELSE IF P<>89 THEN 1
410 :: C$="Y"
-----
1420 G$="Printing Data" :: G
OSUB 1540 :: PRINT #2:TAB(30
);H$: :: IF B>8 THEN PRINT
#2:CHR$(15);
-----
1430 FOR F=0 TO B :: PRINT #
2,USING 1460:A$(0,F);: NEXT
F :: PRINT #2: :: FOR FA=
0 TO B*10 :: PRINT #2:CHR$(4
5);: NEXT FA :: PRINT #2:
:: FOR AA=1 TO C :: PRINT
#2:A$(AA,0);TAB(9);: FOR F
=1 TO B :: PRINT #2,USING 14
70:D(AA,F);: NEXT F :: PRIN
T #
-----
1440 IF C$="Y" THEN PRINT #2
: :
-----
1450 NEXT AA :: FOR FA=0 TO
B*10 :: PRINT #2:CHR$(45);:
NEXT FA :: PRINT #2:CHR$(18
);: CLOSE #2 :: I=I+1 :: F,Z
=1 :: IA=0 :: GA=12 :: HA=21
:: RETURN
-----
1460 IMAGE "##### "
-----
1470 IMAGE "#####.## "
-----
1480 BA=7 :: GOSUB 1530 :: I
F DA<1 THEN 200 :: G$="REMI
NDER: Save Data" :: GOSUB 15
40 :: G$="Press C)clear or R)
eturn"
-----
1490 GOSUB 1510 :: IF P=82 T
HEN 200 :: IF P<>67 THEN 149
0
-----
1500 BA=7 :: GOSUB 1530 :: G
$="Clearing Memory" :: GOSUB
1540 :: FOR AA=0 TO 20 :: F
OR F=0 TO 13 :: A$(AA,F)="
" :: D(AA,F)=0 :: DISPLAY AT(A
A+4,F):D(AA,F): NEXT F :: N
EXT AA :: DA,W,H,X,I=0 :: I$
="" :: GOSUB 1370 :: GOTO 20
0
-----
1510 DISPLAY AT(24,14-LEN(G$
)/2):G$
-----
1520 CALL KEY(3,P,BA):: IF P
A=0 THEN 1520 ELSE RETURN
-----
1530 CALL CLEAR :: CALL SCRE
EN(BA):: FOR G=0 TO 13 :: CA
LL COLOR(G,M,BA):: NEXT G ::
RETURN
-----
1540 DISPLAY AT(2,14-LEN(G$)
/2):ERASE ALL:G$:TAB(14-LEN(G
$)/2);RPT$(" ",LEN(G$)): RE
TURN
-----
1550 BA=7 :: M=16 :: GOSUB 1
530 :: CALL ERR(J,EA,CA,R)::
DISPLAY AT(10,10):ERASE ALL
BEEP:"ERROR": :: " CODE";J
;"LINE";R;"TYPE";EA: :: " Se
e Page 217 X-Basic Manual" :
: D$="" :: IF W>H THEN CLOSE
#1 :: IF X>I THEN CLOSE #2
-----
1560 G$="Press R)eturn" :: G
OSUB 1510 :: IF (P<>82)+(P<>
114)THEN 1560
-----
1570 RETURN 200
-----
1580 BA=7 :: GOSUB 1530 :: G
$="REMINDER: Save Data" ::
GOSUB 1540 :: G$="Press Q)ui
t or R)eturn"
-----
1590 GOSUB 1510 :: IF P=82 T
HEN 200 ELSE IF P<>81 THEN 1
590 :: CALL CLEAR
-----
1600 BA=5 :: GOSUB 1530 :: D
ISPLAY AT(10,6):BEEP:"THANK-Y
OU FOR USING" :: TAB(6);"Greg
's Spread Sheet":TAB(6);"^^^
^^^^^^^^^^^^^^^^^^^^"
-----
1610 DISPLAY AT(16,4):"If yo
u like this Program":TAB(8
);"Send... $5.00 to...":TAB
(4);"Edmonton T.I. Users Gr
oup":TAB(10);"P.O.Box 11983"
:TAB(8);"Edmonton Alberta"
-----
1620 FOR K=1 TO 1500 :: NEXT
K
-----
1630 CALL PEEK(2,GA,HA):: CA
LL LOAD(-31804,GA,HA):: STOP
-----
1640 CALL KEY(S,P,BA):: RETU
RN
-----
1650 PRINT #2:CHR$(27);CHR$(
42);CHR$(0);: PRINT #2:CHR$(
27);CHR$(42);CHR$(1);CHR$(4
8);CHR$(0);CHR$(92);CHR$(34
);CHR$(81);CHR$(8);CHR$(69);C
HR$(2);CHR$(65);CHR$(34);CHR
$(28)
-----
1660 PRINT #2:CHR$(27);CHR$(
36);CHR$(1): RETURN

```

```

*****
* DECATUR 99er HOME COMPUTER USERS GRP *
* APPLICATION FOR MEMBERSHIP *
* *
* DATE / /85 *
* *
* NAME ----- *
* *
* ADDRESS ----- *
* *
* CITY ----- ZIP ----- *
* *
* PHONE ----- *
* *
* WORK PHONE ----- *
* *
* DUES: MEM,STUDENT $15 *
* ADD'L FAMLY $ 5 *
* OR NEWSLETTER ONLY $12 $ ----- *
* (25 MAX) *
*****

```

```

*****
$
$ Who are the Decatur 99'ers? $
$
$ We are an independant, nonprofit organization whose $
$ primary goal is to aid our members in understanding $
$ and applying the use of the TI99/4A Home Computer. $
$
$ Single membership dues are $15.00 per year. $
$
$ Additional members of the family may join for $5.00 $
$ each- up to $25.00 total per family. We ask that $
$ children under 16 be accompanied by an adult. $
$
$ For more information write to the return address or $
$ contact: $
$
$ President: Vice-president: $
$ Jay Seaberg Larry Livergood $
$ 41 Whippoorwill 467 W. William St. $
$ Decatur, Illinois Decatur, Illinois $
$ 62526 62522 $
$ PH 217-877-1631 PH 217-422-6933 $
$
$
*****

```

