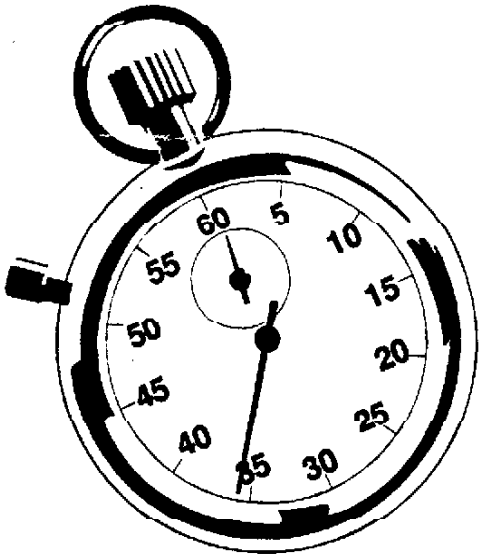
TI-D-BITS

PHILADELPHIA AREA USERS GROUP NEWSLETTER
COVERING THE TI99/4A
AND MYARC 9640 COMPUTERS

JANUARY 1993

Volume 13 Number 1

TIME IS RUNNING OUT



DON'T LOOSE OUT WHEN YOUR

MEMBERSHIP EXPIRES

SEND IT IN TODAY OR BRING IT IN

AT OUR NEXT MEETING

The Philadelphia Area TI-99/4A Users' Group meets twice a month. On the second and forth saturday of the month at the Church of the ATONEMENT, 6200 Greene St. Germantown (Corner of Greene St and Walnut Lane) at 10 A.M. We invite anyone that is interested in the TI-99/4A to visit us. Stop in and see what is available to you for your TI and how membership can benefit you!

Current executive board consists of:

PRESIDENT
Norm Sellers..... 215-353-0475

VICE PRESIDENT
Allan Silversteen 215-885-7910

SECRETARY
Tim Coyne..... 215-947-5881

TREASURE
Don Arsenault..... 215-368-0446

Committees consists of:

TI-D-BITS
Ralph Field..... 215-362-2534
Tim Coyne..... 215-947-5881

LIBRARY
Rich Mascaro..... 215-441-4060

MEMBERSHIP
Ralph Field..... 215-362-2534

EDUCATION
Barry Traver
Allan Silversteen

EQUIPMENT
Allan Silversteen

PROGRAM
(OPEN)

REMEMBER to be considerate when calling any of the above people. Limit your calls to the early evening hours. (6pm to 9pm)

The opinions expressed herein are those of the individual authors and are not necessarily those of the Philadelphia Area TI-99/4A Users' Group or its officers. Nor is the Philadelphia Area TI-99/4A Users' Group or any of its officers responsible for any damage, inconvenience, or loss which may result as a consequence of the use of any written material herein.

TI-D-BITS is published monthly by the Philadelphia Area TI-99/4A Users' Group, c/o Ralph E. Field, 603 N. Broad St., Lansdale, Pa. 19446. All material herein may be reprinted freely by other non-profit User Groups, (unless otherwise stated), as long as proper credit is given to both source and author. Contributions are encouraged, but no payment is made. Editorial, advertising, and classified , copy MUST be in by the LAST day of the previous month. You can either mail your copy to: TI-D-BITS, The Philadelphia Area TI-99/4A Users' Group, c/o Ralph E. Field, 603 N. Broad St., Lansdale, Pa. 19446 or send it via modem by contacting Ralph E. Field at (215)-362-2534. If your piece contains any, diagrams, charts, or code, send a paper copy AT FINAL PUBLICATION SIZE.

The editor of TI-d-Bits or the **executive board of The Philadelphia area TI-99/4a Users' Group reserve the** right to reject any material submitted for publication for any reasons.

The Philadelphia Area TI-99/4A Users' Group's program library is available to all active members at NO CHARGE for copying to your disk. A charge of $2.00 per disk is made for club supplied disks for members. Non members may obtain copies of the library for a fee of $5.00 per disk. A catalog of the library's contents is given to all new members upon request and updates will appear in this publication from time to time. To obtain material from the library, contact the librarian for the best procedure to obtain your requests.

A SYSTEM SEARCH PROGRAM

By Ed Hall
Taken from CPUG Newsletter

What was the name of that program? Seems like it had SEARCH in the name, but that wasn't the whole name. Well, let's see... FIND would work if I knew the whole name, and SYStem SEARCH program I wrote... THAT'S IT!

And here it is so others can use it too. This program is for those who have multiple subdirectories and drives. It is set up to search for partial names so you can find all occurrences of substrings within filenames.

In order to "customize" it for your system, sey up the first data line so it contains the basic drives of your system. In the listing I show floppies 1 through 4 and RAMDISK 5 as well as hard drives 1 and 2. If one of these drives is empty, the error routine will skip it, however this will be slow. Alternately, a disk can be placed in the drive. Once running all subdirectories are picked up and placed in the array so that each will be checked. The subdirectories are checked by level. This may seem strange at first since the first level of each drive is checked before the second level is started, which causes the program to skip back and forth between the hard drives.

When the program is run it prompts for a search string. All filenames available to the system are searched for an occurrence of the search string withen them. If a match is found, the path and filename information is displayed on the screen.

THE PROGRAM LISTING:

```
100 DIM DEVICES$(200):: A,B-
0
110 INPUT "SEARCH STRING? ":
SR$
120 A=A+1 :: READ DEVICE$(A)
:: IF DEVICE$(A)"END" THEN 1
20
130 ON ERROR 130
140 B=B+1 :: IF DEVICE$(B)="
END" THEN 230
150 OPEN #1:DEVICE$(B),INTER
NAL,INPUT ,FIXED
160 INPUT #1:B$,D,E,F
170 INPUT #1:B$,D,E,F
180 IF B$="" THEN 220
190 IF ABS(D)=6 THEN GOSUB 2
60
200 IF POS(B$,SR$,1)>0 THEN
PRINT DEVICE$(B),B$;
205 IF ABS(D)=6 THEN PRINT T
AB(25);"<D>" ELSE PRINT " "
210 GOTO 170
220 CLOSE #1 :: GOTO 130
230 END
240 DATA DSK1.,DSK2.,DSK3.,D
SK4.,DSK5.,WDS1.,WDS2.
250 DATA END
260 DEVICE(A+1)=DEVIVE$(A)
270 DEVICE(A)=DEVICE$(B)&B$
&"."
280 A=A+1
290 RETURN
```

****

INCOME TAX HELPER

Taken from:
LONG ISLAND NEWSLETTER
Modified by Bob DeVilbiss
Corrections made by:
Ralph Field - TI-D-BITS

The following is a list of the symbols and descriptions in case you want to modify the program more to your needs:

M	Maximum number of catagories
M0	Maximum number of data reads
M1	Number of income categories
C1$()	Master category code array
D1$()	Master category discription array
T$	Income/Deduction code
C$	Transaction category code
D	Transaction amount
S$	Transaction description
T1	Subtotal Income-Deduction
T2	Total Income/Deduction

THE PROGRAM LISTING:

```
100 CALL CLEAR
110 OPEN #1:"PIO"
120 PRINT "INCOME TAX RECORD
ING PROGRAM"
130 PRINT
140 PRINT "DATA STATEMENTS S
TART WITH"
150 PRINT "LINE NUMBER 1990"
160 PRINT
170 PRINT "DO YOU WANT TO SE
E THE"
180 PRINT "INSTRUCTIONS? (Y
OR N)"
190 INPUT A$
200 CALL CLEAR
210 IF A$="N" THEN 840
220 PRINT
230 PRINT "THIS PROGRAM INIT
IALIZE THE"
240 PRINT "VARIOUS INCOME/DE
DUCTIONS"
250 PRINT "CATEGORIES.  OUTPU
TIS"
260 PRINT "PRODUCED IN SEPAR
ATE"
270 PRINT "SECTIONS FOR INCO
ME AND"
280 PRINT "DEDUCTIONS.  SUBTO
TALS AND"
290 PRINT "TOTALS ARE PRODUC
ED FOR ALL"
300 PRINT "CATEGORIES."
310 PRINT
320 PRINT "ALL DATA IS ENTER
ED USING"
330 PRINT "-DATA-STATEMENTS.
"
340 PRINT "EXAMPLE:"
350 PRINT "DATA I,W,13.45,EM
PLOYER 1 (INCOME,WAGES,AMO
UNT,SOURCE)"
360 PRINT
370 PRINT "PRESS ENTER TO CO
NTINUE"
380 INPUT G$
390 PRINT "INCOME ITEMS ARE:
"
400 PRINT " W,WAGES"
410 PRINT " P,PENSION"
420 PRINT " TR,TAX RETURN"
430 PRINT " I,INTREST"
440 PRINT " D,DIVIDENDS"
450 PRINT " R,RENT/ROYALTY"
455 PRINT " S,SOCIAL SECURI
TY BENFITS"
460 PRINT " O,OTHER INCOME"
470 PRINT
480 PRINT "DEDUCTION ITEMS A
RE:"
490 PRINT " C,CONTRIBUTIONS
"
500 PRINT " I,INTREST"
510 PRINT " T,TAXES PAID"
520 PRINT " MD,MEDICAL/DENTI
CAL"
530 PRINT " CT,CASUALTY/THEF
T"
540 PRINT " MI,MISC EXPENSE"
550 PRINT " O,OTHER EXPENSE
"
560 PRINT
570 INPUT "DATA STARTS WITH
LINE 1990 PRESS ENTER":A$
590 PRINT "DO YOU WANT A PRI
NTOUT OF"
600 PRINT "THESE INSTRUCTIO
S? (Y OR N)"
620 IF A$="N" THEN 840
630 PRINT #1:"THIS PROGRAM
INITIALIZES THE VARIOUS INCO
ME/DEDUCTIONS CATEGORIES"
640 PRINT #1:"OUTPUT IS PROD
UCED IN SEPARATE SECTIONS"
650 PRINT #1:"FOR INCOME AND
DEDUCTIONS.  SUBTOTALS AND"
660 PRINT #1:"TOTALS ARE PRO
DUCED FOR ALL CATEGORIES."
670 PRINT #1:
680 PRINT #1:"ALL DATA IS EN
TERED USING -DATA-STATEMENTS
690 PRINT #1:"EXAMPLE:"
700 PRINT #1:"DATA I,W,13.45
,EMPLOYER 1"
710 PRINT #1:
720 PRINT #1:"INCOME ITEMS A
RE:"
730 PRINT #1:" W,WAGES P,
PENSION TR,TAX RETURN I,IN
TEREST"
735 PRINT #1:" D,DIVIDENDS
 ,R,RENT/ROYALITY O,OTHER I
NCOME"
740 PRINT #1:
750 PRINT #1:"DEDUCTIONS ITE
MS ARE:"
760 PRINT #1:" C,CONTRIBUT
IONS I,INTEREST T,TAXES PA
ID"
770 PRINT #1:" MD,MEDICAL/D
ENTAL CT,CAAUALTY/THEFT M,
MISC EXPENCE "
780 PRINT #1:" O,OTHER EXP
ENCE"
790 PRINT #1:
800 PRINT #1:"DATA ENTRIES S
TART AT LINE #1990.  DATA ST
ATEMENT (DATA END) MUST FOLL
OW"
```

```
810 PRINT #1:"LAST DATA ENTR
Y."
820 PRINT #1: : : : :
830 INPUT G$
840 CALL CLEAR
850 REM INCOME TAX RECORDING
PROGRAM
860 PRINT "INCOME TAX RECORD
ING PROGRAM"
870 REM **DATA INITIALIZATIO
N**
880 M=15
890 M0=10000
900 M1=8
910 DIM C1$(15)
920 DIM D1$(15)
930 C1$(1)="W"
940 D1$(1)="WAGES (1040 LINE
 7)"
950 C1$(2)="I"
960 D1$(2)="INTEREST INCOME
(LINE 8) & (SCHEDULE B)"
970 C1$(3)="D"
980 D1$(3)="DIVIDEND INCOME
(LINE 10) & (SCHEDULE B)"
990 C1$(4)="TR"
1000 D1$(4)="TAX REFUND (LIN
E 11"
1010 C1$(5)="P"
1020 D1$(5)="PENSION BENEFIT
S (LINE 16a)"
1030 C1$(6)="R"
1040 D1$(6)="RENT/ROYALTY I
NCOME (LINE 17) & SCHEDULE
 E)"
1050 C1$(7)="S"
1060 D1$(7)="SOCIAL SECURITY
 BENEFITS)"
1062 C1$(8)="O"
1064 D1$(8)="OTHER INCOME"
1070 C1$(9)="C"
1080 D1$(9)="CONTRIBUTIONS (
SCHEDULE A)"
1090 C1$(10)="I"
1100 D1$(10)="INTEREST EXPEN
CES (SCHEDULE A)"
1110 C1$(11)="T"
1120 D1$(11)="TAXES PAID (SC
HEDULE A)"
1130 C1$(12)="MD"
1140 D1$(12)="MEDICAL/DENTAL
1150 C1$(13)="CT"
1160 D1$(13)="CASULTY/THEFT
(SCHEDULE A)"
1170 C1$(14)="MI"
1180 D1$(14)="MISC EXPENSE (
SCHEDULE A)"
1190 C1$(15)="O"
1200 D1$(15)="OTHER EXPENCES
"
1210 REM *INCOME CATEGORIES
ARE FIRST 8 POSITIONS OF THE
 ARRAY*
1220 REM END OF CATEGORY ARR
AY INPUT*
1230 REM PRINT OF INCOME ITE
MS - BY CATEGORIES*
1240 PRINT "ALIGN TO TOP OF
PAGE "
1250 PRINT
1260 PRINT "PRESS <ENTER> TO
 CONTINUE"
1270 INPUT G$
1280 PRINT #1:CHR$(14);"
   * INCOME TAX HELPER"
1290 PRINT #1: :
1310 PRINT #1: :
1320 PRINT #1:"**************
******* INCOME *************
*******
1330 PRINT #1:
1340 FOR J=1 TO M1
1360 PRINT #1:D1$(J)
1370 FOR I=1 TO M0
1380 READ T$
1390 IF T$="END" THEN 1470
1400 READ C$,D,S$
1410 IF T$<>"I" THEN 1460
1420 IF C$<>C1$(J)THEN 1460
1440 PRINT #1:TAB(5);S$;TAB(
50);D
1450 T1=T1+D
1460 NEXT I
1480 PRINT #1:TAB(51);"-----
---"
1490 PRINT #1:TAB(42);"TOTAL
";TAB(50);T1
1500 T2=T2+T1
1530 PRINT #1:"-------------
---------------------------"
-"
1540 RESTORE
1550 NEXT J
1560 RESTORE
1580 PRINT #1:TAB(36);"TOTAL
INCOME";TAB(50);T2
1590 T2=0
1600 T1=0
1610 J0=J
1620 REM ** END OF INCOME-ST
ART DEDUCTIONS PRINT **
1630 REM PRINT "ALIGN TO TOP
OF NEXT PAGE PRESS ENTER KEY
OR"
1640 REM INPUT Z$
1650 PRINT "****** DEDUCTION
S******"
1660 PRINT "ALIGN TO NEXT PA
GE AND PRESS ENTER";X$
1670 INPUT X$
```

```
1680 PRINT #1:"**************
******* DEDUCTIONS *********
***********"
1690 PRINT #1:
1700 FOR J=J0 TO M
1710 PRINT D1$(J)
1720 PRINT #1:D1$(J)
1730 FOR I=1 TO M0
1740 READ T$
1750 IF T$="END" THEN 1840
1760 READ C$,D,S$
1770 IF T$<>"D" THEN 1820
1780 IF C$<>C1$(J)THEN 1820
1790 PRINT TAB(5);S$;TAB(50)
;D
1800 PRINT #1:TAB(5);S$;TAB(
50);D
1810 T1=T1+D
1820 NEXT I
1830 PRINT TAB(42);"TOTAL";T
AB(50);T1
1840 PRINT #1:TAB(51);"-----
---"
1850 PRINT #1:TAB(42);"TOTAL
";TAB(50);T1
1860 PRINT #1:"-------------
--------------------------"
1870 PRINT #1:"-------------
-------"
1880 T2=T2+T1
1890 T1=0
1900 RESTORE
1910 NEXT J
1920 PRINT TAB(36);"TOTAL DE
DUCTIONS";TAB(50);T2
1930 PRINT #1:TAB(36);"TOTAL
DEDUCTIONS";TAB(50);T2
1940 T2=0
1950 T1=0
1960 PRINT #1:"=============
============================
================"
1970 REM ** DATA ENTRIES FOR
INITILIZATION **
1980 REM DATA ENTRIES FOLLOW
1990 DATA I,W,32000.00,EMPLO
YER
2000 DATA I,D,50.00,CREDIT U
NION
2010 DATA I,I,150.00,SAVINS
2020 DATA I,O,10.00,MELON
2030 DATA I.TR.50.00.REFUND
2040 DATA I,P,500.00,PENS
2050 DATA I,R,200.00,RPY
2060 DATA I,S,25.00,SOCIAL
2070 DATA D,C,100.00,CON
2080 DATA D,I,100.00,INT
2090 DATA D,T,500.00,TAX
2100 DATA D,MD,450.00,MED
2110 DATA D,CT,800.00,CAT
2120 DATA D,MI,200.00,MISS
2130 DATA D,O,75.00,OTHER
3000 DATA END
```

****

Execute "NEW" from Inside Program

This item has appeared in several user groups newsletters. It is written by Jonathan D. Guidry.

The "NEW" command is very valuable to the programmer. However, it only executes in immediate mode. Have you ever wondered how to make the computer clear the memory from inside a program?

If you tried to use the word NEW after a line number, you got a *COMMAND ILLEGAL IN A PROGRAM* error. The key word NEW, in immediate mode, resets memory positions -31868 and -31952 to FF,FF,FF,FF in hexadecimal.

Here is an example of the usage of reseting memory positions -31868 and -31952. If you plan to use the example below, you must have a memory expansion and E/A, Mini-Memory or Extended BASIC.

```
CALL INIT
CALL LOAD(-31868,255,255,255,255)
CALL LOAD(-31952,255,255,255,255)
```

In Extended BASIC, chain the three statements togther using double colons. If you have a program in memory at the time these are executed, it will be erased. Try it.

These statements are most useful at the end of a program. For example, if you place these statements at the end of a directory program called LOAD in Extended BASIC, it will give you back the prompt after it is finished, and the memory will be erased.

These three statements can also guard that top-secret program you've been working on by erasing the memory if the correct password isn't given at the begining of the program. It also prevents snooping. Also, do not forget that they can be put anywhere in a program.

Guidry encourages correspondence. He can be reached at 731-H Creighton Dr. New Iberis, LA 70560.

TI-BASE Tutorial 5.2

NorthCoast 99'ers (C) Martin A. Saoley

```
*************************************

     DOCASE
       CASE SEL = 0
        WRITE 18,13,"Have a nice day"
        REPLACE ? WITH 1
        BREAK
       CASE SEL = 1
         DO DSK2.PRSTR
        BREAK
       CASE SEL = 2
         DO DSK2.LBLS5
        BREAK
       CASE SEL = 3
         DO DSK2.FNDPRNT1
        BREAK
       CASE SEL = 4
         DO DSK2.EDFL1
        BREAK
       CASE SEL = 5
         DO DSK2.APFL1
        BREAK
     ENDCASE
RETURN
*
* SLCASE          Save as SLCASE/C
* ******  Case Selection 12/1/88
*
 ***********************************

SET PAGE=000
SET LINE=80
CLEAR
LOCAL TEMP C 60
LOCAL BLNK C 1
WRITE 10,4,"Set Printer + press ENTER"
READ 10,30,TEMP
CLEAR
WRITE 10,12,"Printing Roster"
SORT OFF
TOP
   REPLACE TEMP WITH "LE              ";
     ! "              ";
     ! "       ** NorthCoast Roster **"
     PRINT TEMP
     PRINT BLNK
SET LINE=134
PRINT ALL
SET LINE=80
   REPLACE TEMP WITH " L@ "
   PRINT TEMP
CLEAR
RETURN
*                        Version 1.02
* PRSTR       Save as PRSTR/C
* *****   Print Roster  12/03/88
*
 ***********************************

SET PAGE=000
CLEAR
LOCAL TEMP C 40
LOCAL BLNK C 1
WRITE 10,4,"Set Printer + press ENTER"
READ 10,30,TEMP
CLEAR
WRITE 10,12,"Printing Labels"
SORT ON ZP
TOP
WHILE .NOT. (EOF)
   REPLACE TEMP WITH "LE              ";
     ! "          Exp. Date " ! XP
     PRINT TEMP
     PRINT BLNK
   REPLACE TEMP WITH TRIM(FN) ! " ";
             ! MI ! " " ! LN
     PRINT TEMP
     PRINT SA
   REPLACE TEMP WITH TRIM(CT) ! " ";
             ! ST ! ". " ! ZP
     PRINT TEMP
     PRINT BLNK
   MOVE
   ENDWHILE
CLEAR
RETURN
*
* LBLS5   Save as LBLS5/C 12/01/88
* *****       Prints all Labels
*
 ***********************************

LOCAL SEL2 N 3 0
LOCAL MORE N 3 0
REPLACE MORE WITH 1
WHILE (MORE > 0)
  TOP
  DO DSK2.INFSCR2
 WRITE 19,6,"ENTER 1-5"
 READ 19,17,SEL2
     CLEAR
    WHILE (.NOT.(EOF)) .AND.;
      (NM <> SEL2)
     MOVE
    ENDWHILE
   IF (NM = SEL2)
     DO DSK2.DISPNA1
  ENDIF
 WRITE 6,6,"FIND  MORE  NAMES"
 WRITE 8,6,"0 = No     1 = Yes"
 READ 8,25,MORE
 CLEAR
ENDWHILE
RETURN
*
* FNDPRNT1  Save as FNDPRNT1/C
* ********              1:/29/88
*
 ***********************************
```

TIPS FROM THE TIGERCUB

No. 66

Tigercub Software
156 Collingwood Ave.
Columbus, OH 43213

****

My three Nuts & Bolts disks, each containing 100 or more subprograms, have been reduced to $5.00 each. I am out of printed documentation so it will be supplied on disk.

My TI-PD library now has well over 500 disks of fairware (by author's permission only) and public domain, all arranged by category and as full as possible, provided with loaders by full program name rather than filename, Basic programs converted to XBasic, etc. The price is just $1.50 per disk(!), post paid if at least eight are ordered. TI-PD catalog #5 and the latest supplement is available for $1 which is deductible from the first order.

In Tips #65 I said that the TI could calculate to 14-digit accuracy, rather than the 8-digit accuracy of a PC. Actually the number in memory is calculated to 13- or 14-digit accuracy, depending on the number, but is rounded to 10 digits on the screen display, or shown in exponential notation if the number is extremely large or extremely small. If you want to see the complete number, this routine will show the normal screen display and the full number in memory. To see the complete range of numbers our little TI can handle, try inputting -
-9.99999999999999E127 and
-1.0000000000000E-128 and
1.0000000000000E-128 and
9.99999999999999E127.

```
100 OPEN #1:"DSK1.INT2",INTE
RNAL,RELATIVE,UPDATE
110 INPUT N
120 PRINT #1,REC 1:N
130 INPUT #1,REC 1:N$
140 X=ASC(SEG$(N$,1,1)):: Y=
ASC(SEG$(N$,2,1)):: IF Y>99
THEN Y=256-Y :: N$=SEG$(N$,1
,1)&CHR$(Y)&SEG$(N$,3,255)
150 FOR J=2 TO LEN(N$):: X$=
STR$(ASC(SEG$(N$,J,1)))
160 IF ASC(SEG$(N$,J,1))<10
THEN X$="0"&X$
170 P$=P$&X$ :: NEXT J
180 IF X<63 THEN Y$="."&RPT$
("00",63-X)&P$ :: GOTO 230
190 IF X>191 THEN Y$="."&RPT
$("00",X-192)&P$ :: GOTO 230
200 IF X>185 THEN Y$=SEG$(P$
,1,14-(X-185)*2)&"."&SEG$(P$
,14-(X-185)*2+1,255):: GOTO
230
210 Y$=SEG$(P$,1,(X-63)*2)&"
."&SEG$(P$,(X-63)*2+1,255)
220 IF ASC(Y$)=48 THEN Y$=SE
G$(Y$,2,255)
230 IF N<0 THEN Y$="-"&Y$
240 PRINT TAB(2);N :: PRINT
TAB(3);Y$ :: P$="" :: GOTO 1
10
```

But even the smart little TI has its limits. Try this- X=2/3-1/3-1/3 :: PRINT X. See the TI User's Reference Guide page III-13 for the explanation of all this.

Solving an equation such as X^X/X-X would be very difficult to solve by mathematical means, but our computer can find the answer quickly by systematic trial and error, to the 14-point limit of its accuracy.

```
100 DISPLAY AT(3,1)ERASE ALL
:"This program will solve ev
enthe most difficult equatio
nswith one variable."
110 DISPLAY AT(6,1):"Put you
r own equation in  line 21
0, using A for the  known v
alue and X for the  unknown
 value."
120 DISPLAY AT(24,6):"PRESS
ANY KEY" :: DISPLAY AT(24,6)
:"press any key" :: CALL KEY
(0,K,S):: IF S=0 THEN 120
130 CALL CLEAR
140 DISPLAY AT(8,1):"KNOWN V
ALUE? " :: ACCEPT AT(8,14):C
150 X=1 :: DISPLAY AT(12,1):
""
160 GOSUB 210
170 IF A<C THEN Y=X :: X=X*2
 :: GOSUB 210 :: GOTO 170 EL
SE 190
180 IF A>C THEN Y=X :: X=X/2
 :: GOSUB 210 :: GOTO 180
190 Z=(ABS(X-Y))/2 :: Y=X ::
 IF A<C THEN X=X+Z ELSE X=X-
Z
200 GOSUB 210 :: GOTO 190
210 A=X^X/X-X/2
220 IF A<C THEN DISPLAY AT(1
1,5):X ELSE IF A>C THEN DISP
LAY AT(13,5):X
230 IF A=C OR A=B THEN DISPL
AY AT(12,5)ERASE ALL:X :: GO
TO 140 ELSE B=A :: RETURN
```

In Recreational and Educational Computing, published 8 times a year at 909 Violet Terrace, Clarks Summit PA 18411, $27 per year, I found a neat routine to find the greatest common divisor and least common multiple of any two numbers - so I converted it to TI Basic and modified to do the same with multiple numbers.

```
100 CALL CLEAR :: PRINT "PRO
GRAM TO FIND THE GREATESTCOM
MON DIVISOR AND LEAST  COM
MON MULTIPLE OF ANY  NUM
BER OF NUMBERS.": :
110 DIM N(100)
120 PRINT "INPUT ZERO WHEN F
INISHED": :
130 T=T+1 :: INPUT "NUMBER "
&STR$(T)&"? ":N(T):: IF N(T)
=0 THEN 140 ELSE IF N(T)<>IN
T(N(T))OR N(T)<1 THEN T=T-1
:: GOTO 130 ELSE 130
140 AA=N(1):: GCD=N(2)
150 GOSUB 170 :: FOR J=3 TO
```

```
T-1 :: AA=N(J):: GCD=ABS(GCD
):: GOSUB 170 :: NEXT J
160 GOTO 210
170 R=AA-INT(AA/(GCD+ABS(GCD
=0)))*GCD
180 IF R<2 THEN GCD=R+GCD*(1
-R):: GCD=GCD*ABS(GCD>0)+ABS
(GCD=0):: GOTO 200
190 AA=GCD :: GCD=R :: GOTO
170
200 RETURN
210 PRINT "THE GREATEST COMM
ON DIVISOR OF YOUR";T-1;"NUM
BERS IS";GCD
220 L=N(1)*N(2)/GCD :: FOR J
=3 TO T-1
230 IF L/N(J)<>INT(L/N(J))TH
EN L=L*N(J)
240 NEXT J
250 LL=L/2 :: FOR J=1 TO T-1
 :: IF LL/N(J)<>INT(LL/N(J))
THEN J=T-1 :: GOTO 270
260 NEXT J :: L=LL :: GOTO 2
50
270 PRINT "AND THE LOWEST CO
MMON       MULTIPLE IS";L
```

Joy Warner called from the L.A. group, and mentioned that it would be nice to have a program to print out a page of math problems, and a page of answers. So here is one that will randomly create any number of either addition or subtraction problems, within any specified range of numbers, and output the desired number of copies to a printer in two columns of expanded print, numbered in sequence, plus a numbered answer sheet to make it easy for the teacher.

```
100 DISPLAY AT(1,4)ERASE ALL
:"MATH PROBLEM PRINTER" !by
Jim Peterson
110 DIM A(200),H(200),L(200)
:: OPEN #1:"PIO" :: PRINT #1
:CHR$(27)&"@"&CHR$(27)&"W"&C
HR$(1);
120 M$(1)="ADDITION" :: M$(2
)="SUBTRACTION" :: D$(1)="+
" :: D$(2)="- " :: ON$=CHR$(
27)&"-"&CHR$(1):: OFF$=CHR$(
27)&"-"&CHR$(0)
130 DISPLAY AT(3,1):"Do you
want?":"":"1. "&M$(1):"2. "&
M$(2):: ACCEPT AT(3,14)VALID
ATE("12")SIZE(1)BEEP:C
140 DISPLAY AT(8,1):"Range o
f numbers?":"From":"To" :: A
CCEPT AT(9,6)VALIDATE(DIGIT)
BEEP:LN :: ACCEPT AT(10,4)VA
LIDATE(DIGIT)BEEP:HN :: IF L
N>=HN THEN 140 ELSE HN=HN-LN
150 DISPLAY AT(13,1):"How ma
ny problems?" :: ACCEPT AT(1
3,20)VALIDATE(DIGIT)BEEP:P
160 DISPLAY AT(15,1):"How ma
ny copies?" :: ACCEPT AT(15,
18)VALIDATE(DIGIT)BEEP:CC
170 FOR J=1 TO P :: GOSUB 29
0 :: H(J)=N1 :: L(J)=N2
180 IF C=1 THEN A(J)=H(J)+L(
J)ELSE A(J)=H(J)-L(J)
190 NEXT J
200 FOR J=1 TO CC :: GOSUB 3
10 :: FOR K=1 TO P STEP 2
210 T1$=STR$(K)&".    "&STR$
(H(K)):: T2$=STR$(K+1)&".
 "&STR$(H(K+1))
220 PRINT #1:TAB(15-LEN(T1$)
);T1$;TAB(35-LEN(T2$));T2$
230 T1$=D$(C)&STR$(L(K)):: T
2$=D$(C)&STR$(L(K+1))
240 PRINT #1:TAB(15-LEN(T1$)
);ON$&T1$&OFF$&RPT$(" ",20-L
EN(T2$))&ON$&T2$&OFF$
250 PRINT #1:"":"":"" :: IF
K/19=INT(K/19)THEN PRINT #1:
CHR$(12);
260 NEXT K :: PRINT #1:CHR$(
12);:: NEXT J
270 PRINT #1:TAB(16);"ANSWER
S":"":""
280 FOR J=1 TO P STEP 2 :: P
RINT #1:TAB(6);STR$(J)&".  "
;A(J);TAB(26);STR$(J+1)&".
";A(J+1):: NEXT J :: STOP
290 RANDOMIZE :: N1=INT(RND*
HN+LN):: N2=INT(RND*HN+LN)::
 IF N1=N2 THEN 290
300 IF C=2 AND N2>N1 THEN T=
N2 :: N2=N1 :: N1=T :: RETUR
N ELSE RETURN
310 PRINT #1:"      "&M$(C)
&" PROBLEM PRINTER":"":"":""
:"" :: RETURN
```

And this one will do the same with multiplication problems.

```
100 DISPLAY AT(1,4)ERASE ALL
:"MULTIPLICATION PROBLEMS":"
          PRINTER" !by Jim P
eterson
110 DIM A(200),H(200),L(200)
:: OPEN #1:"PIO" :: PRINT #1
:CHR$(27)&"@"&CHR$(27)&"W"&C
HR$(1);
120 ON$=CHR$(27)&"-"&CHR$(1)
:: OFF$=CHR$(27)&"-"&CHR$(0)
130 DISPLAY AT(8,1):"Range o
f multiplicand?":"From":"To"
 :: ACCEPT AT(9,6)VALIDATE(D
IGIT)BEEP:L1 :: ACCEPT AT(10
,4)VALIDATE(DIGIT)BEEP:H1 ::
 IF L1>=H1 THEN 130 ELSE H1=
H1-L1
140 DISPLAY AT(12,1):"Range
of multiplier?":"From":"To"
:: ACCEPT AT(13,6)VALIDATE(D
IGIT)BEEP:L2 :: ACCEPT AT(14
,4)VALIDATE(DIGIT)BEEP:H2
150 IF L2>=H2 THEN 140 ELSE
R=LEN(STR$(H2)):: H2=H2-L2
160 DISPLAY AT(16,1):"How ma
ny problems?" :: ACCEPT AT(1
6,20)VALIDATE(DIGIT)BEEP:P
170 DISPLAY AT(18,1):"How ma
ny copies?" :: ACCEPT AT(18,
18)VALIDATE(DIGIT)BEEP:CC
180 FOR J=1 TO P :: GOSUB 31
0 :: H(J)=N1 :: L(J)=N2
190 A(J)=H(J)*L(J)
200 NEXT J
210 FOR J=1 TO CC :: GOSUB 3
20 :: FOR K=1 TO P STEP 2
220 T1$=STR$(K)&".    "&STR$
(H(K)):: T2$=STR$(K+1)&".
 "&STR$(H(K+1))
230 PRINT #1:TAB(15-LEN(T1$)
);T1$;TAB(35-LEN(T2$));T2$
240 T1$="X "&STR$(L(K)):: T2
$="X "&STR$(L(K+1))
250 PRINT #1:TAB(15-LEN(T1$)
);ON$&T1$&OFF$&RPT$(" ",20-L
EN(T2$))&ON$&T2$&OFF$
260 FOR S=1 TO R+3 :: PRINT
#1:"" :: NEXT S
270 LC=LC+5+R :: RC=LC+5+R :
: IF RC>=60 AND K<P THEN PRI
NT #1:CHR$(12):: LC=5
280 NEXT K :: PRINT #1:CHR$(
12);:: NEXT J
290 PRINT #1:TAB(16);"ANSWER
S":"":""
300 FOR J=1 TO P STEP 2 :: P
```

```
RINT #1:TAB(3);STR$(J)&".  "
;A(J);TAB(23);STR$(J+1)&".
";A(J+1):: NEXT J :: PRINT #
1:CHR$(12):: STOP
310 RANDOMIZE :: N1=INT(RND*
H1+L1):: N2=INT(RND*H2+L2)::
 RETURN
320 PRINT #1:"        MULTIP
LICATION PROBLEMS":"":"":"":
"" :: LC=5 :: RETURN
```

And division -

```
100 DISPLAY AT(1,6)ERASE ALL
:"DIVISION PROBLEMS":"
    PRINTER" !by Jim Peterso
n
110 DIM A(200,2),H(200),L(20
0):: OPEN #1:"PIO" :: PRINT
#1:CHR$(27)&"@"&CHR$(27)&"W"
&CHR$(1);
120 DISPLAY AT(8,1):"Range o
f dividend?":"From":"To" ::
ACCEPT AT(9,6)VALIDATE(DIGIT
)BEEP:L1 :: ACCEPT AT(10,4)V
ALIDATE(DIGIT)BEEP:H1 :: IF
L1>=H1 THEN 120
130 DISPLAY AT(12,1):"Range
of divisor?":"From":"To" ::
ACCEPT AT(13,6)VALIDATE(DIGI
T)BEEP:L2 :: ACCEPT AT(14,4)
VALIDATE(DIGIT)BEEP:H2
140 IF L2>=H2 THEN 130 ELSE
R=LEN(STR$(INT(H1/H2)))*2 ::
 H2=H2-L2 :: H1=H1-L1
150 DISPLAY AT(16,1):"How ma
ny problems?" :: ACCEPT AT(1
6,20)VALIDATE(DIGIT)BEEP:P
160 DISPLAY AT(18,1):"How ma
ny copies?" :: ACCEPT AT(18,
18)VALIDATE(DIGIT)BEEP:CC
170 FOR J=1 TO P :: GOSUB 31
0 :: H(J)=N1 :: L(J)=N2
180 A(J,1)=INT(H(J)/L(J))::
A(J,2)=H(J)-A(J,1)*L(J)
190 NEXT J
200 FOR J=1 TO CC :: GOSUB 3
20 :: FOR K=1 TO P STEP 2
210 LC=LC+1 :: T1$=STR$(K)&"
.    "&RPT$(" ",LEN(STR$(L(K
))))&RPT$("_",LEN(STR$(H(K))
))
220 T2$=STR$(K+1)&".    "&RP
T$(" ",LEN(STR$(L(K+1))))&RP
T$("_",LEN(STR$(H(K+1))))
230 PRINT #1:TAB(15-LEN(T1$)
);T1$;TAB(35-LEN(T2$));T2$
240 T1$=STR$(L(K))&"|"&STR$(
H(K)):: T2$=STR$(L(K+1))&"|"
&STR$(H(K+1))
250 LC=LC+1 :: PRINT #1:TAB(
15-LEN(T1$));T1$;TAB(35-LEN(
T2$));T2$
260 FOR S=1 TO R+3 :: LC=LC+
1 :: PRINT #1:"" :: NEXT S
270 IF 66-LC<5+R AND K<P THE
N PRINT #1:CHR$(12):: LC=5
280 NEXT K :: PRINT #1:CHR$(
12);:: NEXT J
290 PRINT #1:TAB(16);"ANSWER
S":"":""
300 FOR J=1 TO P :: PRINT #1
:TAB(3);STR$(J)&".  ";A(J,1)
;"REMAINDER ";A(J,2):: NEXT
J :: PRINT #1:CHR$(12):: STO
P
310 RANDOMIZE :: N1=INT(RND*
H1+L1):: N2=INT(RND*H2+L2)::
 RETURN
320 PRINT #1:"          DIV
ISION PROBLEMS":"":"":"":""
:: LC=5 :: RETURN
```

Bud Wright wrote this one for Irwin Hott, so he could listen to lower case text with the Speech Synthesizer. Imbed it with ALSAVE, access it with CALL LINK("CAPS",A$) and it will instantly convert any lower case letters to upper case. I found it invaluable in writing keyword search programs.

```
* CAPS/S BY BUD WRIGHT
* VERSION 1.1 10/17/86
STRREF EQU >2014
STRASG EQU >2010
MREG   BSS 32
STRBUF BYTE 255
       BSS 255
       DEF CAPS
CAPS   LWPI MREG
       CLR R0
       LI R1,1
       SETO @STRBUF
       LI R2,STRBUF
       BLWP @STRREF
       MOVB @STRBUF,R2
       SRL R2,8
       JEQ CAPOUT
       LI R1,STRBUF+1
CAPS2  MOVB *R1,R3
       SRL R3,8
       CI R3,96
       JGT CAPS1
CAPS3  SWPB R3
       MOVB R3,*R1+
       DEC R2
       JNE CAPS2
       CLR R0
       LI R1,1
       LI R2,STRBUF
       BLWP @STRASG
CAPOUT LWPI >83E0
       B @>006A
CAPS1  CI R3,122
       JGT CAPS3
       AI R3,-32
       JMP CAPS3
       END
```

Memory full,

Jim Peterson