

~~~~~ TI-101 ~~~~~

## OUR 4/A UNIVERSITY

by Jack Sughrue  
Box 459  
E. Douglas MA 01516

### #5 McGUFFEY'S

Before we spend a class on the TI textbooks I mentioned at the end of our last session, I'd like to mention McGUFFEY'S ECLECTIC READERS from the 1890's. They were the major source of formal, academic learning for young scholars 100 years ago. Today's McGuffey is Don Shorock. Let's analyze a bit of Shorock's eclecticism of the 1990's.

Ms. Bronte, I already gave you his address in my notes three or four classes ago.

Very well. He can be reached for these educational goodies - mostly fairware (and let's hope I don't have to explain *that* again) at P.O.Box 501, Great Bend, KS 67530. Got that? Good. Now try not to interrupt with questions that have already been answered if you were paying attention during our other classes.

I'm going to be using the overhead for this lesson, as some of the intricacies of this educator's materials are fascinating and unusual.

Last session, Class, we had a couple questions from Mr. Shakespeare over there by the window. He said he had a nephew in junior high and two elementary school grandchildren. I think Mr. Shorock's eclectic disks will be of great help here.

First, let me mention that Mr. S has the most extraordinary data base structure built into his programs.

Second, let me put up the menu of his first disk on the overhead here. He has four educational fairware disks: EDUCATION #1, 2, 3, and INVENTIONS.

This is the first menu for #1:

- a) AMERICAN PRESIDENTS
- b) ENGLISH MONARCHS
- c) ANCIENT GREEKS & ROMANS
- d) JOYSTICK AMERICA
- e) WORLD MILEAGE
- f) STATES & CAPITALS (groups)
- g) SOLAR SYSTEM
- h) WORD MATH
- i) GAGGLES OF GEESE
- j) ESTIMATING TRIANGLES
- k) FACTORING
- l) AUDIO MATH
- m) CATALOG
- n) DOCUMENTATION
- o) EXIT

There are 12 programs, plus a chance to look at the catalog from the disk, plus a chance to read all the documentation. By using the alphabet instead of numbers, he is able to have the menu items lined up perfectly (as "10" and beyond would push everything one character to the right). Very neat is our Mr. Shorock. If we pressed "a" for the President

the 10-question activity. This should've been eliminated. That sort of problem does appear in some of these other activities, too.

However, Class, a great opportunity to learn all those new former Soviet Union and other Eastern European and Western Asian countries would be to change this program to include just these "new" countries for flash-card learning. Mr. Shorock has made a great base into which it is fairly easy to plug new data.

I see people peering at their watches. Before we go today and before I assign homework, I have a couple more overhead transparencies to show you, such as this projection of the menu from Education Disk #3: Misc.:

- A - STATES 50
- B - SEMAPHORE SIGNALING
- C - FAMOUS PHILOSOPHERS
- D - FAMOUS COMPOSERS
- E - CHEMICAL ELEMENTS
- F - GEOLOGY TERMS
- G - CANADIAN PROVINCE CAPITALS
- H - MEXICAN STATE CAPITALS
- I - FLAG QUIZ & DEMO
- J - PRES. INAUGURAL DATES
- K - SUBJECT/VERB AGREEMENT
- L - SHORTWAVE TUNING SIGNALS
- M - JULY 4TH DEMO
- N - CATALOG
- O - DOCUMENTATION
- P - EXIT

When I look at FAMOUS COMPOSERS, Class, I'm reminded of the time a group of music historians dug up Beethoven's grave. When they opened his coffin, up popped Beethoven, shouting, "What is the meaning of this? Can't you see I'm busy decomposing?"

L is interesting. The signature tunes which are played by different countries before they begin their shortwave broadcasts are played (Switzerland, Canada, Kuwait, South Africa, etc.). Once learned, there is a quiz, of course.

After all the playing and using and trying and testing, I finally found an error, Class. In STATES 50 the program says New Hampshire does not touch Massachusetts. It does. Not bad. One small mistake in four jam-packed disks of educational programs.

Most of the programs I think you can figure out from the titles. Now this last overhead. Has two menus shown on it; the main and the one by pressing 3 on the main:

- 1) USE PROGRAM
- 2) PRINT DOC
- 3) LOOK AT DOC
- 4) SEE DISK CATALOG

- 1) A word about Fairware
- 2) Why I wrote this program
- 3) How to use this program
- 4) Programming techniques  
segmented array items
- 5) Programming techniques  
randomly filled array
- 6) How LOOK AT DOC works

program another menu appears, as you can see on this transparency:

- 1) NAME YEAR
- 2) NAME PARTY
- 3) NAME PRESIDENT
- 4) NAME STATE
- 5) FOR QUIZ
- 6) LEAVE (to go back to main menu, which is nice, and only "d" and "j" are unable to within their activities)

If, at this point, we press "1" and type in "1962" at the cursor, we get the following:

"Year #2 of the Presidency of John Kennedy; 35th President; Democrat of Massachusetts; served 1961-1963."

Typing "1963" would give both Kennedy and Johnson (who served from 1963 to 1969).

Pressing "2" above will give you the listing of all the parties under which our Presidents have served: 1) Federalist 2) Democratic-Republican 3) National Republican 4) Democratic 5) Whig 6) Republican. And choosing "Whig," for example, will give you William Henry Harrison, 1841-1841; John Tyler, 1841-1845; Zachary Taylor, 1849-1850; and Millard Fillmore, 1850-1853.

When choosing NAME PRESIDENT and typing "John," you will get all the Johns: Adams, Adams, Tyler, Kennedy with all their accompanying info. So you can enter first OR last names and have the program seek out the proper data for you.

To enter STATE you must type in the full name, however, as the program will not accept MA or MASS for MASSACHUSETTS. When you type that full name, though, it will list its four Presidents and their biographical sketches. Entering the name of a state with no President will give you the cursor, just as typing in wrong info will.

Now, when you have mastered this info, you will have the QUIZ (a yes/no job): "Did Benjamin Harrison Rule in the year 1811? (No. His term, as we all know, was from 1889-1893.) [I don't like RULED, however, which is the reappearing term for SERVED that Mr. Shorock (probably a native of England) keeps using: Did Ronald Reagan Rule in 1818? (No. That's right, Class. Ronald Reagan RULED America from 1981 to 1989, though he may have behaved as if he RULED America in 1818).] Actually, I just listed the program and changed the RULE to SERVE, and it made the program so much better in our Democracy. Particularly if this program will be used with children.

The English Monarchs and Ancient Greek and Roman programs are structured similarly.

I don't intend to go through each of his delightful program packages for learners during our time today. Suffice it to say, Class, that you can see the amount of work that goes into a program like this and, by studying the program itself, the unlimited kinds of applications for which one may use these programs.

I would, however, like to delve into a couple of completely different educational programs here. WORD MATH deals with addition, subtraction, multiplication, and division as the bane of all elementary and junior high students: Word Problems. The answers may be typed as "SEVENTEEN" or "17," as in this problem: "Ellen has nine dandelions and Kent has eight marigolds. How many flowers do Ellen and Kent have altogether?" At the menu you may choose specific processes (addition) or all. A running score is kept (as with most of Shorock's games and quizzes), and a wrong answer is corrected and explained. At the end of this program a flashy countdown in words from 100 to 1 takes place, using the TI's

built-in wonders, and more options are given, including continuing the game.

JOYSTICK AMERICA is a geography game. Kind of a precursor, in a philosophical way, to Mr. S's highly successful AIR TAXI, his commercial venture which is a geographical masterpiece. I understand, Class, that he has a further development on even that one. When you write to him, ask. But J.A. has a golf-like scoring system. You're given a par (how many moves it should take you) to go from a random starting point in America (say Western Tennessee) to a random destination (say Ohio or Indiana). As you must move north and east in 3 moves here, you can judge how you are doing by the constantly updated "current location." Complex structure, simple execution.

Although we've analyzed just three learning activities on the first disk, you can already see that directions are kept to a minimum, partially through superb sub menus; the structure is simple; the pathways direct; the learning concrete; the adaptive possibilities endless.

A quick look at the transparency of Disk 2's menu (of math and geography activities only) will show you how Mr. S jumpacks these SSSD disks:

#### GEOGRAPHY GAMES

- 0) North American Cities (comparisons: which is further north? west?)
- 1) Largest Cities (Chicago is the largest city in what state?)
- 2) World Capitals (multiple choice)
- 3) US Mileage (which is closer / how far is it to?)
- 4) Map: Eastern US (does Maine touch Vermont?)
- 5) Map: Europe (does France touch Luxembourg?)
- 6) Map: Latin America (does Equador touch Chile?)

#### MATH GAMES

- 1) Patterns (math drill with wallpaper graphics)
- 2) More or Less (greater and lesser numbers)
- 3) Roman Numerals (teach, convert, quiz)
- 4) Chinese Numbers (teach, convert, quiz)
- 5) Tardis (strictly for us Dr. Who fans; requires SS and TEII)

PATTERNS is a flash-card-type arithmetic drill (3+17, 21X65, 14-11) on a solid background pattern that does not scroll when foreground "work area" does. The "wallpaper" changes and provides a nice 3-D effect. Score is kept as you go along; correct answers are given; a total is displayed at the end of each 10 questions before a new quiz with a new largest amount total being input again. Some toots and whistles here, also. Not a negative program.

CHINESE NUMBERS has three menu options: T - Teaches Chinese numbers (graphically) from 1 to 9999; I - Interprets by translating any Arabic number of your choice into Chinese; Q - Quiz Giver lets you set the maximum amount and gives you 10 problems to solve, scores, and autoloads at end. Like so many of the Shorrock activities, it is easy to get back to any part of the program (but not in some of these cases to get back to the main disk menu).

WORLD CAPITALS has a wonderful menu that includes 1) Latin America 2) Europe 3) Africa 4) Asia 5) Oceania 6) Entire World 7) TI Answers (instead of asks) 8) Show Off Everything 9) New Player (instead of the person who typed name at beginning of game, as this will give game competition opportunities)

I wanted to add one thing here, Mr. Shakespeare, if you use these disks with your young relatives. This program, unfortunately, sometimes repeats questions back to back (and to back again, in some cases) within

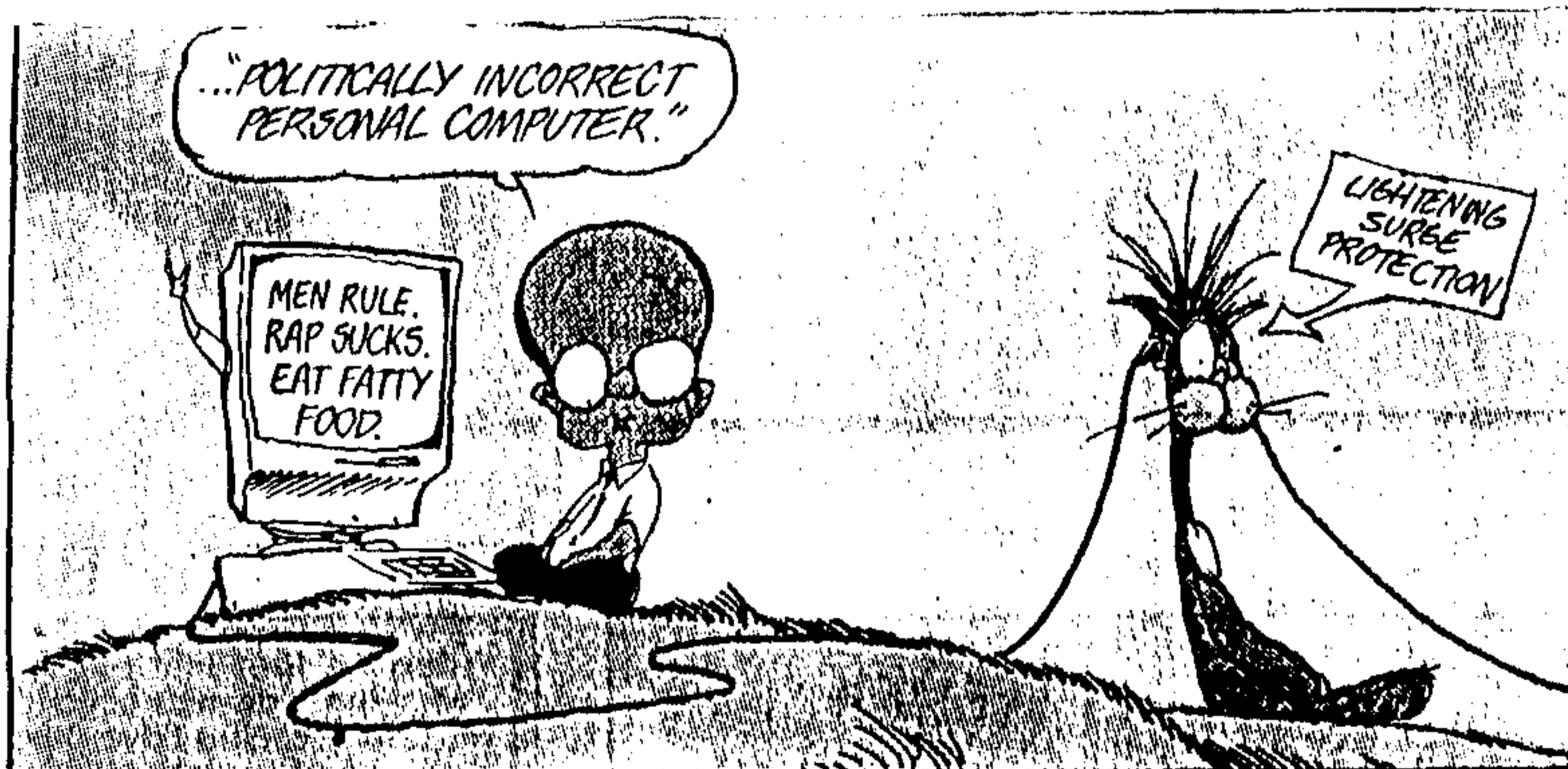
you may leave docs  
for menu any time

From these menus you can see, Class, that these educational programs by the McGuffey of the 1990's are not just for children. The things you can learn about programming and data structuring from the INVENTIONS disk, alone, is worth the price of admission; which, being Fairware in the TI Marketplace is always the best buy in the computer world. So, if you are like Mr. Shakespeare or Mr. Bell over there who are always looking for educational materials for youngsters or even like Ms. Bronte who always wants to get some adult learning materials, you would all be wise to order these disks right away from Mr. Shorock. They are not available in the campus bookstore. Send what you think is a fair amount for each of these disks (\$5 to \$10 per disk would certainly be fair, particularly when you know what is charged for commercialware elsewhere) and help yourself or your young learners in ways that the original McGuffey never dreamed of.

There will be items from each disk on the final.

Your homework is to look through all your disks and cassettes and find the 10 most educational items you can find; ones you personally feel are the most educational, that does the learning task successfully. Bring them to class next time and be prepared to give a 5-minute talk on why you chose these 10.

If you belong to a user group, have everyone in the group do the same and put together some master educational disks and cassettes. This could be a great service to all the newtimers coming into our TI World Community.



No. 69

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 almost 600 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 #68 I published my solution to Dr. Ecker's challenge to alternately assign X the value of A and B without using IF...THEN or any outside help. Computer Monthly has arrived again and his solution is better than mine. Try it with any two numbers -

```
100 A=2.765 :: B=-10
110 X=A+B-X :: PRINT X :: GO
TO 110
```

There has been controversy for years as to whether the TI's pseudorandom number generator is truly random. Dr. Ecker's "Computer Fun & Learning" column in Computer Monthly had a question - if you randomly generate numbers between 0 and 9, how often will you get the same number twice in succession? Three times in succession?

and etc. Since there are 10 numbers to choose from, it seems to me you would get 2 in a row 10% of the time, 3 in a row 1% of the time, 4 in a row .1%...etc. I wrote this to prove it -

```
100 RANDOMIZE
110 C=C+1 :: X=INT(RND*10)::
PRINT X:: IF X=F THEN FL=F
L+1 :: CL(FL)=CL(FL)+1 :: PR
INT "":FL;"":CL(FL):"C:"C:
"X:"X:CL(FL)/C :: GOTO 110 EL
SE FL=0 :: F=X :: GOTO 110
```

After 10,000 tries, I had 2 in a row 8.75% of the time and 3 in a row .83% and 4 in a row .07%. Does that prove anything? I don't know.

(Dr. Ecker points out that those percentages could not ever quite add up to 100%.)

Here is another of my XBasic programs to write assembly source code -

```
100 DISPLAY AT(2,1)ERASE ALL
:"ASSEMBLY HELP SCREEN WRITE
R:"": "This program will wr
ite the": "source code for an
assembly": "routine which ca
n be linked"
110 DISPLAY AT(7,1): "from Ex
tended Basic to dis-": "play
any one of several help": "sc
reens at any designated": "ke
y press or input at any": "po
int in a program."
120 DISPLAY AT(12,1): "The o
riginal source code,": "autho
r unknown, was improved": "by
Karl Roastedt and further":
"modified by Bruce Harrison."
130 DISPLAY AT(20,1): "How ma
ny help screens?" :: ACCEPT
AT(20,24)SIZE(1)VALIDATE(DIG
IT)BEEP: N
140 FOR J=1 TO N :: H%=H%&"H
ELP"&STR$(J)&"," :: NEXT J :
: H%=" DEF "&SEG$(H%,
1,LEN(H%)-1)
150 DATA VMBW EQU >2024,V
MBR EQU >202C,KSCAN EQU
>201C,STATUS EQU >B37C
160 OPEN #1:"DSK1.HELP/8",OU
TPUT :: PRINT #1:H% :: FOR J
=1 TO 4 :: READ M% :: PRINT
```

```
#1:M% :: NEXT J
170 FOR J=1 TO N :: H%="HELP
"&STR$(J):: PRINT #1:H%&" L
WPI WS": LI R13,HEL
PS"&STR$(J)
180 IF J<N THEN PRINT #1:"
JMP SAVSCR"
190 NEXT J :: H%=RPT$(" ",7)
200 PRINT #1:"SAVSCR CLR RO
":H%&"LI R1,SAVIT":H%&"LI
R2,768":H%&"BLWP @VMBR":H%
&"LI R9,NEWSCR":H%&"MOV R
9,R1":H%&"MOV R2,R4"
210 PRINT #1:H%&"LI R3,>60
00": "ADDOFF MOV B #R13+,#R9":
H%&"AB R3,#R9+":H%&"DEC R
4":H%&"JNE ADDOFF":H%&"BLWP
@VMBW"
220 PRINT #1:"KEYLOO BLWP @K
SCAN":H%&"BLWP @KSCAN":H%&"C
B @ANYKEY,@STATUS":H%&"JNE
KEYLOO"
230 PRINT #1:"REPL LI R1
,SAVIT":H%&"BLWP @VMBW": "RET
N LWPI >B3E0":H%&"B @>6
A"
240 PRINT #1:"WS BSS 32
": "SAVIT BSS 768": "NEWSCR
BSS 768": "ANYKEY BYTE >20":
H%&"EVEN"
250 DISPLAY AT(3,1)ERASE ALL
: "Enter data just as you":
want it to appear, in 24": "l
ines. Press Enter for blank"
: "lines."
260 FOR J=1 TO N :: DISPLAY
AT(12,1): "Ready for screen #
"&STR$(J): "": "Press any key"
270 CALL KEY(0,K,S):: IF S=0
THEN 270 ELSE CALL CLEAR
280 ACCEPT AT(1,0):M% :: PRI
NT #1:"HELPS"&STR$(J)&" TEXT
' "&M%&RPT$(" ",30-LEN(M%))
&" "
290 FOR K=2 TO 24 :: ACCEPT
AT(K,0):M% :: PRINT #1:H%&"T
EXT ' "&M%&RPT$(" ",30-LEN(M
%))&" "
300 NEXT K :: NEXT J :: PRIN
T #1:H%&"END"
310 DISPLAY AT(3,1)ERASE ALL
: "Source code has been writ
-": "ten to DSK1 as HELP/8. T
o": "assemble, insert Editor/
": "Assembler module."
320 DISPLAY AT(7,1): "Insert
Assembler disk in drive 1
.": "Select 2 ASSEMBLER": "Loa
d Assembler? Y": "Source file
name DSK2.HELP/8"
```

```
330 DISPLAY AT(12,1): "Object
file name? DSK2.HELP/0": "Li
st file name? Press Enter":
Options? R"
340 DISPLAY AT(15,1): "Load t
he resulting object": "file i
nto your program by": "CALL L
INK": "CALL LOAD("DSK1.HE
LP/0") or,"
350 DISPLAY AT(19,1): "much b
etter, label it with": "ALSAV
E or SYSTEX."
360 DISPLAY AT(21,1): "Access
the screens in your progra
m by": "CALL LINK("HELP1")
": "CALL LINK("HELP2"), etc
."
370 CALL KEY(0,K,S):: IF S=0
THEN 370 ELSE CALL CLEAR
```

For instance, at any point in a program where keyboard input is required and user may not know what to do - ACCEPT AT(24,1):M% :: IF M%="HELP" THEN CALL LINK("HELP1") and the first help screen will pop up to give instructions. Press any key and the previous screen reappears.

This time I am borrowing heavily from the TIMES news letter of England, which has also borrowed from the REC newsletter.

This one is useless, but is a remarkable example of compact complex programming. It shows that there is an algorithm for everything. See if you can figure out how it works -

```
100 CALL CLEAR :: FOR A=1 TO
2 :: FOR B=1 TO 4 :: X=2-AB
S(SGN(B-3)):: FOR C=1 TO X :
: PRINT CHR$(84-7*A+5*B-8*X)
::: NEXT C :: NEXT B :: PRIN
T CHR$(A+31):: NEXT A
```

Another useless one that is easier to figure out -

```
100 DISPLAY AT(1,1)ERASE ALL
: "NUMBER OF MONTH(1-12)"
110 ACCEPT AT(2,12)SIZE(2)VA
LIDATE(DIGIT):A :: IF A<1 OR
A>12 THEN 110
120 DISPLAY AT(3,1):A;"x 4="
```

NEXT PAGE

```

;H34 :: A=H34
130 DISPLAY AT(4,1):A;"*13="
;A+13 :: A=A+13
140 DISPLAY AT(5,1):A;"x 25="
;A*25 :: A=A*25
150 DISPLAY AT(6,1):A;"-200="
;A-200 :: A=A-200
160 DISPLAY AT(8,1):"Input d
ate (1-31):" :: ACCEPT AT(8,
19)SIZE(2)VALIDATE(DIGIT):B
:: IF B<1 OR B>31 THEN 160
170 DISPLAY AT(10,1):A;"+";B
;"=";A+B :: A=A+B
180 DISPLAY AT(11,1):A;"x 2="
;A*2 :: A=A*2
190 DISPLAY AT(12,1):A;"-40="
;A-40 :: A=A-40
200 DISPLAY AT(13,1):A;"x 50="
;A*50 :: A=A*50
210 DISPLAY AT(15,1):"Input
last two digits of year e
g 91:"
220 ACCEPT AT(16,16)SIZE(2)V
ALIDATE(DIGIT):B
230 DISPLAY AT(18,1):A;"+";B
;"=";A+B :: A=A+B
240 DISPLAY AT(19,1):A;"-105
00=";A-10500 :: A=A-10500
250 DISPLAY AT(24,1):"ANY KE
Y FOR ANOTHER"
260 CALL KEY(5,A,B)
270 IF B<1 THEN 260
280 RUN
290 END

```

One for the little ones - change the string to anything you want.

```

1 REM SILLY PROG BY S SHAW
MARCH 1991
2 ! did you see COMPUTER WAR
S-the film? It is said that
the star, who was required t
o type fast into a computer
3 ! could not type, so a pro
gram just like this one was
used to give a good effect!
4 ! now adjust it how you wi
sh and show your friends how
fast you can type
5 ! at end of text string pr
ogram will just stop with th
is listing but can be modifi
ed to do anything you wish!
6 !
100 A$="This is how a non-ty
pist can produce information
on screen quickly,witho

```

```

ut "
110 A$=A$&"having to look at
what keys are being bashed!
Just bash keys and watch ho
w perfect text appears no m
atter what you press."
120 CALL CLEAR :: PRINT A$:
: : : :
130 CALL KEY(5,A,B):: IF B<1
THEN 130
140 C=C+1 :: PRINT SEG$(A$,C
,1):: IF C=LEN(A$)THEN 160
150 GOTO 130
160 GOTO 160

```

And a very fast routine to find prime numbers -

```

100 ! FIRST 100 PRIMES
-QUICKLY-
110 ! Dr H B Phillips
from THE REC NEWSLETTER
March 1988 Vol 3 #2
120 DIM P(300),X(12)
130 A=0 :: B=1 :: D=0.5 :: E
=180
140 M=100 :: L=3 :: F=0
150 ! increase M for more- a
lso increase DIMs.
160 PRINT 2;:: C=B :: IF M=B
THEN END
170 L=INT((M/C)*L+F):: N=L+L
+B
180 FOR I=B TO INT((SQR(N)-B
)8D):: PP=P(I)
190 IF PP=B THEN 230
200 IF PP=A THEN PP=I+I+B ::
PRINT PP;:: P(I)=PP :: C=C+
B :: IF C=M THEN END
210 IF X(I)=A THEN X(I)=(PP*
PP-B)8D
220 FOR J=X(I)TO L STEP PP :
: P(J)=B :: NEXT J :: X(I)=J
230 NEXT I :: IF F=0 THEN S=
I
240 FOR I=B TO L
250 IF P(I)=A THEN PP=I+I+B
:: PRINT PP;:: P(I)=PP :: C=
C+B :: IF C=M THEN END
260 NEXT I :: F=(M-C)*L/E ::
S=L+B
270 GOTO 170

```

And a demonstration of how the INTERRUPT routine works independently of whatever else the computer is doing -

```

100 REM interrupt demo

```

```

110 REM
120 REM MACHINE LANGUAGE
130 REM ROUTINE LOADED AT
140 REM >2600 XB DR E/A WITH
32K
150 REM >7200 MINI MEM NO 32
K
160 REM
170 CALL INIT
180 XM=972B
190 MM=29184
200 LAD=XM
210 REM TEST XB OR MM?
220 CALL LOAD(XM,170)
230 CALL PEEK(XM,X)
240 IF X=170 THEN 270
250 REM NO 32K MUST BE MM
260 LAD=MM
270 A=LAD
280 REM LOAD M/C
290 CALL CLEAR
300 FOR D=540 TO 630 STEP 10
310 CHECK=0
320 FOR N=1 TO 10
330 READ X
340 CALL LOAD(A,X)
350 CHECK=CHECK+X
360 A=A+1
370 NEXT N
380 READ X
390 IF CHECK(<>)X THEN 490
400 NEXT D
410 REM POKE INTERRUPT
420 REM ROUTINE ADDRESS
430 REM INTO >B3C4
440 CALL LOAD(-31804,LAD/256
)
450 REM JUST IDLE AWAY TIME
460 FOR N=1 TO 9940
470 NEXT N
480 STOP
490 PRINT "ERROR IN DATA STA
TEMENT ";D
500 STOP
510 REM EACH DATA STATEMENT
520 REM HAS 10 DATA BYTES
530 REM PLUS A CHECK SUM
540 DATA 192,236,000,092,004
,194,005,131,002,131,987
550 DATA 000,060,026,003,004
,195,006,236,000,094,624
560 DATA 203,003,000,092,060
,172,000,090,006,002,628
570 DATA 017,015,019,010,006
,002,019,004,002,000,94
580 DATA 002,039,010,083,016
,002,002,000,002,086,242
590 DATA 096,003,016,007,002
,000,000,119,010,083,336

```

```

600 DATA 016,002,002,000,000
,072,160,003,002,096,353
610 DATA 064,000,006,192,215
,192,006,192,215,192,1274
620 DATA 016,000,216,044,000
,094,140,000,004,091,605
630 DATA 000,015,000,000,138
,128,000,000,000,000,281
640 END

```

Run that, then press FCTN 4. Enter LIST. Enter NEW. To stop it, enter BYE.

This is an oldie, but well worth repeating. You can use it to turn your cassette recorder on and off, to add speech or music from tape to a running program. With the proper hardware, you could write a program to control almost anything from the cassette port. If it doesn't work, reverse the polarity of the remote. Ed Hall wrote this -

```

100 CALL INIT
110 CALL LOAD(16368,79,70,70
,32,32,32,36,252)
120 CALL LOAD(16376,79,78,32
,32,32,32,36,244)
130 CALL LOAD(18194,37,4,63,2
40)
140 CALL LOAD(9460,2,12,0,45
,29,0,4,91,2,12,0,45,30,0,4
,91,203,78)
150 PRINT "PRESS:" P Play":
"S Stop"
160 CALL KEY(3,A,B)
170 IF B<1 THEN 160
180 ON PDS("PS",CHR$(A),1)+1
GOTO 160,190,200)
190 CALL LINK("ON"):: GOTO 1
60
200 CALL LINK("OFF"):: GOTO
160

```

And that is just about -

MEMORY FULL!

Jim Peterson

SPRINT (Part 2)  
TURBO SPEECH

(or How to Speed up the spoken word)  
by Stephen Shaw  
(Excerpted from the TI99/4A Exchange  
TIMES of Great Britain, issue #6.84  
via HOCUS 99 newsletter Dec 1991)

Now on to something really juicy.  
SPEECH. Old hat huh? Well, this  
information will give you speech in TI  
Basic with the Mini Memory, or if you  
have XBASIC with 32K RAM, will give you  
speech just a bit faster than using  
CALL SAY which slows programs down to  
no end.

For this information I am indebted to  
Neil Lawson who has been delving.  
Speech requires either:

XBASIC with 32K memory or Mini-Memory  
and the Speech Synthesizer.

Program framework (For timing  
purposes):

```
20 CALL INIT
30 S=27648
100 FOR I=1 TO 1000 :: NEXT I
110 PRINT "START....."
120 FOR X=1 TO 20
130 REM TEST ROUTINE HERE
140 FOR T=1 TO 30
150 PRINT ">";
160 NEXT T
170 NEXT X
180 PRINT "END....."
```

This standard routine sets up a  
framework to test our new routine in,  
and gives a basic time reference.

(NB: Times quoted are for my system;  
yours may be different, but the ratios  
should be similar.)

Running the above program, with the  
loops in line 140 running 30 times as  
shown, takes 18.7 seconds from "START"  
to "END". Change line 140 to loop just  
20 times and the timing is 12.7  
seconds.

Now we can insert our two  
possibilities:

The first is the standard CALL SAY, in BASIC:  
130 CALL SAY("THAT IS INCORRECT")

Run this program again: If line 140 is  
looped 20 times, the time is 44  
seconds. If line 140 is looped 30  
times, the time is 50 seconds.

The time for the speech is constant, it  
adds about 21 seconds to the program.

Now for something different, (also  
works with Mini-Memory):  
130 CALL  
LOAD(S,70,"",S,65,"",S,72,"",S,70,"",S,64  
,"",S,80)

If, you now run the program, it says  
the same thing as many times, but look  
at the timings:

If line 140 loops 20 times: 26.3  
seconds.  
If line 140 loops 30 times: 26.5  
seconds.

We know that looping line 140 an extra  
10 times adds 6 seconds...so where have  
those 6 seconds gone

The CALL SAY routine holds everything  
up until it has finished speaking. But  
using the CALL LOAD equivalent, while  
the computer is speaking, gets on  
with the next chore too. The "dead  
time" is used and soaks up these 6  
seconds.

Thus using the CALL LOAD equivalent,  
the computer speaks faster, and also  
permits your program to run more  
quickly if there is work for it to do  
between speech outputs.

That's the clever demonstration!  
(Impressed?) Now for the theory.

References: Editor/Assembler Manual,  
pages 351, 355, 422-427. Reference in  
para 1, page 355, should be the  
Section 22.1.4 not as printed in the  
manual.

Address -27648 is the SPEECH WRITE  
address. We keep feeding it with bytes,  
and in due course the computer speaks.  
The bytes to feed to that address are

6696 68FE 77E9 56B3 70DB 17A5 1913 1D82 1DA2 28B6 3571

First, decide what you want to say from  
the standard vocabulary. Then look in  
the table (pp. 422-427) for the address  
of that word or phrase. "THAT IS  
CORRECT" is given as 6816. That is  
Hexadecimal not a Decimal number. The  
four numbers are reversed, and become  
6168.

Now we offset them by Hex 40 and feed  
them in. As we are dealing with  
decimals with our CALL LOAD, that means  
we add decimal 64 to each digit in  
turn:

(6+64) (1+64) (6+64)  
70 65 72 70

If the numbers were Hex A-F these have  
a decimal value as follows:

A=10 B=11 C=12 D=13 E=14 F=15

Now we must indicate end of word by  
loading a zero, again offset, thus  
0+64=64. Finally, instruct the computer  
to speak by loading Hex 50, Decimal 80.

Thus we have loaded, in order:

Check back to the listing. Note the way  
CALL LOAD has been used: a simple  
command to load the same address with  
several different values.

To assist your experimentation, here  
are some Hex addresses from the manual.  
Remember to (reverse) them, translate  
to decimal and offset.

TEXAS INSTRUMENTS...6696 THAT IS  
RIGHT....68FE  
WHAT WAS THAT.....77E9 READY TO  
START....56B3  
YOU WIN.....70DB  
AGAIN....17A5..  
ANSWER.....1913 CHECK...1D82  
CHOICE.....1DA2 COMMAND.1F1A  
ELSE.....28B6 GOODBYE.3148  
HELP.....3571 HURRY...3757

END



NEXT MEETING TUESDAY, SEPTEMBER 8, 1992 HAPPY BIRTHDAY MUNCH WE'RE 10!!

MUNCH OFFICERS AND NUMBERS (all in 508 area unless noted)

|                 |               |              |                    |         |
|-----------------|---------------|--------------|--------------------|---------|
| President       | W.C. Wyman    | 865/9683     |                    |         |
| Vice President  | Bruce Willard | 852/3250     | MUNCH DUES         |         |
| Secretary       | Jim Cox       |              |                    |         |
| Treasurer       | Jim Cox       | 869-2704     | NEW MEMBERSHIP     | \$25.00 |
| Acting Editor   | Jim Cox       |              | RENEWAL MEMBERSHIP | \$15.00 |
| Adv.Prog. Chair | Dan Rogers    | 248-5502     | NEWSLETTER ONLY    |         |
| Library         | OPEN          |              | SUBSCRIPTION       | \$13.00 |
| Disk Librarian  | Lou Holmes    | 617 965/3584 |                    |         |
| Tape Librarian  | Walter Nowak  | 413 436/7675 |                    |         |
| NEW-AGE/99      | Jack Sughrue  | 476/7630     |                    |         |

AUGUST MEETING. There were 13 members at the August meeting. We cut the meeting short when a severe thunder storm hit, we didn't want to fry our system. Lou was able to help answer some questions about Funnelweb. Stan Mozner won the raffle.

SEPTEMBER MEETING. This month we hope to have demo's of Astroblitz and Magog, two interesting games. We also have a special raffle prize, see info below.

RAFFLE. Every month we have a raffle to help defer the rental cost of our meeting hall. A typical raffle will have game and utility programs T-Shirts, books, bumper stickers, blank discs and all sorts of odds and ends for the T.I. Special this month we have a Glare Guard Professional monitor screen.

LIBRARY NOTICE. Please return any items borrowed from our library. If you can not come to a meeting or give these items to someone who will be at the meeting.

REPRINTS. Reprints are permitted as long as credit is given to M.U.N.C.H.

ARTICLES. I am always looking for articles for this newsletter, anything which interests you will probably interest other members of the TI community, so please share your ideas and opinions with all of us.

DISK LIBRARY. The disk library will be at the meetings from now on. We have copies of all disks in the library and they are available to members for just \$1.50 each for single discs, \$2.00 floppies, \$3.00 double discs and \$4.00 double floppy.

FOR SALE. The group has a TI Count Business Software package available for sale. If interested contact Jim Cox at the above number or the club address. We also have blank disks for sale. The price is \$6.00 for a package of 25 disks.

DISK OF THE MONTH. This month's disk is #110, is Magog.

THE MUNCH VIDEO is ready, members can purchase it for \$5.00, plus \$3.00 postage for mail orders.

