HOCUS

Home Computer
Users Spotlight
a monthly publication of the
Milwaukee Area 99/4 Users Group

C21 8803 Milwaukee

MARCH—1988

MILWAUKEE AREA TI99 GROUP
4122 GLENDALE WAUWATOSA WI 53222

President...D.Walden	5292173
Vice-Pres...J.Schroeder	2644735
Treasurer...P.Norton	4628954
Secretary...B.Kling	5295161
Librarian...E.VonDerEhe	5490593
Librarian...F.Pabian	3273618
Newsletter..G.Hitz	5350133
S.I.G.....Schroeder/Walden/Hitz	

Next Group Meeting - 2nd Saturday
April 9,1988 - 12 noon til 4 PM
Wauwatosa S & L - 7500 West State

North Sub-Meeting - 1st Tuesday
April 5, 1988 - 7 PM til 10 PM
Security S & L - 5555 N Pt Washington

South Sub-Meeting - 3rd Tuesday
March 15, 1988 - 7 PM til 10 PM
Franklin State Bank - 7000 So 76th

Membership Dues $10 - Family $15

<<<<< HOCUS NEWSLETTER INDEX >>>>>

1988 HUGE COMPUTER ELECTRONIC SWAP MEET

The Milwaukee Area TI User Group will be holding their annual Swap-Meet next month at our regular April meeting. Most normal meeting affairs will be suspended to give more time for Meet activities. The table space is Free for all paid-up members, so take advantage of this opportunity to get rid of any computer//electronic hardware, firmware, software or related items that are no longer needed. Also get out there and help spread the word about it so that a large turnout of customers will attend. Everyone and his brother is invited, free admission to all, so you can bring all of your friends and relatives. And remember to bring lots of moolah too because there definitely will be loads of unbelievable bargains not normally available any place at any other time, and you'll be sorry if you miss out on them!

REMEMBER APRIL 9, 1988
SATURDAY AFTERNOON
WAUWATOSA Savings & Loan
12:00 til 3:30 PM

PROGRAMS THAT WRITE PROGRAMS
Part 1
by Jim Peterson

Way back in 1982, in the old 99'er Magazine, Vol. 1 Nos. 3 and 4, John Clulow wrote two articles entitled "How To Write a Basic Program That Writes Basic Programs". At that time I thought I would never understand what he was writing about!

But really, it's simple. Have you ever LISTed a program to the disk, and noticed that the resulting D/V80 file took up many more sectors than the program itself? That is because the TI saves programs in a compacted form, with each statement represented by a single token ASCII.

When a program is saved in MERGE format, by SAVE DSK (file-name),MERGE it is saved in this same compacted form, but in a D/V 163 file. And of course a D/V file can be created by a program - so you can write a program which will create a D/V 163 file in the form of a program, and then MERGE that file into memory and RUN it as a program, and SAVE it as a program.

You ask, why use this roundabout way of writing a program? Why not just key it in directly? Well, for one thing you can write program lines that could not possibly be keyed in directly. As for instance, the famous "line zero". Key this in, run it with a disk in drive 1, then enter MERGE DSK1.ZERO and LIST the result.

```
100 M$="BETCHA CAN'T DELETE
THIS!"
110 OPEN #1:"DSK1.ZERO",VARI
ABLE 163,OUTPUT :: PRINT #1:
CHR$(0)&CHR$(0)&CHR$(131)&CH
R$(200)&CHR$(LEN(M$))&M$&CHR
$(0)
120 PRINT #1:CHR$(255)&CHR$(
255):: CLOSE #1 :: END
```

Actually, there is an easy way to delete that line - but no way to key it in directly.

Here's another one - the full ASCII string.

```
100 OPEN #1:"DSK1.FULLSTRING
",VARIABLE 163,OUTPUT
110 LN=100 :: GOSUB 190 :: A
$=L$&"M$"&CHR$(190)
120 FOR J=1 TO 127 :: C$=C$&
CHR$(J):: NEXT J :: A$=A$&CH
R$(199)&CHR$(127)&C$&CHR$(0)
130 PRINT #1:A$
140 GOSUB 190 :: B$=L$&"M2$"
&CHR$(190)
150 FOR J=128 TO 255 :: D$=D
$&CHR$(J):: NEXT J :: B$=B$&
CHR$(199)&CHR$(128)&D$&CHR$(
0)
160 PRINT #1:B$
170 GOSUB 190 :: F$=L$&"M$"&
CHR$(190)&"M$"&CHR$(184)&"M2
$"&CHR$(0)
180 PRINT #1:F$ :: PRINT #1:
CHR$(255)&CHR$(255):: CLOSE
#1 :: END
190 L$=CHR$(INT(LN/256))&CHR
$(LN-256*INT(LN/256)):: LN=L
N+10 :: RETURN
```

Run that, then enter NEW, then MERGE DSK1.FULLSTRING. The string contains every ASCII from 0 to 255 in sequence, and there is no way to enter many of the unprintable ASCII codes from the keyboard. You can of course create that string in a program -

```
FOR J=0 TO 255 :: M$=M$&CHR$
(J):: NEXT J
```

but it saves a few seconds to have it predefined.

The full ASCII string is very useful for a quick shuffle without duplication. For instance, to scramble the numbers 200-250, -

```
100 M$="
          !""#$%&'()*+,-./
0123456789:;<=>?@ABCDEFGHIJK
LMNOPQRSTUVWXYZ[\]^_`abcdefg
hijklmnopqrstuvwxyz{|}"
110 M2$=""
120 M$=M$&M2$
130 M$=SEG$(M$,200,50)
140 L=LEN(M$):: RANDOMIZE ::
 X=INT(L*RND+1):: N=ASC(SEG$
```


CONT ON PAGE 11

CRASH-EM by Jim Beck

Program is in consol basic
Requires Joystick #1.
A variation of original TI
module game CAR WARS.

```
100 DIM C$(8)
110 DIM BC(9,4)
120 CALL CLEAR
130 CALL SCREEN(2)
140 PRINT "          CRASH
EM": : : : :
150 PRINT "          BY JIM B
ECK"
160 PRINT : : : : : : : :
170 PRINT "PRESS ANY KEY TO
START GAME."
180 FOR D=1 TO 14
190 CALL COLOR(8,16,2)
200 NEXT D
210 CALL KEY(0,K,S)
220 IF S=0 THEN 210
230 CALL CLEAR
240 FOR D=1 TO 14
250 CALL COLOR(8,2,2)
260 NEXT D
270 SCR=0
280 LT=0
290 CALL CHAR(120,"1010101
0101010")
300 CALL CHAR(121,"00000000F
F00
310 CALL CHAR(122,"00000000F
0101010")
320 CALL CHAR(123,"10101010F
0000000")
330 CALL CHAR(124,"00000000
1010101")
340 CALL CHAR(125,"10101010
0000000")
350 CALL CHAR(126,"00000000
1000000")
360 C$(1)="0010387C7C3B7C7C"
370 C$(2)="0000367EFE7E3600"
380 C$(3)="007C7C387C7C3810"
390 C$(4)="0000D8FCFEFCD800"
400 FOR D=1 TO 9
410 FOR DE=1 TO 4
420 READ BC(D,DE)
430 NEXT DE
440 NEXT D
450 RESTORE
460 FOR D=1 TO 4
470 CALL CHAR(127+D,C$(D))
480 CALL CHAR(135+D,C$(D))
490 NEXT D
500 FOR D=2 TO 12 STEP 2
510 CALL HCHAR(D,D+3,124)
520 CALL HCHAR(D,D+4,121,25-
2*D)
530 CALL HCHAR(26-D,D+3,125)
540 CALL HCHAR(26-D,D+4,121,
25-2*D)
550 CALL VCHAR(D+1,D+3,120,2
5-2*D)
560 CALL HCHAR(D,29-D,122)
570 CALL VCHAR(D+1,29-D,120,
25-2*D)
580 CALL VCHAR(26-D,29-D,123
)
590 CALL HCHAR(D+1,D+4,126,2
5-2*D)
600 CALL VCHAR(D+1,D+4,126,2
5-2*D)
610 CALL HCHAR(25-D,D+4,126,
25-2*D)
620 CALL VCHAR(D+1,28-D,126,
25-2*D)
630 NEXT D
640 FOR D=15 TO 17
650 CALL VCHAR(3,D,32,9)
660 CALL VCHAR(15,D,32,9)
670 NEXT D
680 FOR D=12 TO 14
690 CALL HCHAR(D,6,32,9)
700 CALL HCHAR(D,18,32,9)
710 NEXT D
720 CALL HCHAR(13,16,32)
730 CALL COLOR(12,8,2)
740 CALL COLOR(13,16,2)
750 CALL COLOR(14,10,2)
760 CALL COLOR(3,16,2)
770 CALL COLOR(4,16,2)
780 DIR=3
790 DOT=0
800 LT=LT+1
810 IF LT<10 THEN 830
820 LT=1
830 BC=BC(LT,3)
840 LC=BC(LT,4)
850 CALL HCHAR(13,16,LT+48)
860 RPC=BC(LT,1)
870 CPC=BC(LT,2)
880 ON DC GOSUB 1390,1420,14
50,1480
890 RMC=+1
900 CMC=+2
910 RP=23
920 LEV=5
930 OD=32
940 CP=17
950 DIR=DIR+1
960 ON DIR GOSUB 1390,1420,1
450,1480,1510
970 RM=+1
980 CM=+2
990 CALL GCHAR(RP+RM,CP+CM,F
R)
1000 IF FR=32 THEN 1070
1010 IF FR=135+OC THEN 2330
1020 IF FR<>126 THEN 950
1030 DOT=DOT+1
1040 SCR=SCR+5
1050 CALL SOUND(-10,330,9)
1060 IF DOT=180 THEN 1830
1070 CALL HCHAR(RP,CP,32)
1080 RP=RP+RM
1090 CP=CP+CM
1100 CALL HCHAR(RP,CP,DIR+12
7)
1110 CALL GCHAR(RPC+RMC,CPC+
CMC,FR)
1120 ND=32
1130 IF FR=127+DIR THEN 2330
1140 IF FR=32 THEN 1170
1150 IF FR<>126 THEN 1250
1160 ND=126
1170 CALL HCHAR(RPC,CPC,OD)
1180 OD=ND
1190 RPC=RPC+RMC
1200 CPC=CPC+CMC
1210 CALL HCHAR(RPC,CPC,135+
BC)
1220 IF RPC=13 THEN 2030
1230 IF CPC=16 THEN 2180
1240 GOTO 1320
1250 DC=DC-1
1260 IF DC>0 THEN 1280
1270 DC=4
1280 ON DC GOSUB 1390,1420,1
450,1480
1290 RMC=+1
1300 CMC=+2
1310 GOTO 1110
1320 IF CM<>0 THEN 1360
1330 IF RP=13 THEN 1350
1340 IF RP=13-RM THEN 1350
1350 GOTO 990
1360 IF CP=16 THEN 1680
1370 IF CP=16-CM THEN 1680
1380 GOTO 990
```

DATA BASE MANAGERS FOR THE TI-99/4A

By Bill Gaskill, reprint from Chicago Times.

Some owners/authors of the applications I have covered in this article will no doubt be angered by the apparent brutality of it. I choose to view it as honesty rather than brutality. Too many reviewers white wash the weaknesses of TI software they critically review. I will not. I think sometimes that we are afraid that the software market will dry up and blow away unless we give favorable reports on the software products that do appear for our computer. I prefer to think of it in another way; if we promote junk software in a favorable light those that do publish product reviews will lose credibility and those that buy software based upon those reviews will simply be that much more reluctant to get burned a second time. In the process of searching for the perfect data base manager I have purchased several programs and spent over $300. All of the programs that I own have positive points and all have negative points. What I have discovered to date is that the "perfect" data base manager does not exist yet (not even in the business world). What I am going to share with you are my impressions of the programs I own and in doing so will perhaps save you a little time and money if you too are looking for that "perfect" application.

The programs I own are;

ACORN 99 from Oak Tree Systems DBMS from Navarone Industries DATA BASE I from SPC Software DATA BASE 99 from Quality 99 Software DATA BASE 300 from the International Users Group DATA BASE X from Western Ware PR BASE v1.2 and V2.0 from William Warren TURBO DATAMAN from Easy Ware

I have used these programs enough to feel comfortable with each and could probably write several pages about each one. Unfortunately, publication space is limited and such a voluminous article would never see print because of it. Thus I have tried to be brief, but to the point in my comments on each program. Also, please keep in mind that my comments are subjective, based upon how each product meets MY needs and expectations. Yours may be different. For ease of reference I have included some of the information in a comparison table that allow analysis at a glance. In the paragraphs that follow I will try to provide a little detail to each issue and cover special features, lack of what I view as standard features and product performance of each program. I apologize in advance for the cryptic style you will read, however I needed to be brief. The DATA BASE 300 program will not be looked at since it is not available.

DATA BASE MANAGERS FOR THE TI-99/4A

ACORN 99:

Among the top three DBM's available to the TI community. The only relational data base available, also the only one with a programming language interface for custom applications. EXTREMELY powerful and well designed. Can support three active files at one time, allows existing data file formats to be edited, copied to another file, resequenced and can reformat a file structure into another file format. Does not have the ability to show number of records in a file. Can hold more than 1500 records per file on a SS/SD disk (depending on file size). Sorts alpha characters and strings better than numbers. Indexes record location for subfile creation and mainfile is then concatenated to create the subfile as another database. Possesses ability to search using; "equal to, unequal, greater than, less than, ignore" logical operators. Supports relational operators in search routines through the use of a true/false convention that allows selection of records where all parameters are met or any parameters are met. CAN print a single record from a display screen. EXTREMELY slow in operation. Uses 40 column text mode. Allows duplicate key field data entries. Allows printer control codes to be encrypted in setup file. Provides input checking for "numeric, integer, money, string, flag and date" entries. Overall, a fabulous program, with almost limitless potential. The best documentation of the group, giving many examples along with explanations. SUPERB application.

DBMS (Navarone):

Allows 32,000 records per file but only 350 per SS/SD diskette. Limits you to half that amount if you wish to sort the file since it creates a second sorted file that demands equal space on your data disk. Most interesting report generator I have ever seen, a cut and paste type affair that is really neat but poorly documented. Excellent custom screen design module which includes help screens that you design. FAST,FAST,FAST. Requires unique key field entries only, which I find inconvenient. Documentation is better than originally written but still confusing at times and incomplete. Dotes on mundane things and skips over or entirely omits important things. Does totaling in reports but no other computational work. Does not support single record printing but can use the report module to scroll data on screen, write it to disk or send it to your printer. Can append new data fields to the end of an existing record but cannot reformat the record in any other way. Can create subfiles but you have to figure out how to do it for yourself because the documentation does not tell you how. It doesn't even mention subfiles. Allows printer control codes to be encrypted in Report Generation file. Does not perform input checking of any type. All data is considered to be a string entry. Best suited for a hard disk environment. Not difficult to use once you have "played" with it, but can be intimidating at first.

DATA BASE I:

Best suited for mailing lists or other LIST type data files. Cumbersome design setup requiring records to be accessed by their relative position in the file (record number). You must first list the records by a specified field if you don't know the record number. Time consuming. Provides three pre-set mailing label report formats and one custom format for your own design. Will NOT do reports that have heading information. Includes several nice utilities such as a form letter generator, disk file data base which creates a DB1 data base file out of the information on your library of disks. Does not provide for input checking nor length of field entries. Only looks at the length of overall record. Does searches by "equal to" operator only, only on one data field at a time. Requires that you first create an index file and then search. To search by another field you must create another index file. Searches by a maximum of 5 characters in any field. Sorts are limited to 1000 records no matter how many exist in the file, but both alpha and numeric sorts are offered. Subfiles can be created to printer in the main program or to disk by using the Utilities options. Selection is by "equal to" or "between two values" which can be either alpha or numeric type.

DATA BASE 99:

More emphasis put on copy protection than on program performance. Allows custom screen design and claims 28 fields of up to 28 characters each. Would be a neat trick to do since four of the 24 rows on screen are used by program prompts. Fast assembly language interface for report generation. Cannot generate reports with headings and does not permit printer

ontrol codes to be inserted in report data. Does not save a ormat after design so you will have to re-create it each time ou want a report. Data is printed in continuous format vithout regard to page breaks or anything else. Design of ayout is cumbersome, requiring you to conceptualize how nany colons and/or semi-colons are needed to push the data cross the page. Number of colons/semi-colons is limited to 27 characters allowed in a LINPUT command. A terrible ystem. Disk catalog accessed from main menu will crash rogram if you enter an alpha character instead of a number vhen it prompts for the disk drive number to be cataloged. Color is lost after a crash since it was CALLed from the LOAD program. Does not permit single record screen print unless you buy the DB 99 Utilities), must use EDIT option o search for a record or search sequentially. Cannot go directly o a record by its relative position in the file. Will create ubfiles to disk allowing the search by "less than, equal to or reater than" operators. Search is limited to one field for all ractical purposes. Sorts can be performed in ascending order, y any one field. Sort is an actual re-write of the file. All data considered string information. No number crunching (again, nless you buy the DB99 Utilities), no input checking. Documentation consists of two 8 1/2" X 11" sheets of paper rinted on both sides. Program is slow, inflexible, nconvenient in many ways and cumbersome to use. It might ave been an advanced application two years ago. Today it is a inosaur, even with the E/A99 Utilities. Much too expensive.

DATA BASE X:

Very modular, meaning that each function (adding, editing, rinting, deleting etc.) is a separate program that must be oaded each time you want to use that function. Does statistical nalysis of data. Record counter is inaccurate, code of program s jumbled and entirely unstructured. Does not sort data even hough documentation uses the term "sort". What it means is select". When DATA BASE X " sorts" by a particular arameter it is really selecting records for dumping to a printer hat meet that parameter. Does allow selection between ranges. Cannot create subfiles, does not index existing records. Access f a record is done sequentially unless you know the record umber. No way to tell the record number, you must guess. upports 1 or 2 disk drives. Excruciatingly slow. Requires that ou name the data disk DBXDATA for no good reason that I an see, otherwise program errors out. Does not save report efinition but does allow it to be printed in normal or ompressed mode. Definition process is fairly simple but time onsuming. Documentation is the "shabbiest" I have ever seen. is photocopied and put into booklet form with the pages not ven cut straight, so that some information is missing off of ome pages. Overall, this program is JUNK! As with the JG's DATA BASE 300/500, it never really belonged on the arket in the state that it is in. Unfortunately I didn't know at and paid out over $30 to find out.

R BASE:

otally assembly language coded. THE BEST all-around pplication in my opinion. FAST, flexible, does virtually ything a user would want in the way of data handling except umber crunching. It will not do anything in that area. Treats l data as part of a big string just as DBMS and DATA BASE 9 do. As long as you own the PRB Utilities written by John hnson you can create subfiles, other wise you can't. Has on- ne help for commands, creates an index by any input field ou choose and then accesses any record in about 1 second. lso has a FIND feature to look at data sequentially in any ngle field and a GLOBAL option that searches for a single ata entry anywhere in the record. Saves up to five report formats, V2.0 allows you to format a data disk. Custom screen layout with terrific graphics options for borders/windows etc. is available. A tremendous program, well thought out, well designed, artistically executed. FAIRWARE!!! PRB Utilities are free for the asking as long as you provide the disk and mailer. Report design routine is cumbersome and confusing. Prints single record from screen display in either 40 or 80 column mode. Program is very sensitive about I/O device names. My copies (V1.2 and V2.0) both require PIO. to work rather than just PIO or PIO/1 etc. With number crunching abilities this program would be a perfect "flat-file data manager" for most TI users. As it is, the value and performance for a FAIRWARE application or a commercial application too for that matter, is unsurpassed. If you don't have PR BASE then you are missing out on one of the premier productivity tools available to the TI Community.

TURBO DATAMAN:

This is the second most powerful and useful data manager, taking a back seat only to PR BASE. It runs slightly ahead of ACORN because it performs number crunching and is faster in operation. Like ACORN, TURBO DATAMAN allows you to create a dictionary of data items (fields) and then lets you choose from that library of fields to put a record together. Up to 30 fields are allowed per record. Twenty pre-defined records (file formats) can exist on one disk. Allows custom screen layout design, complete with graphics for borders/windows etc. Does input checking, allows secondary screen access like ACORN's Detail Records. Allows formulas to be created and saved that perform the four basic math functions. Report definitions can be saved. Allows wildcard type operators in searches, will print single record from screen display. Provides "less than, greater than, equal to, not equal to, greater than or equal to, less than or equal to" operators in screen display and report generation modules. Permits sub-totals in reports that can be fomatted like TI Extended Basic does with the IMAGE statement. Subfiles can be created through the report generator by sending the output selected to a disk file rather than a printer. The results must be converted back to INTERNAL, FIXED from DISPLAY FIXED before you can use it in the program however. TURBO DATAMAN does not provide you with that utility. The documentation instructs you to "write a program" to do it. Names used for different modules in the program are confusing. Ex; ETCH,SKETCH, SKETCHR, FETCH. Should change names to more accurately reflect function of module. Documentation acceptable but lacks adequate coverage in some areas. Utilities are provided to perform some mundane operations such as counting the amount of records in a database. Reformatting or restructuring of an existing file is not permitted unless the input field is appended to the end of a record format. This program needs some "fine tuning" in some areas but is still an exciting productivity tool with immense possibilities. Its speed of operation is not fast but acceptable. It is faster than ACORN. One can set up the SKETCH program to auto-load if desired but the whole application should be centered around a menu in my opinion. As it is now, you must RUN each module from the READY> prompt when you need to use it, because every module exits with an END statement. If you don't own this program, you should. Whether you want to manage a mailing list or do accounting, TURBO DATAMAN is for you.

Reprinted from:
CALL SOUNDS, AUGUST 1987
Newsletter of the Central Westchester 99'ers,

- CHESS CLOCK -

by WESLEY R. RICHARDSON
BLUEGRASS 99 COMPUTER SOCIETY, INC.

The Extended BASIC program this month is called CHESS CLOCK. It was inspired by a program by Robert Marshall in the August, 1987 issue of MICROpendium. CHESS CLOCK allows two players to alternate running clocks by pressing any key on their side of the keyboard, or by pressing their fire button on the joystick. The clock can be preset to any time from zero to 99h 59m 59s, and will count either up or down.

You may wish to fine tune the rate of the clock for your machine by changing R1 constant for the left player and R2 for the right. To do this, use the formulas:

NEW R1=(OLD R1)*(ACTUAL TIME)/(LEFT CLOCK TIME)

NEW R2=(OLD R2)*(ACTUAL TIME)/(RIGHT CLOCK TIME)

Note that it is best to calibrate the clocks based on the normal maximum range that you intend to use. In spite of calibration these clocks will never be 100 percent accurate, but you can still have fun with them.

```
100 REM CHESSCLOCK
110 REM TI-99/4A EXTENDED BASIC
120 REM WESLEY R RICHARDSON, OCT 1987
130 REM BLUEGRASS 99 COMPUTER SOCIETY
140 R1=0.33294 ! LEFT CLOCK INCREMENT
    CONSTANT
150 R2=0.31917 ! RIGHT CLOCK INCREMENT
    CONSTANT
160 REM VARIABLES A$,B$,C$,I,J,K
170 REM VARIABLES R1,R2,R3,R4,S,T,T1,T2
    ,T$,U,U1,U2,V,V1,V2,W,W1,W2,W$,X,X1
    ,X2,Y,Y1,Y2,Z,Z1,Z2
180 DEF T$(S)=STR$((S-104)*0.25)
190 CALL CLEAR :: CALL SCREEN(5):: FOR
    I=3 TO 8 :: CALL COLOR(I,16,1):: NE
    XT I
200 DISPLAY AT(2,9):"CHESS CLOCK" :: DI
    SPLAY AT(23,5):"WESLEY R RICHARDSON
    "
210 DISPLAY AT(8,1):"YOU MAY USE EITHER
     JOYSTICK":"FIRE BUTTONS OR ANY KEY
     ON":"YOUR SIDE OF THE KEYBOARD TO"
220 DISPLAY AT(11,1):"START YOUR OPPONE
    NT'S CLOCK":"":"PRESSING SPACEBAR S
    TOPS THE":"CLOCKS TO HOLD RESET OR
    END"
230 DISPLAY AT(16,1):"IF JOYSTICKS ARE
    USED THEN":"LEFT  PLAYER USES JOYST
    ICK 1":"RIGHT PLAYER USES JOYSTICK
    2"
240 CALL CHAR(42,"0"):: CALL COLOR(2,2,
    16)
250 CALL CHAR(96,"0103070F1F3F7FFFFF7F3
    F1F0F070301000000000FFFFFFFFFFFF00
    0000000"):! L ARROW
260 CALL CHAR(100,"000000000FFFFFFFFFF
    FF00000000000C0E0F0F8FCFEFFFFFEFCF
    8F0E0C080"):! R ARROW
270 CALL CHAR(104,"071F3C38383838383838
    3838383C1F07E0F83C1C1C1C1C1C1C1C1
    C1C3CF8E0"):! 0
280 CALL CHAR(108,"0001030707010101010101
    0101010107070C0C0C0C0C0C0C0C0C0C0C
    0C0C0F0F0"):! 1
290 CALL CHAR(112,"0F1F3C383838000000003
    071F3C383F3FE0F0381C1C1C3C78F0C0800
    00000FCFC"):! 2
300 CALL CHAR(116,"3F3F000000000707000000
    0000383C3F1FFCFC1C3C78F0E0E0F0381C1
    C1C3CF8F0"):! 3
310 CALL CHAR(120,"0001030307060E0C1C18
    3F3F000000303E0E0E0E0E0E0E0E0E0E0F8F
    8E0E0F8F8"):! 4
320 CALL CHAR(124,"3F3F3838383F3F000000
    0038383C1F0FFCFC000000E0F87C1C1C1C1
    C1C3CF8F0"):! 5
330 CALL CHAR(128,"071F3C38383838383F3C
    3838383C1F07E0F83C1C1C0000E0F83C1C1
    C1C3CF8E0"):! 6
340 CALL CHAR(132,"3F3F000000000000000000
    0101010303037FCFC3C3C787878F0F0F0E0E
    0E0C0C0C0"):! 7
350 CALL CHAR(136,"071F3C3838383C1F1F3C
    3838383C1F07E0F83C1C1C1C3CF8F83C1C1
    C1C3CF8E0"):! 8
360 CALL CHAR(140,"071F3C3838383C1F0700
    0038383C1F07E0F83C1C1C1C3CFCDC1C1C1
    C1C3CF8E0"):! 9
370 U1,V1,W1,X1,Y1,Z1,U2,V2,W2,X2,Y2,Z2
    =104 :: T1,T2=0
380 REM RESTART POINT
390 CALL DELSPRITE(ALL)
400 DISPLAY AT(4,1):"LEFT  PLAYER    RI
    GHT PLAYER"
410 ACCEPT AT(5,1)SIZE(-12):A$ :: ACCEP
    T AT(5,17)SIZE(-12):B$
420 DISPLAY AT(6,1):"WHITE? YN" :: ACCE
    PT AT(6,8)SIZE(-1)VALIDATE("YN"):C$
430 IF (C$="Y")+(C$="y")THEN DISPLAY AT
    (6,1):"WHITE          BLACK" ELSE
    DISPLAY AT(6,1):"BLACK          WH
```

```
   ITE"
440 FOR I=7 TO 23 :: DISPLAY AT(I,1):""
    :: NEXT I
450 DISPLAY AT(7,1):"START TIME?      ST
   ART TIME?"
460 DISPLAY AT(8,8):T$(U1)&T$(V1)&" HO
   URS "&T$(U2)&T$(V2)
470 DISPLAY AT(13,8):T$(W1)&T$(X1)&" MI
   NUTES "&T$(W2)&T$(X2)
480 DISPLAY AT(18,8):T$(Y1)&T$(Z1)&" SE
   CONDS "&T$(Y2)&T$(Z2)
490 CALL NOW(T,8,8):: T=3600*T :: CALL
    NOW(U,13,8):: T=T+60*MIN(59,U):: CA
   LL NOW(U,18,8):: T1=0.5+T+MIN(59,U)
500 CALL NOW(T,8,19):: T=3600*T :: CALL
    NOW(U,13,19):: T=T+60*MIN(59,U)::
   CALL NOW(U,18,19):: T2=0.5+T+MIN(59
   ,U)
510 DISPLAY AT(23,1):"COUNT UP OR DOWN
   ? UD" :: ACCEPT AT(23,20)SIZE(-1)VA
   LIDATE("UD"):W$
520 IF (W$="U")THEN R3=R1 ELSE R3=-R1
530 IF (W$="U")THEN R4=R2 ELSE R4=-R2
540 DISPLAY AT(7,1):""
550 FOR J=0 TO 2 :: FOR I=1 TO 4 :: DIS
   PLAY AT(7+I+5*J,1):" ********
      ********" :: NEXT I :: NEXT J
560 DISPLAY AT(8,12)SIZE(-5):"HOURS" ::
    DISPLAY AT(13,11)SIZE(-7):"MINUTES
   " :: DISPLAY AT(18,11)SIZE(-7):"SEC
   ONDS"
570 CALL MAGNIFY(4)
580 CALL SPRITE(#1,96,1,24,112,#2,100,1
   ,24,112,#3,104,2,56,26,#4,104,2,56,
   58,#5,104,2,96,26)
590 CALL SPRITE(#6,104,2,96,58,#7,104,2
   ,136,26,#8,104,2,136,58,#9,104,2,56
   ,162,#10,104,2,56,194)
600 CALL SPRITE(#11,104,2,96,162,#12,10
   4,2,96,194,#13,104,2,136,162,#14,10
   4,2,136,194)
610 CALL TICK(0,T1,U1,V1,W1,X1,Y1,Z1)
620 CALL PATTERN(#3,U1,#4,V1,#5,W1,#6,X
   1,#7,Y1,#8,Z1)
630 CALL TICK(0,T2,U2,V2,W2,X2,Y2,Z2)
640 CALL PATTERN(#9,U2,#10,V2,#11,W2,#1
   2,X2,#13,Y2,#14,Z2)
650 REM MAIN LOOP
660 DISPLAY AT(22,1):"0=START LEFT   1=
   START RIGHT":"3=RESET          8=QUIT
   "
670 CALL KEY(0,K,S):: IF S=0 THEN 670
680 IF K=51 THEN 380 ! RESTART
690 IF K=56 THEN 860 ! END
700 IF (K<48)+(K>49)THEN 670
710 DISPLAY AT(22,1):" PRESS KEY OR FIR
   E BUTTON":" TO START OPPONENT'S CLO
   CK"
720 IF K=48 THEN 740 ! PLAYER 1
730 IF K=49 THEN 800 ! PLAYER 2
740 REM PLAYER 1
750 CALL COLOR(#1,4,#2,1)
760 CALL KEY(0,K,S):: IF K=32 THEN 790
    ELSE CALL KEY(1,K,S):: IF S=0 THEN
    770 ELSE 800
770 CALL TICK(R3,T1,U1,V1,W1,X1,Y1,Z1)
780 CALL PATTERN(#3,U1,#4,V1,#5,W1,#6,X
   1,#7,Y1,#8,Z1):: GOTO 760
790 CALL COLOR(#1,9):: GOTO 650
800 REM PLAYER 2
810 CALL COLOR(#1,1,#2,4)
820 CALL KEY(0,K,S):: IF K=32 THEN 850
    ELSE CALL KEY(2,K,S):: IF S=0 THEN
    830 ELSE 740
830 CALL TICK(R4,T2,U2,V2,W2,X2,Y2,Z2)
840 CALL PATTERN(#9,U2,#10,V2,#11,W2,#1
   2,X2,#13,Y2,#14,Z2):: GOTO 820
850 CALL COLOR(#2,9):: GOTO 650
860 REM END
870 CALL CLEAR :: CALL CHARSET :: CALL
    DELSPRITE(ALL):: END
880 SUB TICK(R,T,U,V,W,X,Y,Z)
890 T=T+R
900 IF T<0 THEN T=359999.5
910 IF T>=360000 THEN T=0.5
920 S=T/36000 :: U=INT(S):: S=10*(S-U):
    : V=INT(S):: S=6*(S-V):: W=INT(S)
930 S=10*(S-W):: X=INT(S):: S=6*(S-X)::
     Y=INT(S):: Z=INT(10*(S-Y))
940 U=104+4*U :: V=104+4*V :: W=104+4*W
    :: X=104+4*X :: Y=104+4*Y :: Z=104
   +4*Z
950 SUBEND
960 SUB NOW(W,X,Y)
970 ACCEPT AT(X,Y)SIZE(-2)VALIDATE("012
   3456789"):W$ :: W=VAL(W$)
980 SUBEND
```

TIPS FROM THE TIGERCUB

945

TIGERCUB SOFTWARE
156 Collingwood Ave.
Columbus, OH 43213

Distributed by Tigercub Software to TI-99/4A Users Groups for promotional purposes and in exchange for their newsletters. May be reprinted by non-profit users groups, with credit to Tigercub Software.

Over 130 original programs in Basic and Extended Basic, available on cassette or disk, now reduced to just $2.00 each, plus $1.50 per order for cassette or disk and PP&H. Cassette programs will not be available after my present stock of blanks is exhausted.
Descriptive catalogs, while they last, $1.00 which is deductable from your first order.

Tigercub Full Disk Collections, reduced to $10 postpaid. Each of these contains either 5 or 6 of my regular $2 catalog programs, and the remaining disk space has been filled with some of the best public domain programs of the same category. I am NOT selling public domain programs - they are a free bonus!
TIGERCUB'S BEST, PROGRAMMING TUTOR, PROGRAMMER'S UTILITIES, BRAIN GAMES, BRAIN TEASERS, BRAIN BUSTERS!, MANEUVERING GAMES, ACTION REFLEX AND CONCENTRATION, TWO-PLAYER GAMES, KID'S GAMES, MORE GAMES, WORD GAMES, ELEMENTARY MATH, MIDDLE/HIGH SCHOOL MATH, VOCABULARY AND READING, MUSICAL EDUCATION, KALEIDOSCOPES AND DISPLAYS

NUTS & BOLTS (No. 1), a full disk of 100 Extended Basic utility subprograms in merge format, ready to merge into your own programs. Plus the Tigercub Menuloader, a tutorial on using subprograms, and 5 pages of documentation with an example of the use of each subprogram. Reduced to $15.00 postpaid.
NUTS & BOLTS NO. 2, another full disk of 108 utility subprograms in merge format, all new and fully compatible with the last, and with 10 pages of documentation and examples. Also $15 postpaid.

```
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
% NUTS & BOLTS #3 is now       %
% ready, another full disk     %
% of 140 new merge-format      %
% utility subprograms, all     %
% compatible with the pre-     %
% vious. With 11 pages of      %
% documentation, $15 ppd.      %
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
```

TIPS FROM THE TIGERCUB, a full disk containing the complete contents of this newsletter Nos. 1 through 14, 50 original programs and files, reduced to $10 ppd.
TIPS FROM THE TIGERCUB VOL. 2, another diskfull, complete contents of Nos. 15 through 24, over 60 files and programs, also just $10
TIPS FROM THE TIGERCUB VOL. 3, another 62 programs, tips and rdutines from Nos. 25 through 32, $10 postpaid.
TIPS FROM THE TIBERCUB VOL. 4, another 48 programs and files from issues 33 through 41, also $10 postpaid.

Here is a versatile printer utility which will accept all printer control codes, print in 1 to 5 columns with choice of column separation and margin width, allow alternate margins and pause at end of page to turn paper over, and will load and print a diskfull of files one after another. It is set up for the Gemini 10X and may require modification for other printers.

```
100 DIM M$(400),F$(50)
110 GOTO 150
120 K,ST,SET,S,P$,P,CL,DW$,S
$$,I$,D$,E$,NC,CW,TC,TA,TX,A
V,CS,S$,LT,A$,LSP,LP,RM,OK$,
QQ$,X,F$(),SL,F,IP,N$(1),T$,F
LAG,J,PP,LT$
130 CALL CLEAR :: CALL KEY :
: CALL COLOR :: CALL SCREEN
:: CALL SOUND
140 !SP-
150 CALL CLEAR :: CALL KEY(3
,K,ST):: ON WARNING NEXT
160 FOR SET=0 TO 14 :: CALL
COLOR(SET,2,8):: NEXT SET ::
 CALL SCREEN(5)
170 DISPLAY AT(3,6):"TIGERCU
B PRINTALL": :TAB(7);"Copyri
ght 1987":TAB(6);"Tigercub S
oftware" !programmed by Jim
 Peterson
180 DISPLAY AT(12,1):"May be
 distributed without restr
iction providing that";"no p
rice or copying fee is";"cha
rged."
190 DISPLAY AT(18,7):"TURN P
RINTER ON!"
200 DISPLAY AT(20,8):"PRESS
ANY KEY" :: DISPLAY AT(20,8)
:"press any key" :: CALL KEY
(0,K,S):: IF S=0 THEN 200 EL
SE CALL CLEAR
210 DISPLAY AT(12,1):"PRINTE
R DESIGNATION?" :: ACCEPT AT
(14,1)BEEP:P$ :: IF P$$IP$,"
.LF",1)=0 THEN P$=P$&".LF"
220 ON ERROR 230 :: OPEN #1:
P$,VARIABLE 255 :: ON ERROR
STOP :: PRINT #1:CHR$(27);"8
" :: CALL CLEAR :: GOTO 240
230 DISPLAY AT(20,1):"CANNOT
 OPEN PRINTER!" :: RETURN 21
0
240 DISPLAY AT(12,1):"PRINT
SIZE?": :" (1) PICA": " (2)
 ELITE":" (3) CONDENSED"
250 ACCEPT AT(12,13)VALIDATE
("123")SIZE(1):P :: PRINT #1
:CHR$(27);"8";CHR$(P);
260 !The values 80, 96 and 1
36 in the next line are the
 maximum number of pica, elit
e and condensed characters p
er line on Gemini 10X
270 !Change as necessary fo
 your printer!
280 CL=(P=1)*80+(P=2)*96+(P
3)*136 :: CL=ABS(CL)
290 DISPLAY AT(12,1)ERASE A
LL:"DOUBLE-WIDTH? (Y/N) N" :
 ACCEPT AT(12,21)SIZE(-1)VA
LIDATE("YN")BEEP:DW$ :: IF D
W$="Y" THEN PRINT #1:CHR$(27
);"W";CHR$(1);:: CL=CL/2
300 DISPLAY AT(12,1)ERASE A
LL:"SUPERSCRIPT? (Y/N) N" ::
 ACCEPT AT(12,20)SIZE(-1)VAL
IDATE("YN")BEEP:SS$ :: IF SS
="Y" THEN PRINT #1:CHR$(27);
"S";CHR$(0);
310 DISPLAY AT(12,1)ERASE AL
L:"ITALICS? (Y/N) N" :: ACCE
PT AT(12,16)VALIDATE("YN")SI
ZE(-1)BEEP:I$ :: IF I$="Y" T
HEN PRINT #1:CHR$(27);"4";
320 DISPLAY AT(12,1)ERASE AL
L:"DBUBLE-STRIKE? (Y/N) Y" :
: ACCEPT AT(12,22)VALIDATE("
YN")SIZE(-1)BEEP:D$ :: IF D$
="Y" THEN PRINT #1:CHR$(27);
"G";
330 IF P<>3 AND P<>4 THEN DI
SPLAY AT(12,1):"EMPHASIZED?
(Y/N) Y" :: ACCEPT AT(12,19)
VALIDATE("YN")SIZE(-1)BEEP:E
$ :: IF E$="Y" THEN PRINT #1
:CHR$(27);"E";
340 DISPLAY AT(12,1)ERASE AL
L:"NUMBER OF COLUMNS? (1-5)"
 :: ACCEPT AT(12,26)VALIDATE
("12345")SIZE(1)BEEP:NC
350 DISPLAY AT(12,1):"COLUMN
 WIDTH (NUMBER 8F": :"CHARAC
TERS?" :: ACCEPT AT(14,13)VA
LIDATE(DIGIT)BEEP:CW
360 TC=NC*CW :: TA=CL-TC ::
TX=TC+NC*2-2
370 IF TX<=CL THEN 390 :: DI
SPLAY AT(18,1):STR$(NC)&" co
lumns of "&STR$(CW)&" charac
ters":"plus 2-column spacing
 equals"
380 DISPLAY AT(20,1):STR$(TC
)&" characters; maximum":"av
ailable in print size":"sele
cted is "&STR$(CL)&".":"####
Please reselect####" :: GOTO
 240
390 IF NC=1 THEN 410 :: AV=1
NT(TA/(NC-1)):: DISPLAY AT(1
2,1)ERASE ALL:"COLUMN SEPARA
TION?":"MINIMUM 2":"MAXIMUM
 "&STR$(AV)&" AVAILABLE ":"2"
400 ACCEPT AT(15,1)VALIDATE(
```

```
IT)SIZE(-2)BEEP:CS :: IF
2 OR CS)AV THEN 400 ELSE
RPT$(" ",CS)
) TA=TA-CS$(NC-1):: IF TA<
HEN 450
) DISPLAY AT(12,1)ERASE AL
LEFT MARGIN WIDTH?": :"MA
UM "&STR$(TA)&" AVAILABLE
:: ACCEPT AT(12,20)VALIDAT
IGIT)BEEP:LT :: IF LT)TA
N 420
) DISPLAY AT(12,1):"ALTERN
NG LEFT/RIGHT": :"MARGIN?
or pages to be":"later re
duced on both":"sides) (Y
N"
) ACCEPT AT(16,14)VALIDATE
N")SIZE(-1):A$
) LSP=12 :: DISPLAY AT(10,
" ":" ":"LINES PER PAGE?
":" ":" ":" " :: ACCEP
T(12,17)VALIDATE(DIGIT)SI
-3):LP :: IF LP<70 THEN 4

) DISPLAY AT(12,1):"LINE S
ING - 72 INCH" :: DISPLAY
(11,16):"_" :: ACCEPT AT
,16)VALIDATE(DIGIT)BEEP:L

) IF LP/((INT(72/LSP)))<11.5
EN DISPLAY AT(20,1):"MON'
IT!" :: GOTO 450
PRINT #1:CHR$(27);"A";CH
LSP);
RM=TA-LT
DISPLAY AT(12,1)ERASE AL
TR$(NC)&" columns of":GTR
M)&"-character width":"Ie
margin of "&STR$(LT)&" sp
s"
) DISPLAY AT(15,1):STR$(LP
lines per page":"with "&
$(LSP)&"/72 line spacing"
DISPLAY AT(17,1):STR$(CS
spaces between columns":
ght margin of "&STR$(RM)&
paces":"OK? (Y/N) Y"
ACCEPT AT(20,11)VALIDATE
N")SIZE(-1)BEEP:OK$ :: IF
$="N" THEN 240
) DISPLAY AT(12,1)ERASE AL
PAUSE AT END OF PAGE? N"
ACCEPT AT(12,23)VALIDATE(
")SIZE(-1):QQ$
) DISPLAY AT(1,1)ERASE ALL
NPUT FILENAMES TO BE":"PR
ED.":"PRESS ENTER WHEN DO

) I=I+1 :: DISPLAY AT(I+3,
"FILENAME? DSK" :: ACCEPT
 AT(I+3,14)SIZE(-12)BEEP:F$(
I)
570 IF F$(I)="" THEN I=I-1 :
: GOTO 600 ELSE F$(I)="DSK"&
F$(I)
580 ON ERROR 590 :: OPEN #2:
F$(I):: CLOSE #2 :: GOTO 560
590 ON ERROR STOP :: CALL SO
UND(1000,110,0,-4,0):: DISPL
AY AT(20,1):"CANNOT OPEN "&F
$(I):: I=I-1 :: RETURN 560
600 SL=1
610 F=F+1 :: IF F>I THEN 700
:: ON ERROR 620 :: OPEN #2:
F$(F),INPUT :: DISPLAY AT(22
,1):"READING "&F$(F):: ON ER
ROR STOP :: GOTO 630
620 CALL SOUND(1000,110,0,-4
,0):: DISPLAY AT(20,1):"COUL
D NOT OPEN "&F$(F):: STOP
630 FOR IP=SL TO LP$NC :: LI
NPUT #2:M$(IP):: IF LEN(M$(I
P))=0 THEN 670 :: IF NC>1 AN
D POS(M$(IP),CHR$(13),1)<>0
THEN M$(IP)=SEG$(M$(IP),1,LE
N(M$(IP))-1)
640 IF ASC(M$(IP))>126 OR AS
C(M$(IP))<32 THEN IP=IP-1 ::
 GOTO 680
650 IF LEN(M$(IP))<=CW THEN
670 :: T$=SEG$(M$(IP),1,CW):
: CALL SOUND(1000,110,0,-4,0
):: DISPLAY AT(12,1):M$(IP);
" OVER";CW;"CHARACTERS":"TRU
NCATED TO ";T$:"OK?"
660 CALL KEY(3,K,S):: IF S=0
 THEN 660 ELSE IF K<>89 THEN
 STOP ELSE M$(IP)=T$
670 M$(IP)=M$(IP)&RPT$(" ",C
W-LEN(M$(IP)))
680 IF EOF(2)=1 THEN CLOSE #
2 :: SL=IP+1 :: GOTO 610
690 NEXT IP :: IF EOF(2)=1 T
HEN CLOSE #2 :: GOTO 720 ELS
E GOTO 700
700 ON ERROR 710 :: FLAG=1 :
: FOR J=IP+1 TO NC$LP :: M$(
J)="" :: NEXT J :: GOTO 720
710 STOP
720 PP=PP+1 :: IF PP/2=INT(P
P/2)AND A$="Y" THEN LT$=RPT$
(" ",RM)ELSE LT$=RPT$(" ",LT
)
730 FOR J=1 TO LP :: ON NC G
OSUB 750,760,770,780,790 ::
NEXT J :: PRINT #1:CHR$(12):
: SL=1 :: IF F>I THEN STOP E
LSE IF QQ$="N" THEN 630
740 DISPLAY AT(24,1)BEEP:"PR
ESS ANY KEY TO CONTINUE" ::
 CALL KEY(0,K,S):: IF S=0 THE
N 740 ELSE DISPLAY AT(24,1):
"" :: GOTO 630
750 PRINT #1:LT$&M$(J)&CHR$(
10):: RETURN
760 PRINT #1:LT$&M$(J)&LS$&M$
(J+LP)&CHR$(10):: RETURN
770 PRINT #1:LT$&M$(J)&LS$&M$
(J+LP)&LS$&M$(J+LP$2)&CHR$(10
):: RETURN
780 PRINT #1:LT$&M$(J)&LS$&M$
(J+LP)&LS$&M$(J+LP$2)&LS$&M$(J
+LP$3)&CHR$(10):: RETURN
790 PRINT #1:LT$&M$(J)&LS$&M$
(J+LP)&LS$&M$(J+LP$2)&LS$&M$(J
+LP$3)&LS$&M$(J+LP$4)&CHR$(10
):: RETURN
```

This is an improved version of the math program in Tips #36.

```
100 CALL CLEAR :: RANDOMIZE
110 B=INT(3*RND+2):: IF B=B2
 THEN 110 ELSE B2=B
120 F=INT(3*RND+2):: IF F=F2
 THEN 120 ELSE F2=F
130 D=INT(3*RND+2):: IF D=D2
 THEN 130 ELSE D2=D
140 X=F*B*D
150 BB=INT(3*RND+2):: IF BB=
BB2 OR BB=B THEN 150 ELSE BB
2=BB
160 DD=INT(3*RND+2):: IF DD=
DD2 OR DD=D THEN 160 ELSE DD
2=DD
170 F=F$BB$DD
180 DISPLAY AT(3,1)ERASE ALL
:"IF";B;"BOYS CAN CATCH";X;"
FROGS IN";D;"DAYS,"
190 DISPLAY AT(6,1):"HOW MAN
Y FROGS CAN";BB;"BOYS":"CATC
H IN";DD;"DAYS?"
210 ACCEPT AT(7,19):Q
220 IF Q=F THEN DISPLAY AT(9
,1):"THAT'S RIGHT!" :: GOTO
 110
230 DISPLAY AT(9,1):"NO, THA
T'S WRONG."
240 DISPLAY AT(11,1):"IF";B;
"BOYS CAN CATCH";X;"FROGS IN
";B;"DAYS"
250 DISPLAY AT(13,1):"THEN O
NE BOY CAN CATCH";X/B;"FROGS
 IN";D;"DAYS"
260 DISPLAY AT(15,1):"AND ON
E BOY CAN CATCH";X/B/D;"FROG
S IN ONE DAY."
270 DISPLAY AT(17,1):"SO, IF
 ONE BOY CAN CATCH";X/B/D;"F
ROGS IN ONE DAY,"
280 DISPLAY AT(19,1):"THEN";
BB;"BOYS CAN CATCH";X/B/D*BB
;"FROGS IN ONE DAY"
290 DISPLAY AT(21,1):"AND";B
B;"BOYS CAN CATCH";X/B/D*BB*
DD;"FROGS IN";DD;"DAYS."
300 DISPLAY AT(24,1):"PRESS
ANY KEY" :: CALL KEY(0,K,S):
: IF S=0 THEN 300 ELSE 110
```

Here's an idea for an unusual title screen -

```
100 CALL CLEAR :: FOR SET=1
TO 8 :: CALL COLOR(SET,1,1):
: NEXT SET :: CALL CHAR(100,
"0",101,"0")
110 X$(0)="4043241818244202"
 :: X$(1)="4021261818648402"
 :: X$(2)="2020131C38C80404"
 :: X$(3)="101010FF8080808"
 :: X$(4)="0810101907E11020"
120 X$(5)="080808F8F1F01010"
 :: X$(6)="0404C6381C132020"
 :: X$(7)="0284641819262140"
130 A$=RPT$(CHR$(100)&CHR$(1
01),13):: FOR R=1 TO 24 :: C
=C+1+(C=2)*2 :: DISPLAY AT(R
,C):A$ :: NEXT R
140 CALL VCHAR(1,29,1,168)
150 CALL SCREEN(2):: CALL CO
LOR(9,5,16):: FOR S=1 TO 8 :
: CALL COLOR(S,16,2):: NEXT
S
160 DISPLAY AT(5,5):" TIGERC
UB SOFTWARE ":: DISPLAY AT(
8,6):" SQUIRMY SCREEN "
170 FOR J=0 TO 7 :: CALL CHA
R(100,X$(J)):: CALL CHAR(101
,X$(7-J)):: NEXT J
180 CALL KEY(0,K,S):: IF S=0
 THEN 170
```

```
 MEMORY FULL

      Jim Peterson
```

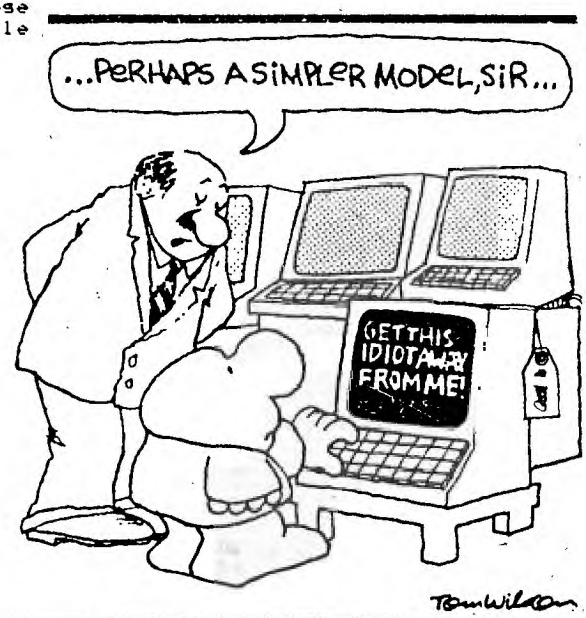
```
100 ' xxxxxxxxxxxxxxxxxxxx
110 ' V 3 . 5 CARD CATALOG 
120 ' xxxxxxxxxxxxxxxxxxxx
130 ! Original: Ed York
      Revamped: Rick Kellogg
      Revision: Ed York
140 !
150 DIM A$(12):: FOR A=1 TO
 12 :: READ A$(A):: NEXT A ::
 CALL CLEAR
160 DATA JAN,FEB,MAR,APR,MAY
,JUN,JUL,AUG,SEP,OCT,NOV,DEC
170 DISPLAY AT(9,9):"DISK LA
BELER": :"     ENTER TODAYS
 DATE": :TAB(11);"MM/DD/YY": 
 : :TAB(11);"__/__/__"
180 ACCEPT AT(16,11)BEEP SIZ
E(-2)VALIDATE(DIGIT):D$ :: I
F D$="__" THEN 180 ELSE IF I
VAL(D$)>12)+(VAL(D$)<1)THEN
 170
190 ACCEPT AT(16,14)SIZE(-2)
VALIDATE(DIGIT)BEEP:C$ :: IF
 C$="__" THEN 190 ELSE IF (V
AL(C$)>31)+(VAL(C$)<1)THEN 1
90
200 ACCEPT AT(16,18)SIZE(-1)
VALIDATE(DIGIT)BEEP:D$ :: D$
="0"&D$ :: GOSUB 610
210 E$=A$(VAL(D$))&" "&C$&",
 19"&D$ :: F$(1)="D/F" :: F$
(2)="D/V" :: F$(3)="I/F" ::
F$(4)="I/V" :: F$(5)="PROGRM
"
220 IMAGE "DISKNAME: 0000000
$30  FREE:0000   USED:0000"
230 IMAGE "000000 SIDED/0000
00 DENSITY     000000000000"
240 IMAGE "0000000000 0000 0
00000 0       0000000000 000
0 000000 0       0000000000
0000 000000 0"
250 IMAGE "0000000000 0000 0
00000 0       0000000000 000
0 000000 0"
260 IMAGE "0000000000 0000 0
00000 0"
270 OPEN #1:"PIO",VARIABLE 1
00 :: PRINT #1:CHR$(15);CHR$
(27);"9";CHR$(0);CHR$(27);"A
";CHR$(5)
280 OPEN #2:"DSK"&STR$(D)&".
",INPUT ,RELATIVE,INTERNAL 
: INPUT #2:I0,C,C,D
290 PRINT #1:RPT$("=",86);CH
R$(27);"A";CHR$(3):: E=0
300 PRINT #1:CHR$(27);"W";CH
R$(1):: PRINT #1,USING 220:6
0,D,C-D
310 IF C<>D AND C<144 THEN
 H$="DOUBLE" :: I$="DOUBLE"
320 IF C<250 AND C>721 THEN
 H$="SINGLE" :: I$="DOUBLE"
330 IF C<361 THEN H$="SINGLE
" :: I$="SINGLE"
340 PRINT #1:CHR$(14):: PRIN
T #1,USING 230:H$,I$,E$ :: P
RINT #1:CHR$(27);"W";CHR$(0)
350 PRINT #1:RPT$("=",86);CH
R$(27);"A";CHR$(5)
360 PRINT #1,USING 240:" Fil
ename ","Size"," Type ","P",
" Filename ","Size"," Type "
,"P"," Filename ","Size"," T
ype ","P"
370 PRINT #1,USING 240:"----
------","----","------","-",
"----------","----","------"
,"-","----------","----","--
----","-"
380 IF E=126 THEN 620
390 FOR F=1 TO 3 :: INPUT #2
:J$(F),G(F),H(F),I(F):: NEXT
 F :: IF LEN(J$(1))=0 THEN 4
40 ELSE IF LEN(J$(2))=0 THEN
 GOSUB 490 ELSE 410
400 PRINT #1,USING 260:J$(1)
,H(1),K$(1),L$(1):: E=E+1 ::
 GOTO 380
410 IF LEN(J$(3))=0 THEN GOS
UB 490 :: GOSUB 530 :: PRINT
 #1,USING 250:J$(1),H(1),K$(
1),L$(1),J$(2),H(2),K$(2),L$
(2):: E=E+2 :: GOTO 380
420 GOSUB 490 :: GOSUB 530 :
: GOSUB 570
430 PRINT #1,USING 240:J$(1)
,H(1),K$(1),L$(1),J$(2),H(2)
,K$(2),L$(2),J$(3),H(3),K$(3
),L$(3):: E=E+3 :: GOTO 380
440 PRINT #1: :: PRINT #1:CH
R$(8);CHR$(27);CHR$(50):: C
LOSE #2 :: CLOSE #1 :: DISPL
AY AT(20,1):"Want another Co
py or Disk? Y": :""
450 ACCEPT AT(20,28)SIZE(-1)
VALIDATE("YN")BEEP:M$ :: IF
 M$="Y" THEN GOSUB 610 :: GOT
O 470 ELSE CALL CLEAR
460 OPEN #1:"PIO" :: PRINT #
1:CHR$(27);"T" :: CLOSE #1 :
: STOP
470 DISPLAY AT(20,1)BEEP:"
INSERT DISK INTO DRIVE "&STR
$(D): :"   PRESS ANY KEY TO
 BEGIN   "
480 CALL KEY(0,J,K):: IF K<1
 THEN 480 ELSE 270
490 IF G(1)>0 THEN L$(1)=" "
 ELSE L$(1)="Y"
500 IF ABS(G(1))=5 THEN K$(1
)=F$(5):: RETURN ELSE A=LEN(
STR$(I(1)))
510 IF A=1 THEN K$(1)=F$(ABS
(G(1)))&"  "&STR$(I(1)):: RE
TURN
520 IF A=2 THEN K$(1)=F$(ABS
(G(1)))&"  "&STR$(I(1)):: RET
URN ELSE K$(1)=F$(ABS(G(1)))
&STR$(I(1)):: RETURN
530 IF G(2)>0 THEN L$(2)=" "
 ELSE L$(2)="Y"
540 IF ABS(G(2))=5 THEN K$(2
)=F$(5):: RETURN ELSE A=LEN(
STR$(I(2)))
550 IF A=1 THEN K$(2)=F$(ABS
(G(2)))&"  "&STR$(I(2)):: RE
TURN
560 IF A=2 THEN K$(2)=F$(ABS
(G(2)))&"  "&STR$(I(2)):: RET
URN ELSE IF A=3 THEN K$(2)=F
$(ABS(G(2)))&STR$(I(2)):: RE
TURN
570 IF G(3)>0 THEN L$(3)=" "
 ELSE L$(3)="Y"
580 IF ABS(G(3))=5 THEN K$(3
)=F$(5):: RETURN ELSE A=LEN(
STR$(I(3)))
590 IF A=1 THEN K$(3)=F$(ABS
(G(3)))&"  "&STR$(I(3)):: RE
TURN
600 IF A=2 THEN K$(3)=F$(ABS
(G(3)))&"  "&STR$(I(3)):: RET
URN ELSE IF A=3 THEN K$(3)=F
$(ABS(G(3)))&STR$(I(3)):: RE
TURN
610 DISPLAY AT(24,7):"CATALO
G DRIVE:  1" :: ACCEPT AT(24
,23)BEEP SIZE(-1)VALIDATE("1
234"):D :: RETURN
620 INPUT #2:J$(1),G(1),H(1)
,I(1):: GOSUB 490 :: PRINT #
1,USING 250:J$(1),H(1),K$(1)
,L$(1):: GOTO 440
630 END
```

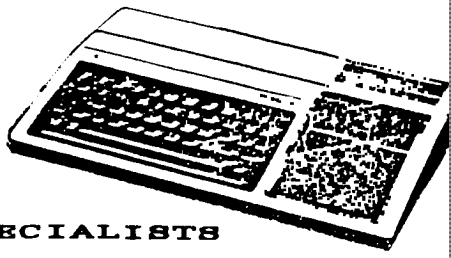
```
(M$,X,1)):: M$=SEG$(M$,1,X-1
)&SEG$(M$,X+1,255)
150 PRINT N;:: IF LEN(M$)=0
THEN STOP ELSE 140
```

One more example - can you run this program and get these results? You won't even be able to key in that last line!

```
>LIST
100 FOR J=1 TO 7 :: READ M$
:: PRINT M$ :: NEXT J
30000 DATA AAAAAAAAAAAAAAAAA
AAAAAAAAAA,BBBBBBBBBBBBB,BB
BBBBBBBBBBB,CCCCCCCCCCCCC,
            DDDDDDDDDDDDDDD
30010 DATA "TESTING",,,,,,,,
,,,,,,,,,,,,,,,,,,,,,,""TEST
ING""
>RUN
AAAAAAAAAAAAAAAAAAAAAAAAAAA
BBBBBBBBBBBBB,BBBBBBBBBBBBB
CCCCCCCCCCCCC
            DDDDDDDDDDDDDD
"TESTING"
,,,,,,,,,,,,,,,,,,,,,,,,,,,
""TESTING""

*READY*
```





TI-99/4A

HOME COMPUTER SPECIALISTS

WE HAVE WHAT YOU NEED FOR YOUR TI!

COMPETITION COMPUTER PRODUCTS

2629 W. NATIONAL AVE. MILWAUKEE, WIS. 53204
(near the Mitchell Park Domes)

STORE HOURS; MON THRU FRI 10-6 SAT 10-3

672-4010

BANKCARDS - CHECKS - DISCOVER CARDS WELCOME!

WOW! DS/DD DISKS .54 EACH!

GENUINE TI JOYSTICKS $10 PER PAIR!
(with this flyer only - regularly $29.95 - while supply lasts)

PICK UP YOUR COPY OF OUR CATALOG SOON

WE WILL BUY ANY TI HARDWARE OR SOFTWARE YOU NO LONGER NEED - CALL!

WE CUSTOM BUILD IBM COMPATIBLE COMPUTERS & TAKE TI ITEMS IN TRADE.

FEATURING PANASONIC & STAR MICRONICS PRINTERS FROM $189!

NEW AND USED TI99/4A COMPUTERS AVAILABLE!

EXPANSION SYSTEMS AVAILABLE - NEW AND USED!

* HUGE SOFTWARE INVENTORY - MORE IN STOCK THAN EVER BEFORE! *
CALL US FOR TECHNICAL HELP. WE WILL HELP YOU WITH YOUR PROBLEMS. WE WILL TRY TO MEET OR BEAT ANYBODY'S PRICES. REMEMBER THAT WE ARE HERE TO HELP IF YOU HAVE A QUESTION OR PROBLEM. WE WANT YOUR BUSINESS AND WE'LL PROVE IT!

OUR 21ST YEAR IN BUSINESS IN MILWAUKEE AT 27TH & NATIONAL!

TED, GENE, MIKE, JIM AND JERRY

1/88