

THE BLOODBANK

Walter H. Blood
2032 North 32nd Street
Kansas City, Kansas 66104
APRIL 1989

[continued from last month]

```

1180 !
1190 ! PART 4
1200 !
1210 GOSUB 2180 :: DISPLAY AT(4,1):"W
e present you with six":"punning ridd
les and answers, omitting one word in
each"
1220 DISPLAY AT(7,1):"answer. 4 poin
ts each":D$:"Example:":TAB(4);"What r
elation is the":"doormat to the doors
tep?"
1230 DISPLAY AT(12,8):"A stepFARTHER"
:: DISPLAY AT(21,1):"<ENTER> Go to
next riddle" :: GOSUB 2160 :: L=19 ::
C=3 :: Q=21
1240 IF T$="Y" THEN GOSUB 2070 :: IF
T>44 THEN 1290
1250 DISPLAY AT(14,1):Q$(Q):: DISPLAY
AT(17,1):R$(Q-20):: DISPLAY AT(19,1)
SIZE(18):A$(Q)
1260 GOSUB 2190 :: ON KY-11 GOTO 1270
,1280,1290,1300,1310
1270 P=PU :: RETURN
1280 Q=Q+1+6*(Q=26):: GOSUB 2220 :: G
OTO 1240
1290 P=6 :: RETURN
1300 P=PD :: RETURN
1310 GOSUB 2140 :: ACCEPT AT(L,C-2)SI
ZE(-18)VALIDATE(UALPHA):A$(Q):: GOSUB
2160 :: GOTO 1280
1320 !
1330 ! SCORE
1340 !
1350 GOSUB 2250 :: DISPLAY AT(4,9):"*
RESULTS *":TAB(9);"*****" :: D
ISPLAY AT(9,7):"Working ..."
1360 RESTORE 1370 :: FOR I=1 TO 26 ::
READ B$ :: FOR J=1 TO LEN(B$):: X=AS
C(SEG$(B$,J,1)):: C$(I)=C$(I)&CHR$(X+
1):: NEXT J :: NEXT I
1370 DATA KTMBGDNM,RXB@LNQD,LHRMNLQ,
TLAQDKK@,@LD$G$XRS,BGGHRSDM,G@MCHB@,H
MMTDMCN,UHMDX@QC,MTHR@MBD
1380 DATA BTBTladq,LHLHbqx,MNMNadchdm
s,WDWD`c,UHUhEx,UNONudq,HBHBkd,LDLDms
n,DXDRHFGS,B@AA@FD
1390 DATA JMD@CR,F@KKNM,RDKEHRG,@C@L,
CDV,BNNR
1400 FOR I=1 TO 10 :: TP=TP-2*(A$(I)=
C$(I)):: NEXT I
1410 FOR I=11 TO 18 :: TP=TP-4*(A$(I)

```

```

=C$(I)):: NEXT I
1420 FOR I=21 TO 26 :: TP=TP-4*(A$(I)
=C$(I)):: NEXT I
1430 TP=TP-10*(A$(19)=C$(19))-14*(A$(
20)=C$(20)):: DISPLAY AT(7,3):"YOUR S
CORE = ":TP
1440 IF TP>91 THEN B$="GENIUS" ELSE I
F TP>77 THEN B$="EXCEPTIONAL" ELSE IF
TP>64 THEN B$="SUPERIOR" ELSE IF TP>
54 THEN B$="GOOD" ELSE B$="NO RATING"
1450 DISPLAY AT(9,2):"YOUR RATING = "
;B$;:"To review game (1-5, 0=Quit)":
;:" Enter your choice: 0":RPT$("_",2
8)
1460 ACCEPT AT(13,22)SIZE(-1)VALIDATE
("012345")BEEP:B$ :: ON POS("012345",
B$,1)GOTO 1620,1470,1490,1510,1530,15
50
1470 FOR I=1 TO 10 :: DISPLAY AT(17,2
):Q$(I);TAB(20);A$(I);;"Correct answe
r is ":C$(I):: GOSUB 1600
1480 GOSUB 2230 :: GOSUB 1590 :: NEXT
I :: GOTO 1460
1490 FOR I=11 TO 18 :: DISPLAY AT(17,
1):A$(I);TAB(13);Q$(I);;" = co
rrect answer" :: GOSUB 1600
1500 GOSUB 2230 :: GOSUB 1590 :: NEXT
I :: GOTO 1460
1510 I=19 :: DISPLAY AT(17,3):A$(I);;
:TAB(3);C$(I);;" = correct answer" ::
GOSUB 1600
1520 GOSUB 2230 :: GOSUB 1590 :: GOTO
1460
1530 FOR I=21 TO 26 :: DISPLAY AT(15,
1):Q$(I):: DISPLAY AT(17,1):R$(I-20):
: DISPLAY AT(19,1)SIZE(18):A$(I)
1540 DISPLAY AT(20,1):C$(I);;" = corre
ct answer" :: GOSUB 1600 :: GOSUB 223
0 :: GOSUB 1590 :: NEXT I :: GOTO 146
0
1550 I=20 :: DISPLAY AT(17,3):A$(I);;
:TAB(3);C$(I);;" = correct answer" ::
GOSUB 1600
1560 GOSUB 2230 :: GOSUB 1590 :: GOTO
1460
1570 DISPLAY AT(22,6):"You GOT this o
ne!" :: RETURN
1580 DISPLAY AT(22,5):"You MISSED thi
s one!" :: RETURN
1590 CALL HCHAR(15,1,32,256):: RETURN
1600 IF A$(I)=C$(I)THEN GOSUB 1570 EL
SE GOSUB 1580
1610 RETURN
1620 P=7 :: RETURN
1630 !
1640 ! PART 5
1650 !
1660 GOSUB 2180 :: DISPLAY AT(4,1):"C
an you think of a seven- letter wor
d whose letters total just 21? 14

```

```

points":D$
1670 FOR I=64 TO 69 :: FOR J=1 TO 28
STEP 7 :: DISPLAY AT(I-56,J):CHR$(I+J
+(J=22))
1680 DISPLAY AT(I-56,J+2):USING "###":
I+J+(J=22)-64 :: NEXT J :: NEXT I
1690 DISPLAY AT(14,1):"G 7 N 14":;
:::" YOUR ANSWER ";A$(20):: GOSUB 21
60 :: L=17 :: C=17
1700 IF T$="Y" THEN GOSUB 2070 :: IF
T>44 THEN 1730
1710 GOSUB 2190 :: ON KY-11 GOTO 1720
,1700,1730,1740,1750
1720 P=PU :: RETURN
1730 P=6 :: RETURN
1740 P=PD :: RETURN
1750 GOSUB 2140 :: GOSUB 2170 :: ACCE
PT AT(L,C-2)SIZE(-7)VALIDATE(WALPHA):
A$(20):: GOTO 1720
1760 !
1770 ! CLOCK
1780 !
1790 T=24 :: R=10240 :: CALL INIT ::
HX$="0123456789ABCDEF" :: KH,HR,MN,SC
=0
1800 DISPLAY AT(20,1):"LOADING TIMER"
:: RESTORE 1940
1810 READ B$ :: IF B$="CHEK" THEN DIS
PLAY AT(20,17):"CHECK";KH :: GOTO 181
0
1820 IF B$="END" THEN 1890
1830 IF LEN(B$)<>4 THEN DISPLAY AT(20
,17):"ERROR";B$ :: GOTO 1810
1840 HI=(POS(HX$,SEG$(B$,1,1),1)-1)*1
6+POS(HX$,SEG$(B$,2,1),1)-1
1850 LO=(POS(HX$,SEG$(B$,3,1),1)-1)*1
6+POS(HX$,SEG$(B$,4,1),1)-1
1860 CALL LOAD(R,HI,LO):: R=R+2 :: KH
=KH+HI :: IF KH>256 THEN KH=KH-256
1870 KH=KH+LO :: IF KH>256 THEN KH=KH
-256
1880 GOTO 1810
1890 CALL PEEK(8196,H,L):: LFAL=H*256
+L :: NEWL=LFAL-16 :: NH=INT(NEWL/256
):: NL=NEWL-NH*256
1900 CALL LOAD(8196,NH,NL):: CALL LOA
D(NEWL,83,84,65,82,84,32,40,0)
1910 CALL LOAD(NEWL+8,83,84,79,80,32,
32,40,40):: CALL LINK("START")
1920 CALL LOAD(10543,T,0,HR,0,MN,0,SC
)
1930 RETURN
1940 DATA C820,28EA,292C,C820,28E8,29
2B,C820,28E6,292E,04E0,2930,04E0
1950 DATA 2932,04E0,2934,0200,282E,C8
00,83C4,045B,04E0,83C4,045B,CHEK
1960 DATA 0300,0000,02E0,2928,0602,16
52,C0A0,28EA,0586,0286,003C,160E,04C6
1970 DATA 0585,0285,003C,1609,04C5,05
84,80C4,1605,04C4,0283,001B,1301,0584

```

```

,CHEK
1980 DATA 06D0,DB00,8C02,06C0,E020,28
E4,DB00,8C02,4020,28E4,D064,28EC,0941
1990 DATA 0221,9000,DB01,8C00,0A41,02
41,0F00,0221,9000,DB01,8C00,0201,9A00
2000 DATA DB01,8C00,D065,28EC,0941,02
21,9000,DB01,8C00,0A41,0241
2010 DATA 0F00,0221,9000,DB01,8C00,02
01,9A00,DB01,8C00,D066,28EC,0941
2020 DATA 0221,9000,DB01,8C00,0A41,02
41,0F00,0221,9000,DB01,8C00,0720,83D6
,CHEK
2030 DATA 02E0,83E0,045B,4000,000D,00
17,003B,0001,0203,0405,0607,0809,1011
,1213
2040 DATA 1415,1617,1819,2021,2223,24
25,2627,2829,3031,3233
2050 DATA 3435,3637,3839,4041,4243,44
45,4647,4849,5051,5253,5455,5657,5859
,CHEK
2060 DATA END
2070 CALL PEEK(10543,X,X,HR,X,MN):: T
=60*HR+MN :: IF T<45 THEN 2090
2080 GOSUB 2130 :: DISPLAY AT(10,9):"
TIME IS UP!" :: FOR I=1 TO 1000 :: NE
XT I
2090 RETURN
2100 !
2110 ! COMMON SUBROUTINES
2120 !
2130 CALL VCHAR(1,3,32,672):: RETURN
2140 FOR X=24 TO 22 STEP -1 :: CALL H
CHAR(X,3,32,28):: NEXT X :: RETURN
2150 DISPLAY AT(21,1):"<ENTER> Mov
e to next line" :: RETURN
2160 DISPLAY AT(22,1):"<FCTN D> Inp
ut answer":"<FCTN 9/6> Prev/Next part
":"<FCTN 5> Quit & see score" :: RE
TURN
2170 DISPLAY AT(20,1):"<ENTER> Recor
d answer and":TAB(13);"go to next par
t" :: RETURN
2180 GOSUB 2130 :: DISPLAY AT(2,5):P$
(F):TAB(5);RPT$("-",20):: RETURN
2190 CALL HCHAR(L+5*(P=4),2,32):: CAL
L HCHAR(L+5*(P=4),2,30)
2200 CALL KEY(3,KY,ST):: IF KY<9 OR(K
Y>9 AND KY<12)OR KY>15 THEN 2190 ELSE
CALL HCHAR(L+5*(P=4),2,32):: KY=KY-7
*(KY=9)
2210 RETURN
2220 FOR X=3 TO 30 :: CALL VCHAR(14,X
,32,6):: NEXT X :: RETURN
2230 DISPLAY AT(24,1):"Press any key
to continue"
2240 CALL KEY(3,KY,ST):: IF ST=0 THEN
2240 ELSE RETURN
2250 DISPLAY AT(1,7)ERASE ALL:"*****
*****":TAB(7);"* BRAIN GAMES *"
:TAB(7);"*****" :: RETURN

```

TI-99/4A SOUND CHIP ACCURACY CHART

NOTE	CALL SOUND frequency as reported on page III-7 of Users Ref. Guide	Actual measured CALL SOUNDS frequencies that generates this note tune.	COMMENTS
A	110	N/A	Volume too soft to meter.
A#	117	N/A	" " " " "
B	123	N/A	" " " " "
C (low C)	131	131	
C#	139	138	Off by 1 HZ, also 5% flat.
D	147	147	
D#	156	156	5% sharp
E	165	165	
F	175	175	5% sharp
F#	185	185	
G	196	196	
G#	208	207	Ref. Guide high by 1 HZ
A	220	220	5% sharp
A#	233	233	
B	247	247	
C (middle C)	262	261	Ref. Guide high by 1 HZ
C#	277	277	
D	294	293	Ref. Guide high by 1 HZ
D#	311	311	
E	330	329-330	
F	349	349-350	
F#	370	369	Ref. Guide high by 1 HZ
G	392	392-393	
G#	415	415-416	
A	440	440-441	
A#	466	466	
B	494	491-493	Ref. Guide high by 1 HZ
C (high C)	523	522-523	
C#	554	553-555	
D	587	584-587	
D#	622	619-622	
E	659	656-659	
F	698	697-701	
F#	740	738-743	
G	784	780-784	
G#	831	826-831	
A	880	878-884	
A#	932	929-935	
B	988	986-994	
C	1047	1041-1050	
C#	1109	1102-1112	
D	1175	1172-1183	5% sharp
D#	1245	1236-1249	
E	1319	1309-1323	
F	1397	1389-1406	
F#	1480	1462-1481	10% flat
G	1568	1565-1586	5% sharp
G#	1661	1657-1681	10% sharp
A	1760	1743-1761	15% flat

Reprinted from November 1988 Cin-Day newsletter

TIPS FROM THE TIGERCUB

#44

Copyright 1987

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 120 original programs in Basic and Extended Basic, available on cassette or disk, NOW REDUCED TO just \$1.00 EACH!, plus \$1.50 per order for cassette or disk and PP&M. Minimum order of \$10.00. Cassette programs will not be available after my present stock of blanks is exhausted. The Handy Dandy series, and Color Programming Tutor, are no longer available on cassette. Descriptive catalogs, while they last, \$1.00 which is deductible from your first order.

Tigercub Full Disk Collections, reduced to \$5 postpaid. Each of these contains either 5 or 6 of my regular 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, BRAIN GAMES, BRAIN TEASERS, BRAIN BUSTERS!, ACTION GAMES, REFLEX AND CONCENTRATION, MANEUVERING GAMES, TWO-PLAYER GAMES, PROGRAMMING TUTOR, PROGRAMMER'S UTILITIES, KID'S GAMES, MORE GAMES WORD GAMES, VOCABULARY AND READING, ELEMENTARY MATH, MUSICAL EDUCATION, MIDDLE/HIGH SCHOOL MATH, KALEIDOSCOPIES AND DISPLAYS

NUTS & BOLTS DISKS

These are full disks of 100 or more utility subprograms in MERGE format, which you can merge into your own programs and use, almost like having another hundred CALLs available in Extended Basic. Each is accompanied by printed documentation giving an example of the use of each.

NUTS & BOLTS No. 1 has 100 subprograms, a tutorial on using them, and 5pp. documentation.

NUTS & BOLTS No. 2 has 108 subprograms and 10 pp. of documentation.

NUTS & BOLTS No. 3 has 140 subprograms and 11pp. of documentation.
NOW JUST \$15 EACH, PPD.

TIPS FROM THE TIGERCUB

These are full disks which contain the programs and routines from the Tips from the Tigercub newsletters, in ready-to-run program format, plus text files of tips and instructions.

TIPS VOL. 1 contains 50 original programs and files from Tips newsletters Nos. 1 thru 14.

TIPS VOL. 2 contains over 60 programs and files from Nos. 15 thru 24.

TIPS VOL. 3 has ANOTHER 62 from Nos. 25 through 32.

TIPS VOL. 4 has 40 more from issues No. 33 through 41.
NOW, JUST \$10 EACH, PPD.

* NOW READY *
TIPS VOL. 5 - Another
49 programs and files
* from issues No. 42 -*
* 50. Also \$10 ppd *

TIGERCUB CARE DISKS
#1, #2, #3 and #4.

Full disks of text files (printer required).

No. 1 contains the Tips newsletters Nos.42 thru #45, etc.

No. 2 and No. 3 have articles mostly on ExBasic programming.

No. 4 contains Tips newsletters Nos. 46-52.

These were prepared for user group newsletter editors but are available to anyone else for \$5 each postpaid.

Thanks to Steve Chapman and Bill Wallbank of Stone & Webster Engineering Corp. TIUG for this one. If V=21 you are in Extended Basic, otherwise you are in Basic. I am not sure it will work with all consoles and modules. -

```
100 RANDOMIZE (0)
110 V=INT(RND*100)
```

How can you input a blank (CHR\$(32)) with ACCEPT AT? As far as I know, you can't. With LINPUT, just hit the space bar, and with INPUT, type " ". But with ACCEPT AT the space bar gives a null string and " " gives " ". However, you can code around it -

```
X$=CHR$(34)&CHR$(32)&CHR$(32)::ACCEPT AT(1,1):
T$::IF T$=X$ THEN T$=CHR$(32)
```

And, to clear up the puzzling behavior of the "quote marks" -

```
100 CALL CHARPAT(34,CH$)
::CALL CHAR(35,CH$)!written by Jim Peterson
110 DISPLAY AT(1,7)ERASE ALL:"THE # PUZZLE:" You can't enter PRINT # or PRINT ### - the computer demands an even number of #."
120 DISPLAY AT(5,1):"1 PRINT ## !prints a null string (nothing):"2 PRINT ## !prints #"
130 DISPLAY AT(8,1):"3 PRINT #### !prints #:"4 PRINT ##### !crashes as STRING-NUMBER MISMATCH"
140 DISPLAY AT(11,1):"5 PRINT ##### !crashes as SYNTAX ERROR"
150 DISPLAY AT(13,1):"6 PRINT ##### !prints ##"
:"7 PRINT ##### !prints ###"
:"8 PRINT ##### !prints ####"
160 DISPLAY AT(16,1):"9 PRINT ##### !prints ##"
:"10 PRINT ##### !crashes as STRING-NUMBER MISMATCH"
170 DISPLAY AT(19,1):"11 PRINT ##### !crashes as SYNTAX ERROR"
:"12 PRINT ##### !prints ##"
```

```
180 DISPLAY AT(22,1):"13 PRINT ##### !prints ##"
:"14 PRINT ##### !prints ###"
:"15 PRINT ##### !prints ####"
190 DISPLAY AT(24,1):"TRY IT! LINE NO.(1-14)?"
:ACCEPT AT(24,25)VALIDATE(DIGIT)SIZE(2)BEEP:LN
::IF LN<1 OR LN>14 THEN
190
200 CALL CLEAR :: ON LN GOSUB 230,240,250,260,280,290,300,310,320,330,340,350,360,370
210 PRINT ::;"Press any key"
220 CALL KEY(0,K,S)::IF S=0 THEN 220 ELSE 110
230 PRINT "" :: RETURN
240 PRINT "*" :: RETURN
250 PRINT "" :: RETURN
260 PRINT ""*"" !crashes as STRING-NUMBER MISMATCH - the * is misinterpreted as a multiplier! Same with +,-,/
270 !with anything else, including numerals, crashes as SYNTAX ERROR - but inserts a space before the character!
280 PRINT ""*"" :: !crashes
290 PRINT """" :: RETURN
300 PRINT ""*"" :: RETURN
310 PRINT ""*"" :: RETURN
320 PRINT """" :: RETURN
330 PRINT ""*"" !crashes
340 PRINT ""*"" !crashes
350 PRINT """" :: RETURN
360 PRINT ""*"" :: RETURN
370 PRINT ""*"" :: RETURN
```

The method of closing an "ajar" file, described in Tips #20, doesn't always work, but this one seems to be reliable -

```
100 ON ERROR 500 :: OPEN #1:"DSK1.TEST" :: INPUT #1:A$ :: PRINT A$ :: STOP
500 ON ERROR 510 :: CLOSE #1
510 INPUT "CHECK DISK AND DRIVE, PRESS ANY KEY": DUMMY$ :: RETURN 100
```

This one is just for the fun of it - it uses the contents of computer memory to create designs -

```
100 DISPLAY AT(3,10)ERASE ALL:"COLORPEEK":TAB(7);"by Jim Peterson"::"Watch the computer's memory"::"displayed in color."
110 DISPLAY AT(12,1):"Choose"::"(1) plain colors"::"(2) bars & checks"::"(3) patterns"
:ACCEPT AT(12,8)VALIDATE("123")SIZE(1):Q::CALL CLEAR::IF Q=1 THEN
170
120 DISPLAY AT(12,5):"wait, please"::IF Q=3 THEN 140
130 FOR CH=32 TO 143::CALL CHAR(CH,RPT$("FO",8))::NEXT CH::GOTO 160
140 RANDOMIZE::FOR CH=32 TO 88::FOR J=1 TO 4::X$=SEG*("0018243C425A667E8199A5BDC3DBE7FF",INT(16*RND+1)*2-1,2)::B$=B$&X$::C$=X$&C$::NEXT J::CALL CHAR(CH,B$&C$)
150 CALL CHAR(CH+55,B$&C$)::B$,C$=""::NEXT CH
160 FOR SET=0 TO 14::CALL COLOR(SET,SET+1,16-SET)::NEXT SET::CALL SCREEN(2)::GOTO 180
170 FOR SET=0 TO 14::CALL COLOR(SET,SET+2,SET+2)::NEXT SET::CALL SCREEN(16)
```

```

180 FOR J=-1 TO -2000 S
TEP -1 :: CALL FEEK(J,A
):: A=A-(A<33)*(A+32)::
A=A+(A>143)*(A/2):: R=
R+1+(R=24)*24 :: CALL H
CHAR(R,1,A,32)
190 C=C+1+(C=32)*32 ::
CALL VCHAR(1,C,A,24)::
NEXT J :: GOTO 100

```

Unlike most of the number games played against the computer, you can win this one -

```

100 CALL CLEAR :: CALL
SCREEN(16):: DISPLAY AT
(3,8):"THE '37' GAME" !
by Jim Peterson
110 DISPLAY AT(5,1):" W
e will take turns picki
ng":"a number from 1 to
5, but":"not the numbe
r that was just":"picke
d."
120 DISPLAY AT(10,1):"
The numbers we pick wil
l be":"added to the tot
al count."
130 DISPLAY AT(13,1):"
Whoever reaches 37 is t
he":"winner, but if you
go over":"37 you lose.
"
140 CALL SHOW(20,1,"Pre
ss any key to start")
150 CALL KEY(0,K,S):: I
F S=0 THEN 150
160 DATA 4,11,17,24,30,
37
170 DATA 262,330,392,52
3,523
180 DATA 1047,784,659,5
23,52 3
190 C,P=0 :: CALL CLEAR
:: CALL MAGNIFY(2):: R
=10 :: FOR J=1 TO 5 ::
CALL SPRITE(#J,48+J,5,R
,10):: R=R+30 :: NEXT J
200 CALL SHUW(24,1,"(Y)
ou or (C)omputer first?
"):: ACCEPT AT(24,28)VA
LIDATE("YC")SIZE(1):Q#
:: DISPLAY AT(24,1):""
210 IF Q#="C" THEN CALL
SHOW(22,8,"I pick 4"):
: CALL COLOR(#4,1):: P=
4 :: C=4 :: CALL SHOW(3
,10,"COUNT=4")

```

```

220 CALL SHOW(20,8,"Pic
k your number"):: ACCEP
T AT(20,26)VALIDATE("12
345"):N :: IF N=P THEN
220
230 IF P>0 THEN CALL CO
LOR(# P,5)
240 CALL COLOR(#N,1)::
P=N :: C=C+N :: CALL SH
OW(3,10,"COUNT= "&STR$(
C)):: IF C=37 THEN 320
ELSE IF C>37 THEN 340
250 RESTORE 160
260 READ X :: IF C<X TH
EN D=X-C ELSE IF X<37 T
HEN 260
270 CALL SHOW(22,8,"I'm
thinking..."):: FOR Y=
1 TO 700 :: NEXT Y
280 IF B>5 AND B/2=INT(
B/2)THEN B=B/2
290 IF B>5 OR B=P THEN
B=1-(P=1)
300 CALL SHOW(22,8,"I
pick "&STR$(B)):: CALL
COLOR(#P,5):: CALL CO
LOR(#B,1):: P=B :: C=C
+B :: CALL SHOW(3,10,"
COUNT= "&STR$(C))
310 IF C=37 THEN 340 E
LSE IF C>37 THEN 320 E
LSE 220
320 RESTORE 170 :: FOR
J=1 TO 5 :: READ F ::
CALL SOUND(100,F,5,F*
1.03,5):: NEXT J :: CA
LL SHOW(12,8,"YOU WIN!
")
330 CALL SHOW(15,8,"Pl
ay again? (Y/N)"):: AC
CEPT AT(15,2 6)VALIDAT
E("YN"):Q# :: IF Q#="N
" THEN STOP ELSE 190
340 RESTORE 180 :: FOR
J=1 TO 5 :: READ F ::
CALL SOUND(300,30000,
30,30000,30,F,30,-4,5)
:: NEXT J :: CALL SHOW
(12,8,"YOU LOSE!"):: G
OTO 330
350 SUB SHOW(R,C,T#)::
FOR J=1 TO 10 :: DISP
LAY AT(R,C):" " :: DIS
PLAY AT(R,C):T# :: NEX
T J :: SUBEND

```

A couple more peculiarities of the computer -

```

100 DISPLAY AT(3,8)ERA
SE ALL:"POS PUZZLE #1"
: : " from Tiger c
ub"
110 DISPLAY AT(9,1):"W
hy does the computer s
ay":"that X=1 if you a
nswer the":"prompt wit
h the Enter key":"(nul
l-string) ?"
120 DISPLAY AT(14,1):"
110 INPUT M#"
130 DISPLAY AT(15,1):"
120 X=POS("TESTING",
M#,1):::"PRINT X :: G
OTO 100"
140 !POS PUZZLE #1 - w
hy does the computer s
ay that X=1 if you ans
wer the prompt with En
ter (null-string) ?
- Jim Peterson
150 INPUT M#
160 X=POS("TESTING",M#
,1):: PRINT X :: GOTO
140

```

And -

```

100 DISPLAY AT(3,8)ERA
SE ALL:"POS PUZZLE #2"
: : " from Tiger c
ub"
110 DISPLAY AT(7,1):"W
hy does the computer s
ay":"that the first po
sition of":"null-strin
g is at whatever":"pos
ition it is told to st
art":"search at?"
120 DISPLAY AT(13,1):"
100 M# -"
130 DISPLAY AT(14,1):"
110 DISPLAY AT(20,1):"
"POS?" :: ACCEPT AT(2
0,6):P"
140 DISPLAY AT(16,1):"
120 X=POS("TESTING",
M#,P):: DISPLAY AT(22,
1):""X="";X :: GOTO 11
0"
150 M#=""
160 DISPLAY AT(21,1):"
POS?" :: ACCEPT AT(21,
6):P
170 X=POS("TESTING",M#
,P):: DISPLAY AT(23,1)
:"X=";X :: GOTO 160

```


Here is an improvement to the PRINTSPEAKER in Tips #40 - in lines 130 and 160, change the CHR\$(1)&"1" to CHR\$(3)&"255". This will avoid problems if the program being converted opens FILE #1.

Irwin Hott informs me that assembly routines which have been imbedded into XBasic programs, using ALSAVE or SYSTEX, can be saved to cassette and reloaded. This could be very useful for those who have a stand-alone or "matchbox" 32k.

And, a mini-game for you to have fun with or improve on -

```

1 ! 2-LINE GAME
   by Jim Peterson
   - use S&D keys to
   paint the white line
   on the highway
2 ! if it is too easy,
   change the 6 in A$=RPT
$(CHR$(143),6) to 5
   and the 5 in C>T+5 to
   4
100 CALL CLEAR :: A$=R
PT$(CHR$(143),6):: CAL
L COLOR(14,2,2,2,16,16
):: CALL SCREEN(4):: T
=11 :: C=14 :: CALL HC
HAR(22,C+2,42):: RANDO
MIZE
110 T=T+INT(3*RND-1)+(
T=21)-(T=1):: PRINT TA
B(T);A$ :: CALL KEY(3,
K,S):: C=C+(K=83)-(K=6
8):: CALL HCHAR(22,C+2
,42):: IF C<T OR C>T+5
THEN STOP ELSE 110

```

And finally, one of the best examples of compact programming I have ever seen -

```

1 !JOHN WITTE'S 3-LINE
VERSION OF JOHN WILLF
ORTH'S WAVE POWER - PU
BLISHED IN GREATER OMA
HA UG NEWSLETTER
100 CALL CLEAR :: A$(1
)="ABCDEFGGFEDCBA" :: F
OR I=1 TO 7 :: CALL CH
AR(72-I,RPT$("0",2*I-2
)&"FFFF",47,"30303EFF7
F3E1E04"):: A$(I+1)=SE
G$(A$(I),2,12)&SEG$(A$
(I),2,1):: NEXT I
110 CALL SPRITE(#5,47,
2,180,180,-23,0,#6,47,
2,80,100,-23,0):: CALL
MAGNIFY(2)
120 FOR I=1 TO 12 :: P
RINT A$(I+(I>7)*2*(I-7
))&A$(1+I+(I>6)*2*(I-6
)):: NEXT I :: GOTO 12
0

```

Memory full
Jim Peterson

COMPUTER DICTIONARY - Author Unknown

Most computer owners are not electronic or computer experts ... though when reading magazines, articles, etc., they use terms, abbreviations, and computer slang known only to the informed. With the following dictionary, even you can understand it!

-
- ✓BO86 - The year you finally pay off your computer.
 - ✓BO87 - The year you finally pay off your peripherals.
 - ✓BO88 - The year your spouse forgives you for buying the computer.
 - ✓Address - Type of attire worn by some female programmers.
 - ✓Algol - The husband of Polygol. Their missing daughter is Polygon.
 - ✓Altair - A place where computers are sacrificed.
 - ✓Array - A blast from a CRT.
 - ✓Backup - Opposite of foreward.
 - ✓Branch - A stick used for beating.
 - ✓Buffer - A programmer who works in the nude.
 - ✓Coding - An addictive drug.
 - ✓Computer - A device designed to speed and automate errors.
 - ✓Cp/m - Program listing for 'Look in the evening section'.
 - ✓Cpu - C3po's mother.
 - ✓Dip - Inventer of a famous switch.
 - ✓Disk Drive - A motor for a frisbee.
 - ✓Duplex - Having two apartments.
 - ✓Forth - One of the top five computer languages.
 - ✓GoGo - Garbage in Garbage out.

KANSAS CITY TI99/4A COMPUTER USERS' GROUP
POST OFFICE BOX 12591
NORTH KANSAS CITY, MISSOURI 64116



FIRST CLASS MAIL



The K.C. TI99/4A Users' Group is in no way affiliated with or sponsored by TI. Mention of a company or product is not considered an endorsement. Opinions expressed are those of the writers and not necessarily those of the Group, its officers, editor and the membership. Information from this Newsletter may be reproduced only if credit is given to both the author and this newsletter.

| NEWSLETTER ONLY MEMBERSHIP \$12/yr |
USER GROUP MEMBERSHIP IS \$20/yr

AMFJ89DN0SAB8
DALLAS TI HOME
Computer Group
P. O. Box 29863
Dallas TX 75229

(continued from page 9)

- ✓Ibm - In London a frequent answer to the question: 'Where is the tour guide?' "i be 'im".
- ✓IC - Understanding as in 'Oh, IC'.
- ✓Initialize - Carving your initials on a floppy disk.
- ✓Interate - A healthy illiterate.
- ✓Joystick - A peripheral intended for use only by consenting adults.
- ✓Keyboard - Resembling a typewriter, a keyboard is used for entering errors into the computer.
- ✓Kilo - What you could have spent your money on if you hadn't bought the computer.
- ✓Language - A system of organizing and defining syntax errors.
- ✓Math Chip - A piece of broken abacus.
- Memory Map - A sheet of paper showing the location of a computer store.
- Mhz - Acronym for 'Megahurtz', meaning 'a million pains'.
- Microfiche - Sardines.
- Nanosecond - Mork's stunt man.
- Newdos - Acronym for 'Not Exactly What The Dealer Offers To Sell'.
- Fascal - Computer language used for college football player known in full as 'Pass Calvin or Else'.
- Password - The nonsense word taped to the CRT.

(Reprinted from the September 1988 Pomona Valley newsletter)