

PARCO
Electrics

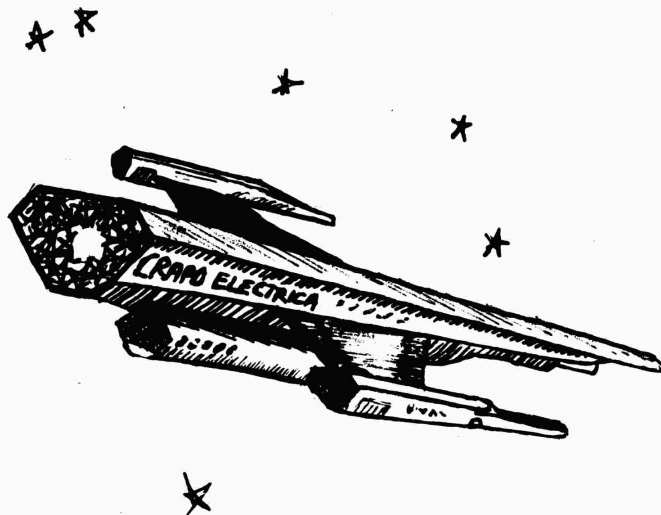
99/4A

M.V.



**INDEPENDENT MAGAZINE FOR THE
USERS OF THE T.I. 99/4A HOME
COMPUTER**

**VOLUME 2
ISSUE 2**



THE PARCO STORY - continued

No No No..Too late!!! Hardly Didmore had woken up, hit the double hyper drive switch and was thrown out into space faster than the speed of a mail shot. Would life on the Starship Electrica continue? Was there ever life on the Starship?

Enter Petty Gripe with a message.. I don't understand but it reads.... "CALL INIT :: CALL LOAD(-31804,34)" The message was fed into the ships main 994a computer which immediately malfunctioned. Wait cried Petty theres more."THIS COMMAND WILL DISABLE THE 994s KEYBOARD.signed R.Twyning."

Not him again muttered Petty as he tucked into a cheese and mango chutney roll which had clearly seen better days.

Fear not cried Frankly, who with the flick of a switch took command of the ships new more powerful TIPC. With this we can take over the world, we'll start with Honitonia.

Enter Colly Frears from the ship Lomdin. He has water cooled teeth, which allow him to eat faster than any other Earthling without serious overheating. Steals food from you if he thinks you are not looking, pokes a finger in your eye if you are. Has one good point, can count to six.

Everybodys hands were going blue the problem was traced to the ships new printing press, which was turning out price lists for the next mail shot.

!!!!!! The words mail shot had a dramatic effect on Petty who had again run off mumbling something into his half empty lunch box.



Night fell on the Starship, luckily no one was hurt. Salad Rosebowl gradually drifted into a deep sleep, still wondering what a 32k centronics extended parsec was doing on an order from the Ministry of Transport. Petty was having difficulty sleeping and had been counting mail bags for the last hour and a half or so.

Frankly was thinking of his next move when, in walked Snoozy Frears screaming I'm here I'm here. She looked formidable in her 1930s cloak which had obviously been bought from a passing Oxfamius ship. She waded in totally out of control, it took six of the crew to get her off the control panel and restore some sort of order.

Franklys next move was now quite clear to him. Grabbing the keys to his MR2 shuttle he made a dash for the door, and was in the shuttle heading for Bedfordius before anyone could stop him. Unfortunately the power of the Blue Light drew alongside, something to do with speed. I don't take drugs pleaded Frankly, but to no avail. Judgment was made by the Inter Galactic Must Not Warp too Fast Tribe. Frankly was to suffer for 21 light days without flying permission.

Hardly gone, the Frears were creating havoc, the magazine in ruin, things did not add up.

Calculators....that was the answer, they were planted all over the ship. Things now began to add up. The TIPC was doing well, the mail shot was out, the magazine ready, Salad Rosebowl had sorted out the order for the Ministry of Transport and Pettys lunch box had been replenished.

Frankly sat in his chair and in a voice of contentment turned to Petty..

Ahead.. warp factor one Mr Gripe.

..... Steady as she goes.

CONTENTS:

PAGE.

2	THE PARCO STORY
4	LETTERS
5	HALLEYS COMPETITION - RESULTS
6	SOUND ADVICE
9	KEYBOARD KRUNCHERS
13	ODD ONE OUT
17	MORE FROM JO ANN TI. WRITER
20	HELP - ON ADVENTURES
22	BRIGHT SPARKS
26	REVIEW - DEFENDER
27	THE RICHARD TWYNING UTILITIES
30	KIDS CORNER
32	HIGH SCORES & TOP FIVE

c 1986 PARCO ELECTRICS

**99/4A MAGAZINE
IS PRODUCED BY**

PARCO ELECTRICS

**2. DEVONSHIRE COURT,
HEATHPARK,**

HONITON,

DEVON.

EX14 8SE

LETTERS LETTERS LETTERS
LETTERS

Dear Sirs,

For those, who like me, own a Mini Memory Module, yet cannot make full use of it because of a total lack of understanding of Assembly language, may I recommend a book ideal for the beginner.

The book is called "Fundamentals of TI-99/4A Assembly Language" and is by M.S. Morley (Tab books, ISBN 0-8306-1722-1) and can be ordered from W.H. Smiths Price 11.50

Perhaps you could print this for the benefit of other TI owners.

I enjoy your magazine immensely, and enclose payment for the next issue.

Keep up the good work

Yours faithfully

M.G. POSKITT

Don't forget Parco have Compute's Beginner's Guide to Assembly Language on the TI 99/4A by P. Lottrup Price 11.95

ED



NEW EDITOR

Dear Parco

Hello again. I see that you are already on Vol.2 of the Mag, the latest issue I have is Vol.1.No.6. Where is No.7 etc???

Glad to see a more "Comic sized" mag, the big ones are easily crumpled.

Still got Crapo Electrica. I see that this has something to do with you, letting everyone know your feelings behind the scenes, disguising your names, e.g. Poorish=Parrish, but I don't know Gripe

(nor do I, ED)

give me a few more issues and all will be clear ((Will it?))

You know your high scores? How do you accept scores and know that someone is lying? Anyone can say "I got Ten Million on Parsec, etc".

May the 4a go on forever.

Yours,

P.A. TOMPKINS

In reply to Mr. Tompkins

Thanks for your letter Mr. Tompkins. I hope these answers will convince you that the PARCO MOB are completely MAD.

There is no No.7 magazine Vol 1 No.7 is in fact Vol 2 No.1, Vol 1 No.8 that's this mag, is Vol 2 No.2. Clear now !!!!!

Now to clarify the characters of Crapo Electrica:

Frankly Poorish is Francis Parrish

Petty Gripe is Peter Grigg

Hardly Didmore was Harry Pridmore

Salad Rosebowl is Sally

Annually Poorish paid Ten Pounds

to have her name in the mag

Rubric Gripe paid Twenty

Colly Frears is Colin Mears

Snoozey Frears paid Petty Gripe

two cheese rolls

Philament Loving ... teleports in

Saturdays only

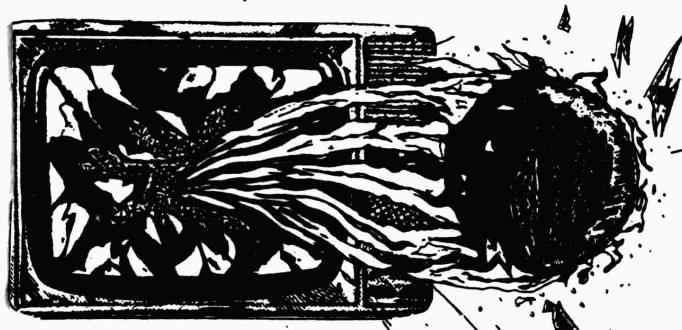
TI owners don't lie about high scores or else.

ED

HALLEYS COMPETITION

RESULTS

Extended Basic



WELL YET AGAIN YOU HAVE SURPRISED US WITH SOME NICE ENTRIES FOR OUR COMPETITION. IN FACT WE FELT THAT NOT ONLY SHOULD WE GIVE A FIRST PRIZE OF A FIFTY POUND VOUCHER, BUT ALSO A SECOND AND THIRD PRIZE EACH OF A TEN POUND VOUCHER. IN FIRST PLACE AND THE WINNER OF THE FIFTY POUND VOUCHER EVEN THOUGH IT TOOK 25 ATTEMPTS TO ACTUALLY LOAD THE PROGRAM, WAS

```
*****
*                               *
*   FIRST PLACE               *
*                               *
*   STEPHEN BARNES           *
*                               *
*   50                        50 *
*****
```

IN SECOND PLACE AND THE WINNER OF A TEN POUND VOUCHER

```
*****
*                               *
*   SECOND PLACE             *
*                               *
*   NED DOUGLAS              *
*                               *
*   10                       10 *
*****
```

IN THIRD PLACE AND THE WINNER OF A TEN POUND VOUCHER

```
*****
*                               *
*   THIRD PLACE              *
*                               *
*   PETER HUTCHINSON         *
*                               *
*****
```

THE FIRST PLACE LISTING FOLLOWS IT CONTINUES ON PAGES 16 & 19 THE FIRST SCREEN IS NOT MIND BLOWING BUT THE NEXT IS.....

TO PAGE 16

```
100 CALL CLEAR :: CALL CHARS : 200 FOR T=3 TO 8 :: CALL COL
ET :: CALL MAGNIFY(1):: ON W : OR(T,3,1,9,15,1):: NEXT T ::
ARNING NEXT :: RANDOMIZE :: : CALL CHAR(98,"0080C0FEFFFE4
CALL COLOR(9,16,2):: CALL KE : 4",97,"1C1C087F081424"):: CA
Y(3,K,S) : LL COLOR(1,11,1,2,7,1):: RET
110 FOR T=1 TO 14 :: CALL CO : URN
LOR(T,3,1):: NEXT T :: CALL : 210 GOSUB 150 :: FOR T=30 TO
CLEAR :: PRINT "THE COMET BY : 1 STEP -1 :: DISPLAY AT(1,1
STEPHEN BARNES. ~~~~~~ )BEEP:"TIME LEFT = ";T;" SEC
~~~~~
120 CALL SCREEN(2):: PRINT : EXT H :: NEXT T :: CALL CLEA
:"YOU HAVE GOT 30 SECONDS T : R :: CALL COLOR(0,16,1):: CA
O" : : "PLAN YOUR WAY TO YOU : LL DELSPRITE(ALL)
R SPACE" : : "SHIP.BUT YOU O : 220 PRINT : "ENTER YOUR ROU
NLY HAVE 3" : : : TE TO YOUR SHIP." : : "U=
130 PRINT "TRIES TO GET TO Y : UP":"D=DOWN":"L=LEFT":"R=RIG
OUR SHIP OR" : : "IT WILL LE : HT" : : INPUT "ENTER ROUTE
AVE WITHOUT YOU..." : : "YOU >" : A$ :: IF LEN(A$)<1 THEN 2
MUST NOT GO IN THE RED." : : 20 ELSE CALL CLEAR :: GOSUB
: : FOR T=1 TO 14 :: CALL : 150
COLOR(T,3,1):: NEXT T : 230 DISPLAY AT(1,1):"LETS SE
140 DISPLAY AT(24,1):"PRESS : E IF YOUR RIGHT." :: MANA=22
ANY KEY TO PLAY." :: CALL KE : : MANB=16 :: FOR T=1 TO LE
Y(3,K,S):: IF S=0 THEN 140 E : N(A$):: IF SEG$(A$,T,1)="L"
LSE TRY=3 :: GOTO 210 : THEN MANB=MANB-1
150 CALL CLEAR :: CALL CHARS : 240 IF SEG$(A$,T,1)="R" THEN
ET :: CALL SPRITE(1,97,9,22 : MANB=MANB+1 ELSE IF SEG$(A$
8-6,168-8):: CALL CHAR(42, : ,T,1)="U" THEN MANA=MANA-1 E
"FFFFFFFFFFFFFFFF",33,"FF818 : LSE IF SEG$(A$,T,1)="D" THEN
181818181FF") : MANA=MANA+1
160 PRINT "*****b*****" : 250 CALL GCHAR(MANA,MANB,GET
*****!!!!!!!!!!!!!! : ): CALL LOCATE(1,MANA8-6,
!!!!!!!!!!!!!! : MANB8-7):: IF GET=98 THEN 3
!!!!!!!!!!!!!! : 10 ELSE IF GET<33 THEN DISP
!!!!!!!!!!!!!! : LAY AT(1,1):"YOU WALKED IN T
!!!!!!!!!!!!!! : HE RED" ELSE 300
170 PRINT "*****" : 260 CALL SOUND(-50,110,0)::
***** : TRY=TRY-1 :: FOR G=1 TO 1000
***** : : NEXT G :: IF TRY=0 THEN
***** : 280 ELSE DISPLAY AT(1,1):"WE
***** : WILL HAVE TO TRY AGAIN"
***** : 270 FOR G=1 TO 1000 :: NEXT
180 PRINT "*****" : G :: CALL CLEAR :: CALL COLO
***** : R(1,1):: GOTO 210
***** : 280 CALL DELSPRITE(ALL):: CA
***** : LL CLEAR :: CALL COLOR(0,16,
***** : 1):: PRINT "I AM AFRAID YOUR
***** : SPACE SHIP HAS LEFT WITHOUT
190 PRINT "*****" : YOU SO YOURMISSION HAS FAIL
***** : ED." :: FOR G=1 TO 2000
***** : 290 NEXT G :: CALL CLEAR ::
***** : GOTO 1030
***** : 300 CALL SOUND(-100,-1,0)::
***** : NEXT T :: GOTO 260
```

SOUND ADVICE

Extended Basic

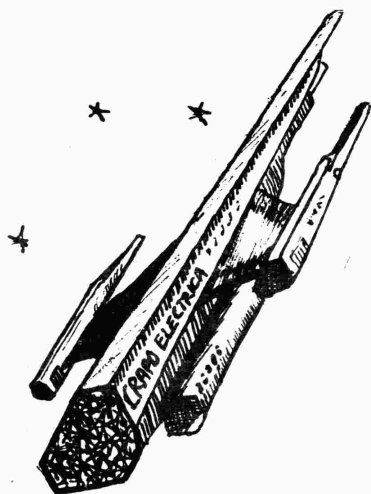
MUSIC TUTOR PART 2

by PHIL DONALD

THE MUSIC TUTOR PART 2 IS ENTIRELY MENU DRIVEN. PHIL HAS TRIED TO MAKE THE TUTOR AS COMPREHENSIVE AS POSSIBLE WITHOUT GOING INTO TOO FINE A DETAIL.

ALTHOUGH THE CHORDS USED IN "MUSIC TUTOR PART 1" MAY HAVE APPEARED STRANGE (TO A MUSICIAN) THEY WERE CHOSEN FOR THEIR CLARITY. THE CHORDS WERE NOT TONIC TRIADS, THEY WERE JUST VARIATIONS OF THEM. WE WILL BE PUBLISHING A PROGRAM SHORTLY IN THIS MAGAZINE FEATURING ALL THE MAJOR TONIC TRIADS AND HOW THEY ARE CONSTRUCTED.

HOPE YOU ENJOY THIS MONTHS MUSIC TUTOR. WE WOULD LOVE TO HAVE YOUR COMMENTS, PRAISE OR CRITICISM.



```

100 REM MUSIC TUTOR PART 2 : 230 CALL CLEAR :: DISPLAY AT
110 REM BY PHIL DONALD : (5,14): "MENU" :: DISPLAY AT(
120 REM FOR PARCO : 6,14): "----" :: DISPLAY AT(1
130 CALL CLEAR :: PRINT : " : 0,2): "PRESS" :: CALL COLOR(1
DEFINING CHARS NOW" : 4,5,5): CALL HCHAR(1,1,136,
140 CALL MAGNIFY(3):: CALL C : 32):: CALL HCHAR(24,1,136,32
HAR(116,"0101010101010101D3 : )
F7F7F3E1C"&RPT$( "0",32),128, : 240 CALL VCHAR(1,2,136,24)::
"0101010101010101D3F7FFF7F3 : CALL VCHAR(1,32,136,24):: D
E1C00"&RPT$( "0",32)) : ISPLAY AT(14,2): "1) MAJOR SC
150 CALL CHAR(33,"FF",140,"0 : ALES" :: DISPLAY AT(16,2): "2
000003C7EFF7F3D01010101010 : ) NOTE VALUES" :: DISPLAY AT
10101"&RPT$( "0",32)):: CALL : (18,2): "3) END"
SCREEN(16):: RESTORE 160 :: : 250 CALL KEY(0,K,S):: IF S=0
FOR LC=1 TO 27 :: READ CH,CH : THEN 250 :: K=K+48 :: IF K<
$ :: CALL CHAR(CH,CH$):: NEX : 1 OR K>3 THEN 250 :: ON K GO
T LC : TO 270,1440,260
160 DATA 34,0000010102020404 : 260 FOR S=1 TO 4 :: CALL SOU
,35,0808081010101020,36,2020 : ND(-100,110*S,1,220*S,1):: N
204040404040,37,404040202010 : EXT S :: STOP
100C,38,0201,39,010604040404 : 270 CALL CLEAR :: FOR R=2 TO
03,101,202040408 : 10 STEP 2 :: CALL HCHAR(R,6
170 DATA 42,0000000101010101 : ,33,26):: NEXT R :: RESTORE
,43,0202020202020202,135,030 : 280 :: FOR CH=1 TO 28 :: REA
306060A122242,58,42820202020 : D RX,CX,DX :: CALL HCHAR(RX,
20202,59,02020202031E2222,60 : CX,DX):: NEXT CH
,4242424242424262 : 280 DATA 5,2,34,6,2,35,7,2,3
180 DATA 61,1E02020202020202 : 6,8,2,37,9,2,38,10,2,39,2,3,
,62,02FF010101010101,63,0101 : 42,3,3,43,4,3,135,5,3,58,6,3
0101010101FE,64,000000001E21 : ,59,7,3,60
4040,91,80808,92,00010106081 : 290 DATA 8,3,61,9,3,62,10,3,
060C,93,8 : 63,1,4,64,2,4,91,3,4,92,4,4,
190 DATA 94,00000000FC03,95, : 93,6,4,94,8,4,95,9,4,96,1,5,
0000000000010204,96,18E,97,0 : 97,2,5,98
000000000008040,98,404040404 : 300 DATA 3,5,93,6,5,99,7,5,1
0408080,99,00000000000C020, : 00,8,5,101
100,1010101010101020 : 310 RESTORE 320 :: FOR SP=1
200 CALL CLEAR :: CALL CHAR( : TO 8 :: READ LX,LY :: CALL S
120,"2424FF242424FF2424"&RPT : PRITE(#SP,116,2,LX,LY):: NEX
$( "0",46),124,"4040405E62424 : T SP
C70"&RPT$( "0",48)):: CALL CO : 320 DATA 78,48,67,74,62,100,
LDR(14,9,9) : 54,126,46,152,38,174,30,200,
210 CALL HCHAR(1,1,136,32):: : 23,226
CALL HCHAR(24,1,136,32):: C : 330 DISPLAY AT(16,1): "DO YOU
ALL VCHAR(1,2,136,24):: CALL : REQUIRE EXPLANATION OF HOW
VCHAR(1,32,136,24):: DISPLA : A MAJOR SCALE IS CONSTR
Y AT(4,13): "MUSIC" : UCTED Y/N"
220 DISPLAY AT(6,14): "BY" :: : 340 CALL KEY(0,K,S):: IF S=0
DISPLAY AT(8,10): "PHIL DONA : THEN 340 :: IF K=78 THEN 50
LD" :: DISPLAY AT(10,12): "PA : 0 :: GOSUB 1520 :: DISPLAY A
RT 2" :: GOSUB 1510 : T(16,1): "BEFORE YOU CHOOSE Y
: OUR MAJORS CALES FROM THE MEN
: U IT IS"

```

SOUND ADVICE

```

350 DISPLAY AT(18,1):"IMPORT : 440 DISPLAY AT(16,1):"THE DI
ANT TO KNOW HOW THEY ARE CO : STANCE BETWEEN THE FOURTH
NSTRUCTED" :: GOSUB 1510 :: : AND FIFTH NOTES IS A FULL T
GOSUB 1520 :: DISPLAY AT(16, : ONE, SO UP A FULL TONEFROM F
1):"TAKE, FOR EXAMPLE, THE S : IS 6" :: GOSUB 1510 :: GOSU
CALED OF C MAJOR" : B 1520 :: CALL COLOR(4,5,8
360 DISPLAY AT(18,1):"FIRSTL : ,13)
Y, WRITE OUT YOUR STAFF, : 450 DISPLAY AT(16,1):"THE DI
OR STAVE (THE FIVE LINES : STANCE BETWEEN THE FIFTH
AND FOUR SPACES) AND ADD TH : AND SIXTH NOTE IS ALSO A FULL
E CLEF. IN THIS CASE THE TR : TONE, SO UP A FULL TONE F
EBLE OR 6 CLEF" : ROM 6 IS A" :: GOSUB 1510 ::
370 GOSUB 1510 :: GOSUB 1520 : GOSUB 1520 :: CALL COLOR(5
:: DISPLAY AT(16,1):"THEN M : ,5,8,13)
RITE IN THE NOTES AS ABOVE. : 460 DISPLAY AT(16,1):"THE DI
THE DISTANCE BETWEEN THE FI : STANCE BETWEEN THE SIXTH
RST TWO NOTES MUST BE A FULL : AND SEVENTH NOTE IS A FULL T
TONE" : ONE, SO UP A FULL TONEFROM A
380 DISPLAY AT(20,1):"A FULL : IS B" :: GOSUB 1510 :: GOSU
TONE UP FROM C IS D. YOU NO : B 1520 :: CALL COLOR(6,5,8
W HAVE THE SECOND NOTED OF THE : ,13)
SCALE" :: CALL COLOR(1,13, : 470 DISPLAY AT(16,1):"THE DI
#2,13) :: GOSUB 1510 :: GOSUB : STANCE BETWEEN THE SEVENT
1520 : H AND EIGHTH NOTE IS A SEMITO
390 CALL COLOR(1,5,8,13) :: NE, SO UP A SEMITONE FROM B
DISPLAY AT(16,1):"THE DISTA : IS C" :: GOSUB 1510 :: GOSU
NCE BETWEEN THE SECOND NO : B 1520 :: CALL COLOR(7,5,8
TE AND THE THIRD NOTE MUST : ,5)
ALSO BE A FULL" : 480 DISPLAY AT(16,1):"YOU WI
400 DISPLAY AT(19,1):"TONE, : LL HAVE NOTICED THAT THE PA
SO A FULL TONE UP FROM D IS E : TERN FOR THE SECOND TETRAC
" :: GOSUB 1510 :: GOSUB 152 : HORD IS EXACTLY THE SAME A
0 :: CALL COLOR(2,5,8,13) : S THE FIRST." :: GOSUB 1510
: DISPLAY AT(16,1):"THE DIST : :: GOSUB 1520 :: CALL DELSPR
ANCE BETWEEN THE " : ITE(ALL)
410 DISPLAY AT(17,1):"THIRD : 490 GOTO 330
AND FOURTH NOTES MUST BE A S : 500 CALL CLEAR :: CALL DELSP
EMITONE, SO A SEMITONE UP FRO : RITE(ALL)
M E IS F." :: GOSUB 1510 :: : 510 PRINT " MAJOR SCA
GOSUB 1520 :: FOR R=1 TO 4 : LES" : : "SCALE
: CALL COLOR(8,5) :: NEXT R : RESS " : "A
420 DISPLAY AT(16,1):"THE GR : A B FLAT/A SHARP
OUPING OF THESE FOUR NOTES : BF B
ARE REFERRED TO AS A TETRAC : B"
HORD. THERE ARE TWO OF THE : 520 PRINT "C
SE IN ALL MAJOR SCALES" :: G : C C SHARP/D FLAT
OSUB 1510 :: GOSUB 1520 : CS D
430 DISPLAY AT(16,1):"HAVING : D E FLAT/D SHARP
COMPLETED THE FIRST TETRAC : EF E
HORD WE CAN PROCEED TO THE SE : E"
COND ONE AND COMPLETE OUR MA :
JOR SCALE." :: GOSUB 1510 ::
GOSUB 1520 :: CALL COLOR(4 :
,13,8,13) :

```

```

530 PRINT "F : 690 RESTORE 700 :: FOR N=1
F F SHARP/6 FLAT : 0 8 :: READ M :: CALL SOUND
FS 6 : 500,M,1) :: NEXT N
G A FLAT/6 SHARP : 700 DATA 233,262,294,311,34
AF" : "Q FOR MENU" : ,392,440,466
540 GOTO 570 : 710 DISPLAY AT(18,1):"B FLA
550 FOR R=4 TO 12 STEP 2 :: : MAJOR" :: GOSUB 1530 :: GO
CALL HCHAR(R,6,33,26) :: NEXT : 0 690
R :: RESTORE 280 :: FOR CH= : 720 IF A$="B" THEN 730 ELSE
1 TO 28 :: READ RX,CX,DX :: : 790
CALL HCHAR(RX+2,CX,DX) :: NEX : 730 GOSUB 550 :: RESTORE 740
T CH : : FOR SP=1 TO 13 :: READ C
560 FOR C=6 TO 32 STEP 2 :: : H,CO,RX,CX :: CALL SPRITE(8
CALL HCHAR(13,C,46) :: CALL H : P,CH,CO,RX,CX) :: NEXT SP
CHAR(1,C,46) :: NEXT C :: RET : 740 DATA 116,2,102,60,116,2,
URN : ,93,80,116,2,83,100,116,2,78,
570 ACCEPT AT(24,1)VALIDATE( : 120,116,2,70,140,116,2,61,16
UALPHA)SIZE(2):A$ :: CALL CL : 0,116,2,55,180
EAR :: IF A$="Q" THEN 230 : 750 DATA 116,2,46,200,120,5,
580 IF A$="A" THEN 590 ELSE : 76,130,120,5,99,70,120,5,68,
650 : 150,120,5,89,90,120,5,61,170
590 GOSUB 550 :: RESTORE 600 : 760 RESTORE 770 :: FOR N=1 1
:: FOR SP=1 TO 11 :: READ C : 0 8 :: READ M :: CALL SOUND(
H,CO,RX,CX :: CALL SPRITE(8 : 500,M,1) :: NEXT N
P,CH,CO,RX,CX) :: NEXT SP : 770 DATA 247,277,311,330,370
600 DATA 116,2,55,60,116,2,4 : ,415,466,494
6,80,116,2,38,100,116,2,29,1 : 780 DISPLAY AT(18,1):"B MAJO
20,116,2,23,140,116,2,14,160 : R" :: GOSUB 1530 :: GOTO 760
,116,2,8,180 : 790 IF A$="C" THEN 800 ELSE
610 DATA 140,2,3,200,120,5,4 : 860
6,90,120,5,21,150,120,5,16,1 : 800 GOSUB 550 :: RESTORE 810
70 : : FOR SP=1 TO 8 :: READ CK
620 RESTORE 630 :: FOR N=1 T : ,CO,RX,CX :: CALL SPRITE(8SP
0 8 :: READ M :: CALL SOUND( : ,CH,CO,RX,CX) :: NEXT SP
500,M,1) :: NEXT N : 810 DATA 116,2,93,60,116,2,8
630 DATA 440,494,554,587,659 : 3,80,116,2,78,100,116,2,70,1
,740,831,880 : 20,116,2,61,140,116,2,55,160
640 DISPLAY AT(18,1):"A MAJO : ,116,2,46,180
R" :: GOSUB 1530 :: GOTO 620 : 820 DATA 116,2,38,200
650 IF A$="BF" THEN 660 ELSE : 830 RESTORE 840 :: FOR N=1 T
720 : 0 8 :: READ M :: CALL SOUND(
660 GOSUB 550 :: RESTORE 670 : 500,M,1) :: NEXT N
:: FOR SP=1 TO 11 :: READ C : 840 DATA 262,294,330,349,392
H,CO,RX,CX :: CALL SPRITE(8 : ,440,494,523
P,CH,CO,RX,CX) :: NEXT SP : 850 DISPLAY AT(18,1):"C MAJO
670 DATA 116,2,102,60,116,2, : R" :: GOSUB 1530 :: GOTO 830
93,80,116,2,83,100,116,2,78, : 860 IF A$="CS" THEN 870 ELSE
120,116,2,70,140,116,2,61,16 : 940
0,116,2,55,180 : 870 GOSUB 550 :: RESTORE 880
680 DATA 116,2,46,200,124,13 : : FOR SP=1 TO 16 :: READ C
,108,50,124,13,84,110,124,13 : H,CO,RX,CX :: CALL SPRITE(8
,52,190 : P,CH,CO,RX,CX) :: NEXT SP
:

```

SOUND ADVICE

```

880 DATA 116,2,93,60,116,2,8 : 1050 RESTORE 1060 :: FOR N=1
3,80,116,2,78,100,116,2,70,1 : TO 8 :: READ M :: CALL SOUN
20,116,2,61,140,116,2,55,160 : D(500,M,1):: NEXT N
,116,2,46,180 : 1060 DATA 311,349,392,415,46
890 DATA 116,2,38,200,120,5, : 6,523,587,622
100,50,120,5,91,70,120,5,85, : 1070 DISPLAY AT(18,1):"E FLA
90,120,5,78,110,120,5,69,130 : T MAJOR" :: GOSUB 1530 :: GO
,120,5,63,150 : TO 1050
900 DATA 120,5,53,170,120,5, : 1080 IF A$="E" THEN 1090 ELS
46,190 : E 1150
910 RESTORE 920 :: FOR N=1 T : 1090 GOSUB 550 :: RESTORE 11
0 8 :: READ M :: CALL SOUND( : 00 :: FOR SP=1 TO 12 :: READ
500,M,1):: NEXT N : CH,CO,RX,CX :: CALL SPRITE(
920 DATA 277,311,349,370,415 : #SP,CH,CO,RX,CX):: NEXT SP
,466,523,554 : 1100 DATA 116,2,78,60,116,2,
930 DISPLAY AT(16,1):"NOTE T : 70,80,116,2,61,100,116,2,55,
HAT ALL NOTES ARE SHARP : 120,116,2,46,140,116,2,38,16
IN C SHARP MAJOR E SHAR : 0,116,2,29,180
P IS F, B SHARP IS C" :: GOS : 1110 DATA 116,2,23,200,120,5
UB 1530 :: GOTO 910 : ,78,70,120,5,46,150,120,5,37
940 IF A$="D" THEN 950 ELSE : ,170,120,5,69,90
1010 : 1120 RESTORE 1130 :: FOR N=1
950 GOSUB 550 :: RESTORE 960 : TO 8 :: READ M :: CALL SOUN
:: FOR SP=1 TO 10 :: READ C : D(500,M,1):: NEXT N
H,CO,RX,CX :: CALL SPRITE(#S : 1130 DATA 330,370,415,440,49
P,CH,CO,RX,CX):: NEXT SP : 4,554,622,659
960 DATA 116,2,83,60,116,2,7 : 1140 DISPLAY AT(18,1):"E MAJ
8,80,116,2,70,100,116,2,61,1 : OR" :: GOSUB 1530 :: GOTO 11
20,116,2,55,140,116,2,46,160 : 20
,116,2,38,180 : 1150 IF A$="F" THEN 1160 ELS
970 DATA 116,2,29,200,120,5, : E 1220
78,90,120,5,46,170 : 1160 GOSUB 550 :: RESTORE 11
980 RESTORE 990 :: FOR N=1 T : 70 :: FOR SP=1 TO 9 :: READ
0 8 :: READ M :: CALL SOUND( : CH,CO,RX,CX :: CALL SPRITE(#
500,M,1):: NEXT N : SP,CH,CO,RX,CX):: NEXT SP
990 DATA 294,330,370,392,440 : 1170 DATA 116,2,70,60,116,2,
,494,554,587 : 62,80,116,2,55,100,116,2,46,
1000 DISPLAY AT(18,1):"D MAJ : 120,116,2,38,140,116,2,29,16
OR" :: GOSUB 1530 :: GOTO 98 : 0,116,2,23,180
0 : 1180 DATA 116,2,14,200,124,1
1010 IF A$="EF" THEN 1020 EL : 3,52,110
SE 1080 : 1190 RESTORE 1200 :: FOR N=1
1020 GOSUB 550 :: RESTORE 10 : TO 8 :: READ M :: CALL SOUN
30 :: FOR SP=1 TO 12 :: READ : D(500,M,1):: NEXT N
CH,CO,RX,CX :: CALL SPRITE( : 1200 DATA 349,392,440,466,52
#SP,CH,CO,RX,CX):: NEXT SP : 3,587,659,698
1030 DATA 116,2,78,60,116,2, : 1210 DISPLAY AT(18,1):"F MAJ
70,80,116,2,61,100,116,2,55, : OR" :: GOSUB 1530 :: GOTO 11
120,116,2,46,140,116,2,38,16 : 90
0,116,2,29,180 : 1220 IF A$="FS" THEN 1230 EL
1040 DATA 116,2,23,200,124,1 : SE 1300
3,84,50,124,13,61,110,124,13 :
,29,190,124,13,52,130 :

```

```

1230 GOSUB 550 :: RESTORE 12 : 1400 DATA 140,2,3,200,124,13
40 :: FOR SP=1 TO 15 :: READ : ,61,50,124,13,52,70,124,13,3
CH,CO,RX,CX :: CALL SPRITE( : 6,110,124,13,29,130,124,13,5
#SP,CH,CO,RX,CX):: NEXT SP : ,190
1240 DATA 116,2,70,60,116,2, : 1410 RESTORE 1420 :: FOR N=1
62,80,116,2,55,100,116,2,46, : TO 8 :: READ M :: CALL SOUN
120,116,2,38,140,116,2,30,16 : D(500,M,1):: NEXT N
0,116,2,23,180 : 1420 DATA 415,466,523,554,62
1250 DATA 116,2,14,200,120,5 : 2,698,784,831
,78,50,120,5,69,70,120,5,63, : 1430 DISPLAY AT(18,1):"A FLA
90,120,5,46,130,120,5,37,150 : T MAJOR" :: GOSUB 1530 :: GO
,120,5,31,170 : TO 1410
1260 DATA 120,5,21,190 : 1440 CALL CHARSET :: CALL CL
1270 RESTORE 1280 :: FOR N=1 : EAR :: CALL CHAR(34,"0000182
TO 8 :: READ M :: CALL SOUN : 4422418",35,"182438202020202
D(500,M,1):: NEXT N : 0",36,"183C382020202020",37,
1280 DATA 370,415,466,494,55 : "183C382024283020",38,"183C3
4,622,698,740 : 82428342830")
1290 DISPLAY AT(18,1):"F SHA : 1450 RESTORE 1460 :: FOR R=1
RP MAJOR" :: GOSUB 1530 :: 6 : TO 15 :: READ CR,CC,CN :: C
OTO 1270 : ALL HCHAR(CR,CC,CN):: NEXT R
1300 IF A$="G" THEN 1310 ELS : 20,35,7,10,36,7,14,36,7,18,3
E 1370 : 6,7,22,36,12,9,37,12,11,37,1
1310 GOSUB 550 :: RESTORE 13 : 2,13,37,12,15,37
20 :: FOR SP=1 TO 9 :: READ : 1470 DATA 12,17,37,12,19,37,
CH,CO,RX,CX :: CALL SPRITE(# : 12,21,37,12,23,37
SP,CH,CO,RX,CX):: NEXT SP : 1480 CALL HCHAR(17,8,38,16):
1320 DATA 116,2,62,60,116,2, : DISPLAY AT(20,1):"16 SEMIQ
55,80,116,2,46,100,116,2,38, : UAUERS = 8 QUAUERS = 4 CROTC
120,116,2,30,140,116,2,23,16 : HETS = 2 MINIMS = 1 SEMI
0,116,2,14,180 : BREVE"
1330 DATA 116,2,9,200,120,5, : 1490 GOSUB 1510 :: GOSUB 152
21,170 : 0 :: DISPLAY AT(16,1):"32 DE
1340 RESTORE 1350 :: FOR N=1 : MISEMIQUAUERS (NOT PICTU
TO 8 :: READ M :: CALL SOUN : RED HERE) ARE EQUAL TO 16 SE
D(500,M,1):: NEXT N : MIQUAUERS AND SO ON." :: DIS
1350 DATA 392,440,494,523,58 : PLAY AT(20,1):"M FOR MENU"
7,659,740,784 : 1500 CALL KEY(0,K,S):: IF S=
1360 DISPLAY AT(18,1):"G MAJ : 0 THEN 1500 :: CALL CLEAR ::
OR" :: GOSUB 1530 :: GOTO 13 : IF K<>77 THEN 1500 ELSE 140
40 : 1510 DISPLAY AT(23,10):"PRES
1370 IF A$="AF" THEN 1380 EL : S ANY KEY" :: CALL KEY(0,K,S
SE 500 : ): IF S=0 THEN 1510 :: RETU
1380 GOSUB 550 :: RESTORE 13 : RN
90 :: FOR SP=1 TO 13 :: READ : 1520 FOR R=16 TO 24 :: DISPL
CH,CO,RX,CX :: CALL SPRITE( : AY AT(R,1):" " :: NEXT R ::
#SP,CH,CO,RX,CX):: NEXT SP : RETURN
1390 DATA 116,2,55,60,116,2, : 1530 DISPLAY AT(24,1):"PLAY
46,80,116,2,38,100,116,2,30, : SCALE AGAIN Y/N" :: CALL KEY
120,116,2,23,140,116,2,14,16 : (0,K,S):: IF S=0 THEN 1530 :
0,116,2,8,180 : IF K=89 THEN RETURN ELSE 5
00

```

KRUNCH * KRUNCH * KRUNCH * KRUNCH * KRUNCH * KRUNCH

KEYBOARD KRUNCHERS

THE LIST *****

PAGE ****	TITLE *****	PROG IN *****
10	SUPER SOLITAIRE By D.MARTIN	BASIC
12	BASEMENT BOB By S.NASH	EX.BASIC
14	MISSILE DUEL By T.ANDERSON	EX.BASIC

KRUNCH
KRUNCH
KRUNCH
KRUNCH



KRUNCH * KRUNCH * KRUNCH
KRUNCH * KRUNCH * KRUNCH

I HOPE THIS LITTLE LOT WILL GET YOU KRUNCHING INTO THE MIDNIGHT HOURS. ALTHOUGH THE PROGRAMS HAVE BEEN WELL CHECKED FOR BUGS, IF YOU FIND A LITTLE NASTY ANYWHERE PLEASE LET ME KNOW, SO THAT I CAN LET EVERYONE KNOW.

SUPER SOLITAIRE BY D.MARTIN, IS A BOARD GAME THAT LOOKS SIMPLE UNTIL YOU GET HALF WAY THROUGH. BASEMENT BOB BY S.NASH, WELL IF YOU CAN SORT THIS ONE OUT YOU DESERVE A MEDAL. MISSILE DUEL BY T.ANDERSON IS A TWO PLAYER GAME, AN IQ OF AROUND 4000 WOULD HELP WITH THE STRATEGY OF THIS ONE.

IF YOU HAVE A PROGRAM YOU WOULD LIKE TO SUBMIT FOR LISTING IN KEYBOARD KRUNCHERS, OR IN FACT ANY ARTICLE FOR THE MAGAZINE. WRITE TO ME.

THE EDITOR
994 MAGAZINE
2. DEVONSHIRE COURT
HEATHPARK
HONITON
DEVON. EX14 8SB

I HOPE YOU ENJOY THIS MONTHS PROGRAMS

HAPPY KRUNCHING

SUPER SOLITAIRE

TI Basic

BY D. MARTIN ★

```

10 REM ** SUPER SOLITAIRE ** : 420 REM ** ADVANCED GAME ** : 860 CALL HCHAR(RW,18,136) : 1270 REM ** LEFT **
: 870 NEXT RW : 1280 CALL GCHAR(R-1,C-2,X)
20 RANDOMIZE : 430 C=2*INT(RND*9)+8 : 880 FOR CL=8 TO 24 STEP 2 : 1290 IF (X<>136)*(X<>144)THE
30 CALL CLEAR : 440 IF (C<14)+(C>18)THEN 470 : 890 MEM(10,CL)=5 : N 1040
40 CALL SCREEN(2) : 450 R=2*INT(RND*9)+4 : 900 CALL HCHAR(10,CL,136) : 1300 CALL HCHAR(R,C,32)
: 460 GOTO 480 : 910 MEM(12,CL)=5 : 1310 C=C-2
50 REM ** COLORS & CHARS ** : 470 R=2*INT(RND*3)+10 : 920 CALL HCHAR(12,CL,136) : 1320 GOTO 1020
: 480 AD$=CHR$(136) : 930 MEM(14,CL)=5 :
60 FOR S=1 TO 12 : 490 NM$=CHR$(144) : 940 CALL HCHAR(14,CL,136) : 1330 REM ** RIGHT **
70 CALL COLOR(S,8,1) : 500 GOTO 660 : 950 NEXT CL : 1340 CALL GCHAR(R-1,C+2,X)
80 NEXT S : 510 REM ** KEYBOARD ** : 960 MEM(R,C)=0 : 1350 IF (X<>136)*(X<>144)THE
90 FOR S=13 TO 16 : 520 KY=0 : 970 CALL HCHAR(R,C,144) : N 1040
100 READ C : 530 KY$=CHR$(136) : 980 LFT=44 : 1360 CALL HCHAR(R,C,32)
110 CALL COLOR(S,C,1) : 540 JY$=CHR$(144) : 990 STR=0 : 1370 C=C+2
120 NEXT S : 550 CALL HCHAR(21,3,136) : 1000 CALL CHAR(152,"8IFF") : 1380 GOTO 1020
130 FOR CH=1 TO 8 : 560 CALL HCHAR(23,3,144) : 1010 R=R+1 : 1390 CALL SOUND(-10,-1,0)
140 READ C,CH$ : 570 GOTO 320 : 1020 CALL HCHAR(R,C,152) :
150 CALL CHAR(C,CH$) : 580 REM ** JOYSTICK ** : 1030 REM ** MAIN LOOP ** : 1400 REM ** PICK UP **
160 NEXT CH : 590 KY=1 : 1040 IF KY=0 THEN 1070 : 1410 PEG=(MEM(R-1,C)=0)
: 600 KY$=CHR$(144) : 1050 CALL KEY(1,KE,S) : 1420 UP=((MEM(R-3,C)<>5)+(ME
170 REM ** FILL ARRAY ** : 610 JY$=CHR$(136) : 1060 CALL JOYST(1,A,U) : M(R-5,C)<>0))
180 DIM MEM(24,28) : 620 CALL HCHAR(21,3,144) : 1070 CALL KEY(3,K,S) : 1430 DOWN=((MEM(R+1,C)<>5)+(
190 FOR RW=2 TO 22 STEP 2 : 630 CALL HCHAR(23,3,136) : 1080 IF (K=69)+(U=4)THEN 116 : MEM(R+3,C)<>0))
200 FOR CL=6 TO 26 STEP 2 : 640 GOTO 320 : 1090 IF (K=88)+(U=-4)THEN 12 : 1440 LEFT=((MEM(R-1,C-2)<>5)
210 MEM(RW,CL)=2 : 650 REM ** SET UP SCREEN ** : 1100 IF (K=83)+(A=-4)THEN 12 : + (MEM(R-1,C-4)<>0))
220 NEXT CL : 660 CALL CLEAR : 1110 IF (K=68)+(A=4)THEN 134 : 1450 RIGHT=((MEM(R-1,C+2)<>5)
230 NEXT RW : 670 RESTORE 2100 : 1120 IF ((K=32)+(KE=18))*(ST : )+(MEM(R-1,C+4)<>0))
: 680 FOR CH=1 TO 12 : 1130 IF ((K=32)+(KE=18))*(ST : 1460 IF PEG+(UP*DOWN*LEFT*RI
240 REM *** MENU *** : 690 READ R1,C1,CH1,RP1,R2,C2 : R=0)THEN 1410 : GHT)THEN 2030
250 KY$=CHR$(136) : 700 CALL HCHAR(R1,C1,CH1,RP1 : 1140 IF K=13 THEN 1770 ELSE : 1470 STR=5
260 NM$=CHR$(136) : 710 CALL VCHAR(R2,C2,CH2,RP2 : 1150 REM ** UP ** : 1480 CALL HCHAR(R-1,C,144)
270 JY$=CHR$(144) : 720 NEXT CH : 1160 CALL GCHAR(R-3,C,X) : 1490 CALL HCHAR(23,7,136)
280 AD$=CHR$(144) : 730 M$="SUPER SOLITAIRE" : 1170 IF (X<>136)*(X<>144)THE : 1500 CALL CHAR(152,"81423C")
290 CALL CLEAR : 740 RW=1 : N 1040 : 1510 MEM(R-1,C)=0
300 PRINT " SUPER SOLI : 750 GOSUB 1970 : 1180 CALL HCHAR(R,C,32) : 1520 R1=R
TAIRE"::::" PRESS:"::NM$; : 760 M$=" DAVID MARTIN 1984" : 1190 R=R-2 : 1530 C1=C
: 1 FOR THE NORMAL GAME ">::AD : 770 RW=23 : 1200 GOTO 1020 : 1540 CALL SOUND(-50,550,0)
$: " 2 FOR THE ADVANCED GAME : 780 GOSUB 1970 : 1210 REM ** DOWN ** : 1550 GOTO 1040
: " : 790 REM ** SET UP BOARD ** : 1220 CALL GCHAR(R+1,C,X) : 1560 CALL SOUND(-10,-5,0)
310 PRINT :KY$;" 3 TO SELECT : 800 FOR RW=4 TO 20 STEP 2 : 1230 IF (X<>136)*(X<>144)THE : 1570 REM ** DROP **
KEYBOARD"::JY$;" 4 TO SELEC : 810 MEM(RW,14)=5 : N 1040 : 1580 R2=INT((R1+R)/2)-1
7 JOYSTICK" : 820 CALL HCHAR(RW,14,136) : 1240 CALL HCHAR(R,C,32) : 1590 C2=INT((C1+C)/2)
320 CALL KEY(3,K,S) : 830 MEM(RW,16)=5 : 1250 R=R+2 : 1600 IF (MEM(R-1,C)=5)+(MEM(
330 IF (K<49)+(K>52)THEN 320 : 840 CALL HCHAR(RW,16,136) : 1260 GOTO 1020 : R2,C2)=0)+(MEM(R1-1,C1)=5)TH
340 CALL SOUND(100,550,0) : 850 MEM(RW,18)=5 : 1610 CALL HCHAR(R-1,C,136) : EN 2030
350 ON K-48 GOTO 370,430,520 : 860 REM ** LEFT ** : 1620 CALL HCHAR(R2,C2,144)
,590 : 870 NEXT RW : 1630 CALL HCHAR(23,7,96)
360 REM ** NORMAL GAME ** : 880 FOR CL=8 TO 24 STEP 2 : 1640 CALL CHAR(152,"8IFF")
370 R=12 : 890 MEM(10,CL)=5 : 1650 MEM(R2,C2)=0
380 C=16 : 900 CALL HCHAR(10,CL,136) : 1660 MEM(R-1,C)=5
390 NM$=CHR$(136) : 910 MEM(12,CL)=5 : 1670 CALL SOUND(-50,660,0)
400 AD$=CHR$(144) : 920 CALL HCHAR(12,CL,136) : 1680 STR=0
410 GOTO 660 : 930 MEM(14,CL)=5 :

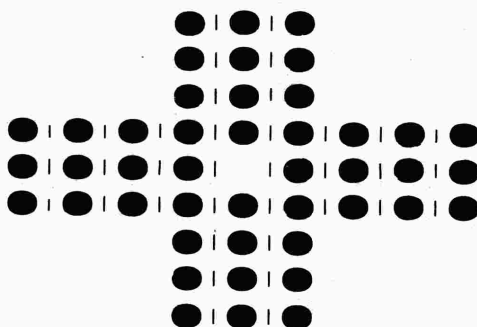
```

```

1690 LFT=LFT-1          : 1790 DN INT((44-LFT)/9)+1 GO : 1950 END          : 2080 DATA 136,3C7EFFFFFFF7E
1700 IF LFT>0 THEN 1040 : TO 1800,1820,1840,1860,18801 : 3C,144,3C4281818181423C,152,
1710 CALL CLEAR          : 800 PRINT "VERY POOR"          : 1960 REM ** PRINT AT ** : 81FF
1720 CALL SOUND(100,262,0,33 : 1810 GOTO 1900          : 1970 M=LEN(M$)          :
0,0,392,0)              : 1820 PRINT "POOR"          : 1980 CW=((32-M)/2)-1    : 2090 REM * DATA FOR BOARD *
1730 CALL SOUND(1000,262,0,3 : 1830 GOTO 1900          : 1990 FOR @=1 TO M      : 2100 DATA 3,14,128,5,4,19,12
30,0,392,0)            : 1840 PRINT "GOOD"         : 2000 CALL HCHAR(RW,CW+@,ASC( : 9,4,9,21,128,4,10,25,129,5,1
1740 PRINT " YOU CLEARED THE : 1850 GOTO 1900          : SEG$(M$,@,1)))        : 6,21,128,4,17,19,129,4
BOARD. . .":TAB(9);"WELL D : 1860 PRINT "VERY GOOD"    : 2010 NEXT @            : 2110 DATA 22,14,128,5,17,12,
ONE!":": YOU WERE FANTA : 1870 GOTO 1900          : 2020 RETURN            : 129,4,16,8,128,4,10,6,129,5,
STIC"                  : 1880 PRINT "EXCELENT"     : 2030 CALL SOUND(-20,-5,0) : 9,8,128,4,4,12,129,4
1750 GOTO 1900          :                               : 2040 GOTO 1040          : 2120 DATA 3,13,131,1,3,19,13
                               :                               :                               : 0,1,8,12,131,1,8,20,130,1,9,
                               :                               :                               : 7,131,1,9,25,130,1
1760 REM ** GIVE UP **   : 1900 PRINT :::"WOULD YOU LI : 2050 REM * DATA FOR CHARS * : 2130 DATA 15,7,130,1,15,25,1
1770 CALL CLEAR          : KE ANOTHER TRY ?"       : 2060 DATA 12,14,9,9    : 31,1,16,12,130,1,16,20,131,1
1780 PRINT "YOU MANAGED TO R : 1910 CALL KEY(3,K,S)     : 2070 DATA 96,3C4299A1A199423 : ,21,13,130,1,21,19,131,1
EMOVE";44-LFT::"PIECES FROM : 1920 IF K=89 THEN 290    : C,128,FF,129,010101010101010 :
THE BOARD":":RATING: "; : 1930 IF K<>78 THEN 1910  : 1,130,8040201008040201,131,0 :
                               : 1940 CALL CLEAR          : 102040810204080

```

2021 note-
yes, @ is an acceptable
variable.



SUPER SOLITAIRE BY DAVID MARTIN *****

THE OBJECT OF THE GAME IS TO REMOVE AS MANY "STONES" AS POSSIBLE.
THIS IS DONE BY JUMPING A STONE OVER ANOTHER WHICH MUST BE ADJACENT
TO THE STONE BEING PLAYED(AS IN DRAUGHTS).THE ULTIMATE IS TO LEAVE
THE LAST STONE IN THE MIDDLE OF THE BOARD

PLAYING THE GAME *****

KEYBOARD.. *****

USE THE ARROW KEYS "E.S.D.X" TO MOVE THE CURSER TO THE STONE
YOU WISH TO MOVE.
HIT THE SPACE BAR TO PICK UP THE STONE
USE THE ARROW KEYS "E.S.D.X" TO MOVE THE CURSER TO THE NEW
POSITION YOU WISH TO PLACE THE STONE
HIT THE SPACE BAR TO DROP THE STONE IN NEW POSITION.

JOYSTICK.. *****

USE STICK TO MOVE CURSER
FIRE BUTTON TO PICK UP AND PUT DOWN STONE

TO FINISH *****

PRESS ENTER TO FIND SCORE

BY **S. NASH**★

BASEMENT BOB

& THE SCARY MONSTERS

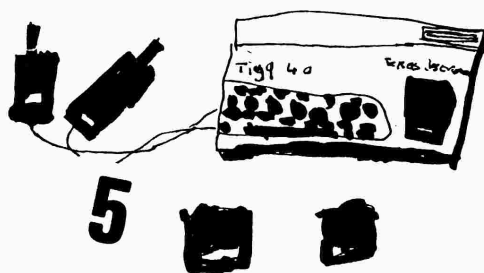
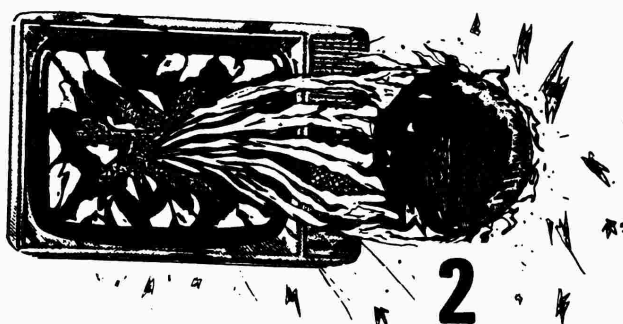
Extended Basic

```

100 ! multi-statement lines : 210 CALL CHAR(128,"0F3F7F777 : 350 DISPLAY AT(9,1):"if he s : 470 V=.5 :: J,Q=0 :: T=6 ::
increase speed of operation, : BFFFCDBDFEF77B7CEFCFC7EF0FCF : mashes all of the monster : 60SUB 690 :: M=-1 :: 60SUB 8
some multi-statement lines a : EEDEFF3FDBFBF7EEED733F3F3E : eggs \J by jumping on them : 40 :: CALL HCHAR(1,2,60,5)
re over 5 lines long.to ente : ):: A$="* * * * * : he will get a key >?" : 480 CALL SOUND(500,1760,0,44
r such a long line try this- : * * * * * : 360 DISPLAY AT(15,1):"and th : 0,5,220,8):: V=V*2 :: E=8 ::
110 ! type the line-but do n : 220 CALL CHAR(36,"1F7F7F7F1 : en takes the key back to the : M=1
ot type the spaces either si : F1F1FFFE7F7F3C0C0C0F0F8FEFE : door he came through and ju : 490 60SUB 750 :: 60SUB 840 :
de of the statement separato : FE8F8F8FF7F7FEFE3C3030F0E") : mps!more eggs will be laid f : : R=3-20*(E=0):: C=28 :: CAL
rs :: .when the 5 lines have : 230 CALL CHAR(136,"071F3F3B3 : or him to smash." : L SPRITE(#11,76,5+F,R*8-7,C*
been filled press enter,the : F1C1B03030D3146B4B4844E0FBE : 370 DISPLAY AT(21,1):" sound : 8-7):: 60SUB 770
n : CDCFC38D8F0C0B84C22212112") : s easy! not so!!" :: 60SUB 6 : 500 60SUB 850 :: CALL JOYST(
120 ! edit the line.the spac : : P=62 :: Q=2 :: CALL MAGNI : 80 : 1,X,Y):: IF Y THEN 60SUB 520
es will have been inserted b : FY(3) : 380 DISPLAY AT(6,1)BEEP:"ten : ELSE IF C+X/4=0 OR C+X/4=33
y the computer,pushig the la : 240 DISPLAY AT(22,6)ERASE AL : monsters are prowling!": : OR((R+2)/5)<>INT((R+2)/5)TH
st characters onto a 6th lin : L BEEP:"by s nash for parco" : bob avoids them by jumping. : EN 500 ELSE C=C+X/4
e onto which more can be typ : :: CALL CHAR(80,"0000000000 : but beware of the hungry : 510 W=SGN(X):: P=72+6*XW :: C
ed. : 000103030301000101010060F0F0 : vampire bats (. : ALL LOCATE(#11,R*8-7,C*8-7):
130 ! if that is still not e : F8F0C0F0F87F7E7878F8DC0C07") : 390 DISPLAY AT(13,1):" bob s : : CALL PATTERN(#11,P+0+0*(X=
nough then move the cursor o : 250 CALL CHAR(68,"060F0F1F0F : tarts with 5 lives butloses : 0)):: Q=-0 :: 60TO 500
ver the line deleting the co : 060F1FFE7E1E1E1F3B30E0000000 : a life each time he istouche : 520 D=SGN(Y)+2 :: ON D 60TO
mputer added spaces.finish t : 00000080C0C0C08000808080") : d by a monster or hitsa bat. : 530,560,540
he typing,press enter,edit t : 260 CALL CHAR(64,"060F0F1F0F : " :: 60SUB 680 : 530 60SUB 850 :: IF R=23 THE
he : 060F1F3F6FCF1F39306000000000 : 400 DISPLAY AT(6,1):" bob ha : N 560 ELSE CALL GCHAR(R+2,C,
140 ! line and with the subs : 0000000080C060301080C020"):: : s a couple of tricks!": : : B):: IF G<>58 THEN 560 ELSE
equent computer added spaces : RANDOMIZE : sounds more interesting!": : R=R+1 :: 60TO 550
you should end up with a 7 : 270 CALL CHAR(76,"0000000000 : : "if he eats a power pill : 540 60SUB 850 :: CALL GCHAR(
line multi statement line. : 00000103060C080103060460F0F0 : =" : "he can swallow one b : R+1,C,B):: IF G<>58 THEN 570
150 ! dont bother typing lin : F8F060F0F8FCF6F3F89C0C06") : at!" : ELSE R=R-1
es 100 - 150 : 280 CALL CHAR(72,"0103030303 : 410 DISPLAY AT(12,1):" somet : 550 P=86 :: CALL PATTERN(#11
160 DISPLAY AT(10,4)BEEP ERA : 01070F1F1B1B0B0706060E80C0C0 : imes after smashing anegg he : ,P+0):: Q=-0 :: CALL LOCATE(
SE ALL:"B A S E M E N T B O : C0C080E0F0F8DB8D0E060607") : becomes a killer! andis abl : #11,R*8-7,C*8-7):: CALL JOYS
B": :: : " A N D T : 290 CALL CHAR(84,"091B1B1B19 : e to eat a monster!." : T(1,X,Y):: 60TO 520
H E": :: : " S C A R Y M O : 1F0F0303070E1C1B1C0C0480C0D8 : 420 DISPLAY AT(16,1):" just : 560 RETURN
N S T E R S" !s.nash 2.7.85 : DC8CF8CF8C0C0C0C0C0C0C0C0C0 : to nourish him as he goes a : 570 R=R-1 :: 60SUB 670 :: R=
170 CALL CHAR(140,"030507070 : 300 CALL CHAR(88,"01031B3B31 : bout his job there area lot : R-1 :: 60SUB 670 :: 60SUB 67
70703030000080C0E0703010EFEF : 3F1F030303030303030390D8D8 : of biscuits to eat #." :: 6 : 0 :: CALL GCHAR(R,C,B):: ON
FE7FEF0F6FEE0783C3E1E3CF8E0" : D89B8F8FC0C0E070381B38302") : OSUB 680 : POS("035861406241",STR$(G),1
,92,"030F1F3F3F1F0F03C0F0F8F : 310 CALL CLEAR :: CALL SCREE : 430 DISPLAY AT(10,1)BEEP:" a : )/2+1 60TO 630,600,540,590,6
CFC8F8F0F") : N(9):: CALL COLOR(1,2,9,2,2, : s the game progresses the : 20,610,580
180 CALL CHAR(58,"FFC0C0C0FF : 9,3,2,9,4,2,9,8,14,9,9,2,9,1 : score values increase." : 580 IF NOT E AND Q THEN J=J+
C0C0C0FF030303FF030303",132, : 0,2,9,11,2,9,12,2,9) : 440 DISPLAY AT(18,1):"use jo : 10*V :: 60SUB 690 :: E=8 ::
"60908C070F1D1F1F0F070204040 : 320 DISPLAY AT(12,5)BEEP:"ha : ystick #1." : "to jump or cl : 60TO 480 ELSE 630
20000060931E0F0B8F878F0E0402 : ve alpha lock off": : :pres : imb up push thestick forward : 590 CALL SOUND(150,440,0)::
0204") : s.....": : :""i"".....f : .to climb down pull the stic : F=1 :: CALL COLOR(#11,6+5*XU)
190 CALL CHAR(40,"1B18181818 : or instructions": : :""ente : k back." :: 60SUB 680 : : : CALL HCHAR(R,C,32):: J=J+
997E3C031F3D7979FFD999E0F8BC : r""...for scary monsters" : : 450 CALL CLEAR :: CALL SCREE : 5*V :: 60TO 630
9E9EFF9B9999FF999999FF9999") : 330 CALL KEY(S,K,S):: IF K=1 : N(2):: CALL COLOR(1,9,15):: : 600 CALL SOUND(100,-5,5):: J
200 CALL CHAR(60,"1C1C3E5D5D : 3 THEN 450 ELSE IF K<>105 TH : CALL CHAR(32,"00F7F7F007F7F : =J+V :: CALL HCHAR(R,C,32)::
1C14360000003C3C00000000007F : EN 330 : 7F") : 60TO 630
7F545444007844C4C444443B") : 340 CALL CLEAR :: DISPLAY AT : 460 60SUB 710 :: CALL VCHAR( : 610 CALL SOUND(500,880,0,440
(3,1):" help our little frie : 2,28,43,3):: CALL VCHAR(2,29 : ,0,220,0,-7,0):: J=J+7*V ::
nd bob to make life a misery : ,43,3):: CALL HCHAR(1,28,41) : Q=1 :: CALL HCHAR(R,C,32,2):
for the monsters that inh : :: CALL HCHAR(1,29,42) : : 60SUB 740 :: M=-1 :: 60SUB
abit the basements!" : : 770 :: 60TO 630

```

★ ★ ★ ★ ★ ★ ★ ★ ★ ★ ★ ★ ★ ★ ★ ★



13

MISSILE DUEL

Extended Basic

```

100 REM MISSILES BY TIM ANDERSON : 320 FOR T=5 TO 27 :: READ HI : 630 DATA 3,3,7,7,1,1,5,1,12, : 840 IF N$(PL)="COMPUTER" THE
RSON : (T):: CALL VCHAR(19-HI(T),T, : 14,10,17,12,15,17,16,14,13,1 : N GOSUB 1550 ELSE 860
110 DIM HI(30),ILN(31) : 96,HI(T)):: NEXT T : 1,11,12,9,7,4 : 850 FOR D=1 TO 150 :: NEXT D
120 RANDOMIZE :: CALL CLEAR : 330 GOTO 720 : 640 DATA 0,2,2,3,1,0,0,0,1,1 : :: DISPLAY AT(23,1):"" :: 6
:: CALL SCREEN(6) : 340 RESTORE 530 :: RETURN : 7,10,11,13,12,7,5,3,1,0,1,0 : QTO 880
130 ON ERROR 1330 : 350 RESTORE 540 :: RETURN : 0,0 : 860 DISPLAY AT(23,1):" LAUNC
140 DEF RD=INT(RND*20)+1 : 360 RESTORE 550 :: RETURN : 650 DATA 0,0,0,3,3,3,0,6,9,1 : H VEL [0 TO 100%] ?" :: ACCE
150 DEF COL=INT(100/N)*2 : 370 RESTORE 560 :: RETURN : 2,12,13,12,16,10,9,6,5,4,3,2 : PT AT(23,26)VALIDATE(DIGIT)S
160 CALL CLEAR :: FOR CH=96 : 380 RESTORE 570 :: RETURN : 1,0,0,0,0 : IZE(-3):VEL :: IF VEL<0 OR V
TO 104 STEP 18 :: CALL CHAR( : 390 RESTORE 580 :: RETURN : 660 DATA 0,0,0,5,8,2,0,6,7,8 : EL>100 THEN 860
CH,RPT$("FF",8)):: NEXT CH : 400 RESTORE 590 :: RETURN : 9,5,5,5,5,12,0,0,0,0,4 : 870 DISPLAY AT(23,1)BEEP:"
170 CALL CHAR(97,"0000000000 : 410 RESTORE 600 :: RETURN : 670 DATA 4,0,0,7,0,10,10,10, : LAUNCH ANGLE [0 TO 90] ?" ::
COCOC018BDBFF7E7EF18C010107 : 420 RESTORE 610 :: RETURN : 10,9,0,0,0,4,5,6,7,0,0,1,0,0 : ACCEPT AT(23,27)VALIDATE(DI
0D1513216080B0E0B0ABC88406") : 430 RESTORE 620 :: RETURN : 680 DATA 1,2,3,4,5,6,7,0,0,1 : 6IT):ANG :: IF ANG<0 OR ANG>
180 CALL CHAR(60,"102040FEFE : 440 RESTORE 630 :: RETURN : 5,0,0,7,6,5,4,3,2,1 : 90 THEN 870
402010",62,"0804027F7F020408 : 450 RESTORE 640 :: RETURN : 690 DATA 2,4,7,9,8,10,2,7,13 : 880 VEL=VEL*0.28 :: ANG=(ANG
"):: BITA$="28100226442020C0 : 460 RESTORE 650 :: RETURN : 1,18,12,11,10,3,3,4,4,5,5,3,1 : *PI/180):: CC=COS(ANG):: RC=
C" :: BITB$="476365002264070 : 470 RESTORE 690 :: RETURN : 1,0,0 : SIN(ANG):: CVEL=INT(DIR*VEL*
0" : 480 RESTORE 700 :: RETURN : 700 DATA 0,0,0,0,6,0,0,0,0,0,0 : CC):: RVEL=INT(-1*VEL*RC)
190 CALL CHAR(101,"183C7EC32 : 490 RESTORE 710 :: RETURN : 0,0,12,16,0,0,0,0,0,0,0,0 : 890 IF N$(PL)<>"COMPUTER" TH
41B1818",40,RPT$("FF",8)):: : 500 RESTORE 660 :: RETURN : 0,0,0,0,0 : EN 910
GOSUB 1280 : 510 RESTORE 670 :: RETURN : 710 DATA 0,0,0,0,0,0,0,0,0,0,0 : 900 FOR D=1 TO 100 :: NEXT D
200 FOR PLAYER=1 TO 2 :: CAL : 520 RESTORE 680 :: RETURN : 0,16,12,0,0,0,0,0,0,0,6,0,0 : :: A=0.7 :: F=0 :: GOTO 930
L CLEAR :: DISPLAY AT(14,2)B : 530 DATA 0,0,0,2,5,5,8,10,1, : 0,0,0,0,0 : 910 DISPLAY AT(23,1)BEEP:" P
EEP:"NAME OF PLAYER "&STR$(P : 16,10,16,11,12,14,10,10,11,1 : 720 REM DISPLAY AND WIND SET : ISPLAY AT(24,1):"
LAYER):: ACCEPT AT(16,4)SIZE : 1,11,6,9,4,2,0 : 730 DISPLAY AT(2,2)SIZE(LEN( : "
(8):N$(PLAYER) : 540 DATA 2,4,9,6,11,11,11,10 : N$(1))BEEP=N$(1):: DISPLAY : 920 CALL KEY(0,K,S):: IF S=0
210 IF LEN(N$(PLAYER))=0 THE : 10,11,11,10,9,8,7,7,13,6,5, : AT(2,29-LEN(N$(2)))SIZE(LEN( : THEN 920 ELSE IF K<>32 THEN
N N$(PLAYER)="COMPUTER" : 5,2,0,0,0,0 : N$(2))BEEP=N$(2):: DISPLAY : 910 :: A=0.7 :: F=0 :: DISP
220 NEXT PLAYER : 550 DATA 3,5,7,9,11,12,14,10 : AT(20,8)SIZE(4):"WIND" : LAY AT(23,1):"YOUR MISSILE I
230 CALL CLEAR :: DISPLAY AT : 15,16,6,6,12,16,15,15,14,14 : 740 CALL SPRITE(13,99,2,142, : S ON IT'S WAY"
(14,2)BEEP:"BEST OF HOW MANY : 13,11,10,0,0,0,0 : 16,12,100,2,142,24,15,99,16, : 930 REM MISSILE FIRE + TRAVE
ISLANDS ?" :: ACCEPT AT(14 : 560 DATA 0,0,0,1,11,13,14,14 : 142,229,14,100,16,142,237) : L
,28)SIZE(-1)VALIDATE(DIGIT) : 15,15,16,12,6,6,16,15,10,14 : 750 CALL HCHAR(4,4,48+HH):: :
NO : 12,11,9,7,5,3 : CALL HCHAR(4,30,48+II) : 940 FOR Q=1 TO 30 STEP 2 ::
240 CALL CLEAR :: CALL SCREE : 570 DATA 1,1,1,2,3,3,4,6,11, : 760 REM MAIN GAME ROUTINE : CALL SOUND(100,Q*110,30-Q)::
N(6):: PL=1 :: DIR=1 :: HH,I : 16,16,16,14,15,13,11,10,9,8, : 770 CALL SPRITE(1,97,1,134, : NEXT Q
I=0 :: QW=0 :: 6,6,4,5,3 : 780 DISPLAY AT(22,11):N$(PL) : 950 CALL SOUND(3500,-7,10)::
250 FOR X=1 TO 31 :: ILN(X)= : 580 DATA 3,4,5,6,6,7,9,10,11 : 790 IF WCH<5 THEN WDIR=60 EL : CALL COLOR(1,2)
INT(RND*16)+1 :: NEXT X : 13,14,5,6,16,6,11,6,4,3,3,2 : SE WDIR=62 : 960 CALL MOTION(1,RVEL,CVEL
260 IN=INT(RND*20)+1 : 1,1,1 : 800 IF WDIR=60 THEN WV=WV*-1 : +WV*8)
270 REM ISLAND SET UP : 590 DATA 1,1,1,2,3,3,5,7,7,8 : 810 IMAGE ##.# M/S : 970 CALL COINC(1,146,24,8,K
280 CALL COLOR(9,13,1,10,5,1 : 11,6,7,8,8,8,8,4,2,3,2,1, : 820 DISPLAY AT(20,13)SIZE(8) : ):: CALL COINC(1,146,237,8,
,2,5,1) : 0 : 830 REM PLAYER SELECT VEL+DI : L):: IF K+L<0 THEN GOTO 1150
290 FOR T=5 TO 27 :: CALL VC : 600 DATA 0,1,2,3,2,4,8,8,8,8 : 840 IF WDIR=60 THEN WV=WV*-1 : 980 CALL CONTACT(DIR,HT,DR,D
HAR(3,T,32,16):: NEXT T :: I : 8,7,6,11,8,7,7,5,3,3,2,1,1, : 810 IMAGE ##.# M/S : C,PL)
N=IN+1 :: IF IN=32 THEN IN=1 : 610 DATA 0,0,0,0,0,0,1,1,2,3 : 820 DISPLAY AT(20,13)SIZE(8) : 990 ON HT GOTO 1000,1050,110
300 CALL HCHAR(19,1,40,96) : 4,6,8,11,12,14,15,8,6,3,2,1 : :USING 810:WVEL :: CALL HCHA : 0,760
310 ON ILN(IN)GOSUB 340,350, : 0,0 : R(20,23,WDIR) : 1000 GOSUB 1360 :: RVEL=RVEL
360,370,380,390,400,410,420, : 620 DATA 0,0,1,2,3,6,8,14,15 : 1010 IF RVEL>-1 THEN 1010
430,440,450,460,470,480,490, : 12,11,8,6,4,3,2,1,1,0,0,1,2 : ELSE 960
500,510,520 : 3,1 : R : 1010 IF RVEL>5 THEN RVEL=5

```

```

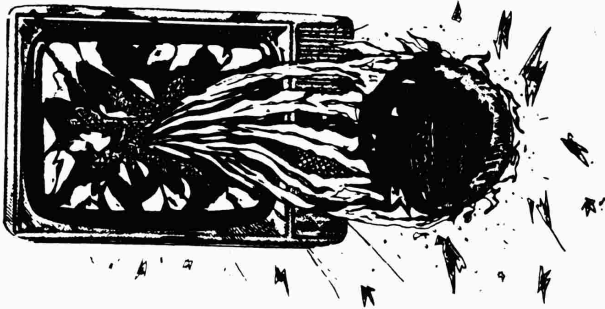
1020 IF F=1 THEN 1030 ELSE C : 1230 DIR=DIR*-1 :: GOTO 280 : 1440 PRINT " THE MISSILE WIL : 1590 !
ALL PATTERN(#1,101):: CALL C : 1240 CALL WINNER(HH,II,N$( ), : L TRAVEL IN THE SAME FASHION : 1600 !
OLOR(#1,16):: CVEL=0 :: F=1 : YY) : AS A THROWBALL, THE WIND H : 1610 RETURN
:: CALL SOUND(-1,988,9) : 1250 ON YY GOTO 1260,230,200 : AVING LITTLEFFECT" :
1030 CVEL=CVEL+WV*A :: IF AB : 1260 CALL CLEAR : 1450 PRINT : " HOWEVER WHEN : 1620 SUB SCORE(HH,II,PL,M,QW
S(CVEL)>10 THEN CVEL=10*SGN : 1270 END : IT REACHES ITSPEAK IT WILL : )
(CVEL)ELSE CVEL=CVEL : : CHANGE FORM. INTHIS FORM THE : 1630 IF QW=0 THEN H(1),H(2)=
1040 GOTO 960 : 1280 REM START : MISSILE IS GREATLY AFFEC : 0 ELSE H(1)=HH AND H(2)=II
: 1290 DISPLAY AT(6,6):" : TED BY THE WIND" : 1640 IF M=PL*2 THEN 1650 ELS
: MISSILE DUEL : 1460 PRINT : " BEWARE IT IS : E 1660
1050 REM HILL HIT : : ***** : : QUITE POSSIBLETO DESTROY YO : 1650 Z=1 :: IF PL=1 THEN PL=
1060 CALL DELSPRITE(#1):: CA : (11,5):"A GAME ONE OR TWO CA : UR OWN BASE! TAKE INTO CON : 2 ELSE PL=1
LL SOUND(250,220,1,-1,0):: C : N PLAY" : SIDERATION THE EFFECT OF THE : 1660 H(PL)=H(PL)+1 :: HH=H(1
ALL SPRITE(#7,98,9,DR,DC):: : 1295 DISPLAY AT(13,10):"COPY : WIND !" : ): II=H(2)
XR=INT((DR+7)/8):: XC=INT((D : RIGHT TIM ANDERSON 1985" : 1470 PRINT : : TAB(5);"PRES : 1670 IF Z=1 THEN 1690
C+7)/8) : 1300 FOR D=1 TO 90 :: NEXT D : S ANY KEY" : 1680 IF PL=1 THEN PL=2 ELSE
1070 FOR X=(XR-1)TO XR :: FO : : DISPLAY AT(21,2)BEEP:"DO : 1480 CALL KEY(0,K,S):: IF S= : PL=1
R Y=XC-1 TO XC+1 :: CALL HCH : YOU WANT INSTRUCTIONS ?N" : 0 THEN 1480 ELSE CALL CLEAR : 1690 Z=0 :: QW=1
AR(X,Y,32):: NEXT Y :: NEXT : : ACCEPT AT(21,28)SIZE(-1)VA : 1490 PRINT " BOTH FORMS OF M : 1700 SUBEND
X : : LIDATE("ynNY"):Y$ : : VE AND WILL EXPLODE IF THEY : 1710 SUB CONTACT(DIR,HT,DR,D
1080 CALL DELSPRITE(#7):: IF : 1310 IF Y$="Y" OR Y$="Y" THE : GO TOO HIGH.THEY ARE ALSO RA : C,PL)
PL=1 THEN PL=2 ELSE PL=1 : N 60SUB 1380 : NGE LIMITED" : 1720 CALL POSITION(#1,DR,DC)
1090 DIR=DIR*-1 :: GOTO 770 : 1320 RETURN : 1500 PRINT : "IF THE MISSIL : : IF DC<=8 OR DC>=252 THEN
: 1330 CALL ERR(W,X,Y,Z) : ES HIT THE ISLAND THEY A : 1760 ELSE IF DR>151 THEN 176
1100 REM SEA HIT : 1340 PRINT W;X;Y;Z : 1350 STOP : RE LIKELY TO EXPLODE, HOWE : 0
1110 CALL DELSPRITE(#1):: CA : 1360 CALL COINC(#1,#2,8,K):: : VER, IT IS POSSIBLE FOR : 1730 IF DR<200 AND DR>158 TH
LL SOUND(-250,-1,0):: CALL S : CALL COINC(#1,#4,8,L):: IF : THEM TO GLANCE OFF SMALL ARE : EN 1760
PRITE(#7,98,9,DR,DC) : K+L<0 THEN 1150 : AS UNAFFECTED" : 1740 CALL GCHAR(INT((DR+7)/8
1120 FOR D=1 TO 50 :: NEXT D : 1370 RETURN : 1510 PRINT : "FINALLY DON'T : ),INT((DC+7)/8),Z):: IF Z=32
1130 CALL DELSPRITE(#7):: IF : 1380 CALL SCREEN(7)::REM INS : FORGET THAT ANANGLE OF ZERO : THEN 1750 ELSE IF Z=40 THEN
PL=1 THEN PL=2 ELSE PL=1 : TRUCTIONS : IS HORIZONTAL AND 90 VERTIC : 1760 ELSE 1770
1140 DIR=DIR*-1 :: GOTO 770 : 1390 PRINT " MISSILE DUEL I : AL": : 1750 HT=1 :: SUBEXIT
: S A GAME FOR ONE OR TWO PLAY : 1520 PRINT : " FOR A DEMONSTR : 1760 HT=3 :: CALL MOTION(#1,
1150 REM HIT HUT : ERS. IF ONLY ONE PLAYER GAME : ATION SIMPLY PRESS ENTER : 0,0):: CALL SOUND(-1,998,15)
1160 CALL MOTION(#1,0,0):: I : IS REQUIRED PRESS ENTER WHE : TO BOTH OF THE NAME P : :: SUBEXIT
F K=-1 THEN M=2 ELSE M=4 : N ASKED FOR PLAYER 2 " : ROMPTS": : 1770 IF Z=96 THEN HT=2 :: CA
1170 CALL POSITION(#1,DR,DC) : 1400 PRINT : " ENSURE THE : 1530 PRINT " PRESS ANY KEY T : LL MOTION(#1,0,0):: CALL SOU
:: CALL DELSPRITE(#1):: C=5 : ALPHA LOCK IS DEPRESSED B : 0 BEGIN" : ND(-1,998,15)ELSE HT=1
1180 FOR N=0 TO 15 :: CL=INT : EFORE ENTERING NAMES." : " : 1540 CALL KEY(0,K,S):: IF S= : 1780 SUBEND
(N/2)+1 :: CALL COLOR(#M,CL, : YOU MAY CHOOSE THE NUMBER : 0 THEN 1540 ELSE RETURN :
#M+1,CL):: CALL SPRITE(#7,98 : OF ISLANDS YOU PLAY UP TO : 1550 DISPLAY AT(23,1):" :: : 1790 SUB WINNER(HH,II,N$( ),Y
,9,DR,DC):: CALL SOUND(100,1 : A MAXIMUM OF 9" : VEL=INT(RND*26)+60 :: ANG=IN : Y)
10,N,-7,N):: CALL DELSPRITE( : 1410 PRINT : " THE WINNER W : T(RND*41)+45 : 1800 CALL DELSPRITE(ALL):: C
#7):: C=C+1 :: NEXT N : ILL BE THE ONE TO HIT THE O : 1560 ANG=INT(RND*49)+35 :: I : ALL CLEAR :: CALL SCREEN(7)
1190 CALL CHAR(94,BITA$,95,B : POSITION BASE OVER HALF TH : F VEL+ANG>132 THEN 1560 : 1810 IF HH>II THEN 1820 ELSE
ITB$):: CALL PATTERN(#M,94, : IS NUMBER. THE DEFAULT IS T : H VEL SELECTED "&STR$(VEL)&" : B=HH :: A=II :: J=2 :: L=1
M+1,95):: CALL SOUND(1000,-7 : HE BEST OF 3" : Z" :: FOR D=1 TO 150 :: NEXT : 1820 A=HH :: B=II :: J=1 ::
,1,110,10):: CALL COLOR(#M,7 : 1420 PRINT :TAB(6);"PRESS AN : D : L=2
,9,DR,DC):: CALL SPRITE(#7,98 : Y KEY" : 1430 CALL KEY(0,K,S):: IF S= : 1580 DISPLAY AT(23,3):"LAUNC : 1830 DISPLAY AT(14,6)BEEP:"T
1200 FOR D=1 TO 260 :: NEXT : 0 THEN 1430 ELSE CALL CLEAR : H ANGLE "&STR$(ANG)&" DEGREE : HAT'S YOUR LOT"
D :: CALL DELSPRITE(#M,#M+1) : : S" : 1840 FOR D=1 TO 350 :: NEXT :
1210 CALL SCORE(HH,II,PL,M,Q : : : : D
W) : : : :
1220 IF HH>=(ND+1)/2 OR II>= : : : :
(ND+1)/2 THEN 1240 ELSE 1230 : : : :

```

```

1850 CALL CLEAR :: DISPLAY A : 1870 FOR D=1 TO 300 :: NEXT : 1890 YY=1 :: SUBEXIT
T(8,4):"CONGRATULATIONS "&N$ : D : 1900 DISPLAY AT(21,1):" TH :
(J): " YOU BEAT "& : 1880 DISPLAY AT(21,2)BEEP:"H : E SAME PLAYERS ?Y" :: ACCEPT :
N$(L) : OW ABOUT ANOTHER GAME ?Y" :: : AT(21,22)VALIDATE("ynNY")SI :
1860 DISPLAY AT(14,6):USING : ACCEPT AT(21,26)VALIDATE("y : ZE(-1):Y$ :: IF Y$="Y" OR Y$ :
" BY # HITS TO #":A,B : nNY")SIZE(-1):R$ :: IF R$="y : ="Y" THEN YY=2 ELSE YY=3 :
: " OR R$="Y" THEN 1900 : 1910 SUBEND :
: : : :

```

HALLEYS COMPETITION FROM PAGE 5

```

310 CALL COLOR(0,16,1):: CAL : 410 CALL CHARPAT(33,A$):: CA
L HCHAR(1,1,32,1000):: PRINT : LL CHAR(34,A$):: SHI=6 :: EN
"CONGRATULATIONS YOU MADE I : E=5 :: DISPLAY AT(5,2)SIZE(1
T TO YOUR SPACESHIP." : : "P : 1):"SHIELD=";SHI :: DISPLAY
LEASE WAIT.....IN TRANSIT." : AT(5,16)SIZE(13):"ENERGY=";E
: : : CALL DELSPRITE(ALL) : NE
320 CALL CHAR(140,"030F1F3F7 : 420 CALL COLOR(0,16,1):: CAL
F7FFFFFFF7F7F3F1F0F03FFFFF : L HCHAR(8,3,128,11):: CALL H
EFEFCFBFC0C0C0F0F8FCFEFEFFFF" : CHAR(16,3,129,11):: CALL VCH
): FOR G=1 TO 500 :: NEXT G : AR(8,2,130,9):: CALL VCHAR(B
: : CALL CLEAR :: PRINT "YOU : ,13,131,9)
R MISSION IS TO SEEK OUT" : 430 DISPLAY AT(3,1):"FRONT V
330 PRINT : "HALLEYS COMET AN : IEW SPACE MAP" :: CALL
D DESTROY IT" : : "BEFORE IT R : VCHAR(1,15,131,19):: FOR T=1
EACHES EARTH." : : "IF YOU FAI : TO 10 :: A=INT(RND*7)+1 ::
L TO DO THIS THE" : : "COMET W : B=INT(RND*9)+1 :: CALL HCHAR
ILL RAM EARTH AND" : : "CAUSE : (A+8,B+3,46):: NEXT T
EARTH TO EXPLODE " : : 440 CALL HCHAR(19,2,128,30):
340 PRINT : "BECAUSE OF THE N : : CALL HCHAR(1,2,129,30):: C
UCLEAR DUMPS" : : "TO FIND THE : ALL VCHAR(2,1,130,17):: CALL
COMET YOU MUST" : : "ENTER TH : VCHAR(2,32,131,17)
E NUMBER OF THE GRID" : : "YOU : 450 CALL COLOR(2,16,1,13,14,
WANT TO GO TO." : : "IN EACH : 1,1,16,1,14,3,1):: CALL CHAR
GRID YOU WILL FIND" : : : (97,"8142240000244281"):: CA
350 PRINT "THE COMET OR BE U : LL SPRITE(#1,97,5,12*8-7,8*8
NDER ATTACK" : : CALL COLOR(1 : -7):: GOTO 480
4,1,0,1,1):: FOR G=1 TO 30 : 460 CALL COLOR(2,1,1):: FOR
00 :: NEXT G :: CALL CHAR(13 : T=1 TO 7 :: FOR H=20 TO 25 :
9,RPT$( "F",16),46,"1"):: CAL : : CALL SPRITE(#H,46,16,91,55
L CLEAR : : ,COS(H)*15,SIN(H)*20):: NEXT
360 CALL HCHAR(12,27,140):: : H :: NEXT T :: CALL COLOR(2
CALL HCHAR(12,29,143):: CALL : ,16,1):: FOR T=20 TO 25
HCHAR(14,27,141):: CALL HCH : 470 CALL DELSPRITE(#T):: NEX
AR(14,29,142):: CALL HCHAR(1 : T T :: RETURN
3,27,139,3) : : 480 CALL CHAR(98,"0000000001
370 CALL VCHAR(12,28,139,3) : : 033F7F"):: CALL SPRITE(#2,98
: A$="EARTH" :: FOR T=1 TO 5 : ,5,83,28*8):: DISPLAY AT(20,
: : CALL HCHAR(16,26+T,ASC(S : 1):"TEXT":"----" :: K=INT(RN
EB$(A$,T,1)):: NEXT T :: FO : D*8)+1 :: COMA=K+8 :: COMB=1
R T=8 TO 16 :: CALL HCHAR(T, : 9
19,33,8):: NEXT T : 490 CALL HCHAR(12,17,ASC("Y"
380 DISPLAY AT(8,16):"012345 : ):: CALL HCHAR(7,22,ASC("X"
678" :: COUNT=0 :: FOR T=8 T : )
0 16 :: DISPLAY AT(T,15)SIZE : 500 CALL CHAR(100,"000000000
(2):COUNT :: COUNT=COUNT+1 : 00000FF"):: CALL SPRITE(#7,1
: NEXT T : 00,16,COMA*8-7,COMB*8-7):: D
390 DISPLAY AT(18,14)SIZE(18 : ISPLAY AT(22,1):"ENTER YOUR
):"COMETS POSITION" : : Y NUMBER(1-8)"
400 CALL CHAR(33,"FFB1818181 : 510 DISPLAY AT(5,16)SIZE(13)
8181FF",128,"FF",129,"000000 : "ENERGY=";ENE :: CALL SPRIT
00000000FF0101010101010101" : E(#16,100,11,129,COMB*8-7)::
: : CALL CHAR(131,"8080808080 : IF COMB>24 THEN CALL COLOR(
808080"):: CALL COLOR(13,1,1 : #16,7)
,2,1,1) : :

```

```

520 CALL MOTION(#1,0,0):: AC
CEPT AT(22,26)SIZE(1)BEEP VA
LIDATE("12345678"):MOVA :: D
ISPLAY AT(22,1):"ENTER YOUR
X NUMBER(1-8)"
530 ACCEPT AT(22,26)BEEP SIZ
E(1)VALIDATE("12345678"):MOV
B :: CALL HCHAR(22,1,32,64):
: CALL COLOR(#2,1):: GOSUB 4
60
540 CALL LOCATE(#2,8*8+(MOVA
*8-7),18*8+(MOVB*8-7)):: CAL
L COLOR(#2,5):: CALL GCHAR(M
OVA+8,MOVB+18,6ET):: CALL HC
HAR(MOVA+8,MOVB+18,34)
550 IF SHI<3 AND ENE>100 THE
N SHI=SHI+1 :: ENE=ENE-100 :
: DISPLAY AT(22,1):"EXTRA SH
IELD NEEDED" :: CALL SOUND(-
4000,-8,0)ELSE 570
560 FOR OP=ENE+100 TO ENE ST
EP -2 :: DISPLAY AT(5,16)SIZ
E(13):"ENERGY=";OP :: NEXT O
P :: DISPLAY AT(5,2)SIZE(9):
"SHIELD=";SHI :: CALL HCHAR(
22,1,32,96)
570 CALL COINC(ALL,HIT):: IF
HIT=-1 THEN DISPLAY AT(22,1
)BEEP:"YOU HAVE FOUND THE CO
MET" ELSE GOTO 710
580 IF ENE<300 THEN DISPLAY
AT(23,1):"BUT YOU DO NOT HAV
E ENOUGH ENERGY TO DESTROY
IT(300) " ELSE CL=B :: CL=CL
*8 :: GOTO 610
590 CALL CHAR(124,"C"):: CAL
L SPRITE(#13,124,7,12*8-7,3*
8-7,0,4)
600 CALL POSITION(#13,TOP,PO
T):: IF POT<13*8-7 THEN 600
ELSE CALL DELSPRITE(#13):: G
OTO 860
610 DISPLAY AT(23,1):"SHOOT
IT BEFORE IT LEAVES THE SC
REEN" :: CALL CHAR(104,"0000
00F0F0",105,"00003C7E7E3C"):
: CALL SPRITE(#12,105,7,56+C
L-7,25,0,4)
620 CALL SPRITE(#13,104,11,5
6+CL-7,24,0,4):: CALL MOVE(E
NE):: CALL COINC(#1,#12,20,Q
): IF Q=-1 THEN CALL KEY(1,
K,S):: IF K=18 THEN 680

```

```

630 CALL POSITION(#12,QW,ER)
:: IF ER>12*8-7 THEN 660 ELS
E CALL MOVE(ENE):: CALL COIN
C(#1,#12,2,Q):: IF Q=-1 THEN
650
640 GOTO 630
650 CALL KEY(1,K,S):: IF K<>
18 THEN 630 ELSE 680
660 CALL MOTION(#1,0,0):: CA
LL DELSPRITE(#13,#12):: DISP
LAY AT(22,1)BEEP:"YOU MISSED
THE COMET TRY TO FIN
D IT AGAIN BEFORE IT HITS EA
RTH " :: FOR G=1 TO 1000 ::
NEXT G :: GOTO 860
670 REM
680 CALL MOTION(#1,0,0,#12,0
,0):: CALL DELSPRITE(#13)::
CALL POSITION(#12,QW,ER):: C
ALL CHAR(112,"10006000800020
01"):: FOR T=10 TO 20 :: CAL
L SOUND(-200,-7,T)
690 CALL SPRITE(#T,112,7,QW,
ER,SIN(T),COS(T)):: NEXT T :
: GOTO 1040
700 REM
710 IF GET=34 THEN DISPLAY A
T(22,1):"YOU HAVE BEEN HERE
BEFORE AND THERE IS NO ALI
ENS LEFT" ELSE 730
720 FOR G=1 TO 1000 :: NEXT
G :: GOTO 860
730 IF GET<>33 THEN 860 :: A
LIEN=INT(RND*14)+1
740 DISPLAY AT(22,1):"SHOOT
THE ALIENS BEFORE THEY HIT YO
U"
750 FOR T=ALIEN TO 1 STEP -1
:: DISPLAY AT(24,1):"ALIENS
LEFT=";T :: ALA=INT(RND*8)+
1 :: ALB=INT(RND*8)+1 :: ALA
=ALA+8 :: ALB=ALB+3 :: CALL
CHAR(36,"0000000008")
760 CALL SPRITE(#6,36,14,ALA
*8-7,ALB*8-7):: GOSUB 850 ::
GOSUB 820 :: CALL CHAR(37,"
0000000001C"):: CALL PATTERN(
#6,37):: GOSUB 850
770 GOSUB 820 :: CALL CHAR(3
6,"000000141C14"):: CALL PAT
TERN(#6,36):: GOSUB 850 :: 6
OSUB 820 :: CALL CHAR(37,"00
0024243C2424")
780 CALL PATTERN(#6,37):: 60
SUB 850 :: GOSUB 820 :: CALL
CHAR(36,"004444546C5444")::
CALL PATTERN(#6,36):: GOSUB
850 :: GOSUB 820
790 CALL CHAR(37,"B1818199E7
99B181"):: CALL PATTERN(#6,3
7):: GOSUB 850 :: GOSUB
820 :: CALL SCREEN(7):: CALL
DELSPRITE(#6)800 CALL SOUND
(-100,-7,0):: CALL SCREEN(2)
:: SHI=SHI-1 :: IF SHI<0 THE
N 1060 ELSE DISPLAY AT(5,2)S
IZE(9):"SHIELD=";SHI

```

MORE FROM JO ANN

ON

TI. WRITER



Find String, Replace String, Move, Copy - why would I replace something I couldn't find in the first place? Actually, sounds like I've spent too much time adventuring - have to give it a break before I really go crazy, although my friends say I've been that way as long as they remember me. Funny since they keep trying to say they'd like to forget me. Should I take that personally?

Our last article just spent a little time on getting into the TI Writer/Word Processor. For some it may have been too much and others not enough. I hope my continuing doesn't really mess anyone else up - anyone as bad off as I am will definitely need help!

Hopefully, we have typed and saved our first document. My girlfriend said she's going crazy trying to follow my directions. I tried to explain to her that's probably true, but she said I have to know what I'm talking about - I have the TI Writer Manual. "Doesn't mean a thing", I replied. I can't read!

Anyway, after having saved a document you find you need to make changes. It's natural (for me anyway). So I certainly had to learn how to correct the changes so I wouldn't look like too complete a fool. Before I save over something I want to keep, I SHOW DIRECTORY to get a list of what's on my save disk. While in the command mode, type (SD) and press (ENTER) to select Show Directory. You can see the contents of your diskette on the screen at any time while using the Text Editor. Anything in the Text Buffer will not be lost so no problems there. You will see (DIS/VAR) for TI Writer files, and you can see space used and available. Anything other than (DIS/VAR) will not be displayed on the Show Directory technique. So, get into Command Mode (FCTN 9), type (SD) for Show Directory, then (ENTER), and it will ask which disk drive you wish to utilize. Type (1) or (2) depending, (ENTER), and there it is. Pressing (ENTER) brings you back to where you left off and you can go on from there.

After loading the file you wish to edit (LF) (ENTER) (DSK?.LETTER), you can use the (FCTN ARROW) keys as one means of going to the point you wish to edit. (FCTN E) for up, (FCTN X) for down, etc. Don't forget to hold down the FUNCTION key or else you will blot out your text. Arrows go left, right, up or down. TAB (FCTN 7) moves to the next tab originally placed, WORD TAB (CTRL 7) moves to the next word, and (CTRL V) moves the cursor to the beginning of the line on which it is positioned. Oops, testing out I almost forgot to hold down the FUNCTION key - case of not listening to what I was saying. Normal again!

Okay, got a problem on Line 0038. You've loaded up your document and FCTN X to get to Line 0038. CTRL 7 to the word you wish to spell correctly. You can either type over it, or if it needs to be made longer, FCTN 2 to insert the space required. Type it correctly. If you used FCTN 2 to insert, CTRL 2 will reposition (reformat) your text to its almost original position. You should

now have a corrected sentence and can go on to get even more confused. If you wish to take out (delete) either a letter or word, (FCTN 1) will do it for you. Place the cursor over the letter you wish to delete, press (FCTN 1) slowly (it auto-repeats) and you can watch one or more letters disappear as you wish. (FCTN 3) will delete an entire sentence saving time.

Back to reformat, (CTRL 2). Suppose you left out a couple of words. Rather than type a whole line over again, why not insert those few words instead? Position the cursor where you want to insert the forgotten words, press (FCTN 2). Notice the space put in. Type the words you want inserted, then press (CTRL 2). The subsequent lines will be brought up and you have a finished project. The Reformat stops wherever a Carriage Return symbol is placed.

Want to check your letter other ways? Scrolling - sideways, up or down, can be accomplished without changing text. Next Window (FCTN 5) moves the cursor to the next window. Remember, we have three windows, 40 columns each. (CTRL T) for back tab, or (FCTN 7) for forward tab. Next Paragraph can be reached via (CTRL 4) and the unforgettable ROLL DOWN (FCTN 4) and ROLL UP (FCTN 6). Hardly ever use them myself. That doesn't mean they are worthless, that just means I have enough on my mind to stay just confused enough.

If you are searching for a word you think you might have spelled incorrectly, scrolling may be helpful but there are easier ways. Find String and Replace String can help in these areas. I just used it to change FCTN from FNCT. I hope you know which is correct as I'm still lost. I know, look it up in my manual! Anyway, position the cursor at the foremost spot you think you might have to start looking. Or position it at the first line of your text to be sure you don't miss anything. Having done that, go Command Mode (FCTN 9). Press FS (FindString) then (ENTER). The computer will help by asking for a response that we hope we know how to answer. Okay, Command Mode, FS, ENTER, you see FIND enter /string/ : . Let's take my example. I had typed FNCT where I should have had FCTN. So I would type a slash (/) then the word(?) FNCT and then another slash (/). Hitting enter will tell the computer to look for my first dumb mistake and I will have the opportunity to change it when the computer pulls it up on screen for me at the exact spot it is located. The cursor will show it up to you as if laughing and saying "I never would have done that ha-ha". Oh well, I change it and will try again if I think there is another mistake. Remember, the computer will be looking for the word exactly as you typed it. And if you want to start at a particular line for searching you can go Command Mode, (S) (ENTER) for Show Line. The line SHOW LINE enter line number will appear. Typing 0001 (ENTER) will put you at line 0001. Command Mode, (FS) (ENTER) and the rest will start your search immediately at line 0001 throughout your text. The first occurrence will be brought up for you.

Replace String is also helpful. With this command, you can replace a character, phrase, or word with another character, phrase or word. Whatever your pleasure. Let's try Replace String. If we had the word occurrence - originally typed in our text as occurence - we can change it to correctly show 2 r's with Replace String. No need to find each word, insert, reformat, etc. Position the cursor at the point in text you wish to start searching. Line 0001 is always good (just in case). So, position your cursor, Command Mode (FCTN 9). Type (RS) (ENTER). You will see the following: REPLACE

enter /old string/new string/ : - Let's change all of our occurrence words to occurrence by doing the following. Type the following: /occurrence/occurrence/ (ENTER). Old string being the word to replace, and new string the word replacing it. The computer will help again by asking via prompt if you want to change all instances, some instances, as follows: Type (Y) to replace the first instance being displayed and find the next instance. Type (N) if you do not want to change the instance being displayed. Type (A) to replace the old string with the new string in all subsequent instances. And last but not always least, Type (S) to stop the command and return to edit mode. If you (S) for stop, the cursor will position itself on the first character of the last string found. Got it? I hope so, I'd hate to go through it again - I'm already lost!

Right now, I'm going to save this article and print it out for proofreading before I forget what I just said that didn't make any sense. No need to leave - I can Command Mode (FCTN 9), (F), (ENTER), (PF) for Print File, (ENTER) and ask the computer to print out the text without having to load up again after it's done. All I do is give it the device name (mine being PIO) and there she goes. Of course, any special commands (underline, doublestrike, etc.), will print out in command form, but I can also verify the correct commands for that. These commands will be discussed later when I recover from my last adventure, if I ever do.

Until then, Happy TI Writing, and for my fellow adventurers, Happy Adventuring! See you soon.....

★ ★ ★ ★ ★ ★ ★ ★ ★ ★ ★ ★ ★ ★ ★ ★

HALLEYS COMPETITION

FROM PAGE 16

```
810 NEXT T :: GOTO 860
820 CALL COINC(1,1,5,0W)::
  IF 0W=-1 THEN 830 ELSE RETU
RN
830 CALL KEY(1,K,S):: IF K<>
18 THEN RETURN ELSE CALL CHA
R(38,"0100200100060008"):: C
ALL PATTERN(16,38):: CALL SO
UND(200,-7,0)
840 CALL DELSPRITE(16):: ALI
EN=ALIEN-1 :: ENE=ENE+7 :: G
OTO 750
850 CALL MOVE(ENE):: GOSUB 8
20 :: CALL ALIN(ALA,ALB):: G
OSUB 820 :: CALL MOVE(ENE)::
  GOSUB 820 :: CALL ALIN(ALA,
ALB):: RETURN
860 CALL HCHAR(22,1,32,96)::
  COMB=COMB+.4 :: IF COMB<27
THEN 500
870 CALL SCREEN(2)
880 CALL CLEAR :: CALL DELSP
RITE(ALL):: CALL CHARSET ::
CALL CHAR(46,"I"):: FOR T=18
TO 28 :: RANDOMIZE :: CALL
PEEK(-31808,K,L):: CALL SPRI
TE(1,T,46,16,K+1,L+1):: NEXT
T
890 CALL CHAR(33,"030F1F3F7F
7FFFFFFF7F7F3F1F0F03FFFFFFF
FEFCBF0C0C0F0F8FCFEFEFFFF",
38,"FFFFFFFFFFFFFFFF")
```

```
900 CALL CHAR(42,"3C7EFFFFF
FF7E3C00030F3F3F0F03"):: CAL
L VCHAR(12,16,38,3):: CALL H
CHAR(13,15,38,3):: CALL VCHA
R(12,15,33)
910 CALL HCHAR(12,17,36):: C
ALL HCHAR(14,15,34):: CALL H
CHAR(14,17,35):: CALL COLOR(
1,3,1,2,3,1)
920 CALL SPRITE(1,42,7,13*8
-7,8,0,4,12,43,11,13*8-7,1,0
,4)
930 CALL POSITION(1,A,B)::
  IF B<15*8-7 THEN 930 ELSE CA
LL DELSPRITE(1,12):: CALL S
OUND(-1000,-7,0):: CALL COLO
R(1,9,1,2,9,1)
940 REM
950 REM
960 REM
970 REM
980 REM
990 REM
1000 CALL MAGNIFY(2):: CALL
CHAR(143,"8001010050002000")
:: FOR T=1 TO 28 :: CALL SPR
ITE(1,T,143,7,13*8-7,17*8-7,C
OS(T)*5,SIN(T)*5):: CALL SOU
ND(-1000,-7,T)
```

```
1010 CALL COLOR(1,1,1,2,1,1)
:: NEXT T :: CALL DELSPRITE(
ALL):: CALL CLEAR :: FOR T=1
TO 14 :: CALL COLOR(T,3,1):
: NEXT T :: PRINT "THE EARTH
HAS BEEN DESTROYED" :: :
1020 PRINT "YOUR MISSION HAS
FAILED" :: :
1030 PRINT "TO PLAY AGAIN RU
N THE - PROGRAM" :: GOTO
1090
1040 CALL DELSPRITE(ALL):: C
ALL CLEAR :: CALL CHARSET ::
PRINT "CONGRATULATIONS YOU"
:"DESTROYED THE COMET AND TH
E": "EARTH IS SAFE" :: :
1050 FOR T=0 TO 14 :: CALL C
OLOR(T,3,1):: NEXT T :: GOTO
1030
1060 CALL CLEAR :: FOR T=1 T
O 28 :: CALL COLOR(1,T,1):: N
EXT T :: CALL COLOR(12,15)::
  CALL MOTION(12,0,-3):: FOR
6=1 TO 250 :: NEXT 6 :: CALL
CHAR(37,"1000600080002001")
1070 CALL POSITION(12,0W,ER)
:: FOR T=1 TO 28 :: CALL SPR
ITE(1,T,37,7,0W,ER,SIN(T)*2,C
OS(T)*2):: CALL SOUND(-200,-
7,T):: NEXT T
1080 CALL CLEAR :: CALL DELS
PRITE(ALL):: FOR T=1 TO 14 :
: CALL COLOR(T,3,1):: NEXT T
:: PRINT "YOU WERE BLOWN T
O BITS" :: : :: GOTO 1030
1090 END
1100 SUB MOVE(ENE)
```

```
1110 CALL JOYST(1,S,D):: CAL
L POSITION(1,W,E):: W=W-D*2
:: E=E+S*2 :: S=W :: D=E ::
  CALL LOCATE(1,S,D):: IF S<
8*8-7 THEN CALL LOCATE(1,8*
8-7,D)
1120 IF S>16*8-7 THEN CALL L
OCATE(1,16*8-7,D)ELSE IF D<
3*8-7 THEN CALL LOCATE(1,S,
3*8-7)ELSE IF D>12*8-7 THEN
CALL LOCATE(1,S,12*8-7)
1130 CALL KEY(1,K,S):: IF K=
18 THEN CALL COLOR(1,11)::
  CALL SOUND(-2,-1,0):: CALL C
OLOR(1,5):: ENE=ENE-1
1140 SUBEND

1150 SUB ALIN(ALA,ALB)
1160 A=INT(RND*6)+1 :: IF A>
6 THEN 1160 ELSE ON A GOTO 1
170,1180,1190,1180,1190,1170
1170 ALA=ALA-1 :: GOTO 1200
1180 ALA=ALA+1 :: GOTO 1200
1190 ALA=ALA
1200 ALA=ALA-(ALA<8)+(ALA>16
)
1210 A=INT(RND*6)+1 :: IF A>
6 THEN 1210 ELSE ON A GOTO 1
220,1240,1230,1220,1230,1240
1220 ALB=ALB-1 :: GOTO 1250
1230 ALB=ALB+1 :: GOTO 1250
1240 ALB=ALB
1250 REM
1260 ALB=ALB-(ALB<4)+(ALB>12
):: CALL LOCATE(16,ALA*8-7,A
LB*8-7):: SUBEND
```

HELP

ON ADVENTURES

RETURN TO PIRATE'S ISLAND



A trip to the cavern,
might otherwise be bright,
Unfortunately no provision,
was made to have light,
See Pirate's Island Adv# 2,
a shed could be your clue,
Directions and your instinct,
would very well be due.



Going through the crawlway,
with glasses or a treasure,
Might under other circumstances,
be considered a pleasure,
A blind man has the instincts,
even without his sight,
Even with a squint he might try
another route.



What would the technical name be
for having a fear of sharks,
Fortunately, in this adventure,
the fear would be a lark.
Most people love the water, or
so I am told,
I only suggest before you go,
take a breath and hold!



GOLDEN VOYAGE



Humpty Dumpty sat on a wall,
Humpty Dumpty had a great fall,
Besides being broken,
For not just a token,
Remains needed to be cleared away.



Walking on rocks without any socks,
Would be quite a feat!
I have only 1 suggestion - try
protecting your feet!



STRANGE ODYSSEY



Strange Odyssey causing you a trip? It's not bad,
I've had worse.

Sometimes playing adventures could be considered
a curse.

This one's pretty easy, pushing and pulling all the
while.

Push and then pull, but before you go, smile.
You'll enjoy the ride.



Examine the instrument before you destroy, reset
the instrument - it's not just a toy.
Take the dog for a walk? Why not, can't be too bad.
Otherwise he just might get really mad.

GHOST TOWN

BY
SCOTT
COPELAND

The old cowboy and indian movies
always were the best.
The meeting of the cavalry and indian
leader were the test.
When the cavalry leader said "Hello",
the Indian Chief, while raising his
arm with the sign of peace,
replied....



A vain person would look in it,
An ugly person might cover it,
An angry person would do even worse,
7 years bad luck as a result of it.

The key to my heart, the entrance
to a part,
Unlock to find the way,
to help you through the day.

■ N, S, E, W, I'm lost, can't find my way.
A compass sure would help a bit, or this
adventure I just might quit.

Direction is not the only value it will hold,
What other value does a compass unfold?

★ SAVAGE ISLAND PART ONE ★

When going to the beach, dig a sand
castle with your hands,
Examine what you've done, you'll be
pleased with how it ends.
Swim up, swim down, then swim all around.
4 directions you can go, eventually you'll
go aground.



★ SAVAGE ISLAND PART TWO ★

A swimmer holds his breath, soon to let it out.
Storing air is helpful, you cannot do without.
Just remember, before you faint, to calm it
down - have some restraint.

A rose by any other name would smell as sweet.
Perhaps you should try to give yourself a treat.
A flower by any other name can not be beat.
If smelling doesn't work, why not try to eat?

BRIGHT SPARKS

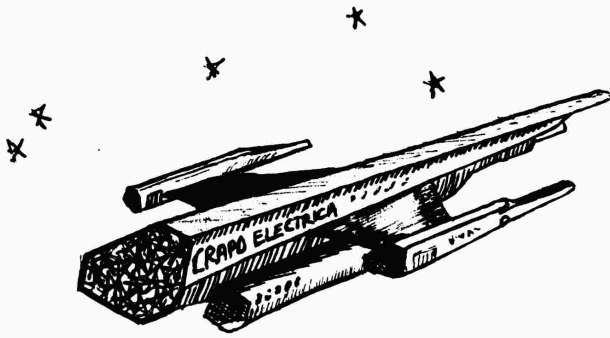


Working out the correct values of frequency, duration and volume for sound seems hard enough in basic, but if any of you have read the chapter on sound in the EDITOR/ASSEMBLER MANUAL will realise how good the basic method really is. In this tutorial I will hope to show you how to work out a sound table (using the short extended basic program) and how to incorporate them into your own assembly language programs.

Like graphics you must have a table in the VDP ram which contains the relevent values, but unlike graphics there is only one table which contains all the information about a particular sound. However it is not as simple as just stating the duration, frequency and volume as I will explain.

The first byte of a sound list as they are called must contain the number of bytes in the list excluding the duration. Each sound list must contain the frequency bytes or the volume byte or both for 1 to 3 of the sound channels and lastly the volume byte. That is a bit of a mouthful so here is a table showing how a typical sound list is made up.

BYTE	BITS	CONTENTS
1	0-7	THE NUMBER OF BYTES IN THE SOUND LIST EXCLUDING THE DURATION
2	0	FREQUENCY BYTE 1
	1-3	000,010 OR 100 DEPENDING IF YOU ARE USING CHANNEL 1,2 OR 3
	4-7	4 LSB OF THE FREQUENCY CODE
3	0-1	THE SECOND FREQUENCY BYTE
	2-7	00 6 MSB OF THE FREQUENCY CODE
4	0	BYTE FOR THE VOLUME
	1-3	001,011 OR 101 DEPENDING IF YOU ARE USING VOICE 1,2 OR 3
	4-7	THE VALUE FOR THE VOLUME
		0000 LOUDEST TO 1111 NO SOUND
5	0-7	DURATION >00 NO DURATION TO >FF WHICH IS JUST OVER 4 SECS.



There are one or two interesting things about the table. maybe you noticed that there are only 10 bits to specify the frequency, this means that the computer can only play a maximum of 1024(2¹⁰) different tones since the maximum different binary combinations is 1024. The actual value you use for the sound lists is not the actual frequency but the frequency code, which is calculated by the formula:-

$\text{FREQUENCY CODE} = 111860.8 / \text{FREQUENCY}$

The other is that there are only 4 bits for the volume giving 15 different values whereas the basic manual would have you believe there are 30 different values.

It does not really matter if you have not understood all or any of this since I have written a small program which will calculate the actual assembler values for frequency, volume and duration from the values you would use in basic. (Allowing you to pinch other peoples sound effects from magazine listings.)

I think it would be best to show you how to use this program by using a few examples.

1. To convert the values of CALL SOUND(500,110,0) to a sound list.

Firstly run the program. When the channel prompt appears on the screen type "1" and press ENTER. When the duration prompt appears

type "500" and press ENTER. Type "110" for frequency and then "0" for volume. After a brief pause the hexadecimal values for the sound should appear at the bottom of the screen. This is not quite the finished sound list since the first byte is not printed. For this sound list it is 3, since there are 2 bytes for the frequency and 1 for the volume, the duration byte is never included. So the completed sound list would be:-

```
NO.   FREQ  VOL  DUR
>03,>89,>3F,>91,>1E
```

To stop the above sound list you need another one, this can be to play another sound or to stop the sound altogether. To stop the sound you must set the duration to 0 and the volume to 32 (on the channel you want to stop, in this case 1). You should use the convertor program once again to find out the assembler values, but this time when the frequency prompt appears just press ENTER. The values appearing on the bottom of the screen should be >9F,>00 to give the sound list list:-

```
>01,>9F,>00
```

2 If you want to play a chord say the equivalent to CALL SOUND(500,110,0,220,5)

Firstly you would do the same as above to get the values for (500,10,0). Now you must find the values for (500,220,5) making sure you specified sound channel 2. The number of frequency and volume bytes is 6 so the first byte in the sound list would be 6. Below is the completed sound list.

```
NO.   FREQ1 VOL1 FREQ2 VOL2 DUR
>06 >89 >3F >91 >AC >1F >B2 >1E
```

This time when you finish you must stop both sound channels by giving them both a volume of 32

and a duration of 0. Here is the sound list that will do this:-

```
NO. VOL1 VOL2 DUR
>02 >9F >BF >00
```



Hopefully you will now be able to generate your own sound tables but obviously this is not much good without a program to execute them. Below is a tone (110 Hz) then program/subprogram which will play firstly a a chord (110Hz 220Hz) then stop the sound. The same program can be made to play any sound list with a few modifications as you will find out later.

EDITOR/ASSEMBLER VERSION

```
DEF SOUND1 *
REF VMBW * SECTION1
SNDBUF EQU >1000 *
H01 BYTE >01 *
EVEN *

SOUND1 *
LI R0,SNDBUF *
LI R1,SNDTBL * SECTION2
LI R2,17 *
BLWP @VMBW *
LIMI 0 *
LI R10,BUFFER *
MOV R10,@>83CC * SECTION3
SOCB @H01,@>83FD *
MOVB @H01,@>83CE *
LIMI 2 *
LOOP MOVB @>83CE,@>83CE *
JNE LOOP * SECTION4
LIMI 0 *
RT *

SNDTBL BYTE >03,>89,>3F,>91,>1E *SECTION5
BYTE >06,>89,>3F,>91,>AC,>1F,>B2,>1E
BYTE >02,>9F,>BF,>00
END
```

MINI-MEMORY

```
LI R0,>1000 *
LI R1,ST * SECTION2
LI R2,17 *
BLWP @>6028 *
LIMI 0 *
LI R10,>1000 * SECTION3
MOV R10,>83CC *
LI R10,>0100 *
SOCB R10,@>83FD *
MOVB R10,@>83CE *
LIMI 2 *
LP MOVB @>83CE,@>83CE *
JNE LP * SECTION
LIMI 0 *
B *R11 *
ST DATA >0389,>3F91,>1E06 *SEC5
DATA >893F,>91AC,>1FB2 *
DATA >1E02,>9FBF,>0000 *
END
```

I will firstly deal with the editor/assembler listing. Firstly you should note that the parts of the program beginning with the asterisks are remarks and need not be typed in, secondly it should be assembled with the R option of the assembler. As the remarks show I have divided the program into several sections, I shall now discuss these sections in some detail.

Graham Marshall.

BRIGHT SPARKS



SECTION 1: The first thing it does is to specify the programs name using the DEF command, secondly tells the computer that the program will be using the VMBW routine. It also sets the word at SNDBUF to >1000 as this is where the sound table is going to be in the VDP ram.

SECTION 2: This is the routine which places the sound lists into the sound buffer starting at >1000 in the VDP ram. There is one thing that you must change here if you are using your own sound lists, R2 contains the total number of bytes in the sound table (which is contained in section 5 of the program). In this case it is 17 since in the three sound lists there are total of 17 bytes.

SECTION 3: You don't need to bother much about this bit as it will be the same no matter what sound lists you are wanting to play. This is the part of the program which does the sound processing.

SECTION 4: In this program it just waits until the sound list is finished and uses the RT command to return to the editor/assembler menu. Although if you are careful you can have the computer drawing a picture, or anything else for that matter while playing the sound table. To do this although if you are writing to VDP ram then you must dissable the interrupts, by using LIM1 0, or else the computer will crash. You must also enable the interrupts several times a second, using LIM1 2 to allow the sound to be played. The best idea is to do this in the main loop of your program.

LIM1 2
LIM1 0

If you do not want to do this you can just let the sound list finnish before your program continues, only replacing RT by the next line of your program.

SECTION 5: This is the section

where the sound lists are stored. The editor assemblers BYTE directive makes things quite easy since each sound list may be stored on a seperate line. I have used the example that I showd you earlier on so you could see how the sound lists are used in a program.

Instead of repeating all this for the mini memory version I will merely state the differences between the two programs. The first difference is that there is no SECTION 1 since the prgram at present has no name, instead is run by the easy bug command E7D00, and to save memory the values of the buffers, etc. are loaded directly into regiseters rather than using labels. In SECTION2 VMBW is replaced by >6028 since this is the address of the VMBW routine in the mini memory. In SECTION5 RT is replaced by B *R11 since the line by line assembler does not have the RT instruction, they both do the same thing anyway. The biggest difference is that the mini memory does not have the BYTE directive so you must use the DATA directive instead. This is not as handy since each new sound list cannot always begin on a new line, making it slightly more difficult to edit a sound list. One advantage of using the mini memory is that it is easy to change the sound list by using easy bug without retyping the whole program.

To do this is quite simple, once you have checked that the program has worked and saved it on tape you can start. To change the sound table enter the easybug command M7D3A. You are now free to enter a new sound table a byte at a time, or just to change the current one. Once you have done this you press "." then type M7DOA and type the total number of bytes in the new sound table(this value must be in hexidecimal since easybug does not handle decimal noumbers). I have already over-run my usual space so I must stop here, but I hope this article helps you conquer the TMS9919 sound processor.

REVIEW
REVIEW
REVIEW
REVIEW
REVIEW
REVIEW
REVIEW
REVIEW
REVIEW
REVIEW
REVIEW
REVIEW



REVIEW
REVIEW
REVIEW
REVIEW
REVIEW
REVIEW
REVIEW
REVIEW
REVIEW
REVIEW
REVIEW
REVIEW

STRATEGIES

1. Fly low. Good DEFENDER players seldom fly more than two inches above the planet, in order to protect their humanoids.
2. Mutants are not as hard to destroy as it might appear. Thrust, then reverse direction and adjust your height simultaneously, and fire.
3. Listen to the game sounds. Most of the time you'll hear a humanoid cry for help before you see it. Use the Scanner to find it. Try playing DEFENDER using the Scanner alone a few times to get a good idea of how it works.
4. You can carry a humanoid around with you so that the aliens can never destroy all ten of them without destroying you. But if you touch the planet, the humanoid will immediately jump free.
5. Use Smart Bombs only in extreme emergencies, or if there are enough aliens on the screen to push you past a 10,000 point mark.
6. When you're thrusting, move up and down and fire continuously to sweep your area clean.
7. Aliens become confused if you quickly reverse twice, giving you time to blast them.

THE SCANNER

Your view-screen shows your immediate area, but the Scanner at the top of the screen shows a much wider view. The area covered by your view-screen is bracketed in the center of the Scanner, so you can see what's happening to your right and left—a crucial factor with so many aliens around. By using your Scanner you can tell which kinds of aliens are where, and—more important—which of your humanoids are in danger. Humanoids and the different types of alien ships are distinguished by different colored blips.

ALIEN SHIPS

Each of the alien ships has a different function.

LANDERS, the first to appear, kidnap humanoids and fire white charges.

BOMBERS lay mines to trap you—you can't shoot mines, so you must avoid them.

BAITERS appear if you take too long to finish off a wave—they move faster than Defender and fire white charges as they home in on him.

MUTANTS—transformed humanoids—are very dangerous. Once a humanoid is lost to you, spare no effort in trying to destroy it. They'll fly directly above or below you (where you can't hit them) and then charge.

The **POD** is the most dangerous, because when it's destroyed it releases Swarms.

SWARMERS track you closely, so zap them as quickly as you can.

All aliens except Landers can exit at the bottom of the screen and emerge from the top, or vice versa.

RESCUING HUMANOIDS

You begin the game with ten humanoids. When a humanoid is kidnapped, it cries out for help. The Scanner is the fastest way to spot its position.

There are two ways of rescuing a humanoid: either destroy the Lander and let the humanoid fall back to the planet; or destroy the Lander, catch the falling humanoid, and carry it back to the planet. Letting the humanoid fall only works if it's low enough to fall safely. If the humanoid falls safely, you score 250 points, plus 150 points for destroying the Lander. If the humanoid dies, you score only the 150 points for the Lander.

Catching a humanoid in midair is more difficult and earns you more points. To catch a humanoid, shoot the Lander and maneuver Defender to touch the humanoid, then carry it to the planet. This earns you 1000 points, plus 150 for the Lander. If you catch the humanoid but don't return it safely, you still earn 500 points, plus 150 for the Lander.

If the Lander succeeds in carrying a humanoid to the top of the screen, the humanoid becomes a Mutant and returns to attack Defender. When all of the humanoids become mutants, or are destroyed, the planet explodes and vanishes. You continue to play and score points. With every fifth wave, a destroyed planet reappears and all ten humanoids are replaced.

THE RICHARD TWYNING

UTILITIES

NO.1

THIS LITTLE LOT WILL TURN THE SCREEN UPSIDE DOWN
IF YOU CAN THINK OF A USE FOR IT PLEASE LET US KNOW
AS WE ARE STRUGGLING !!!

```

10 DIM SCR$(768) : 200 CALL CHAR(33,"0008080808 : 300 CALL CHAR(73,"001C080808 : 400 CALL CHAR(113,"0000001C2
20 PRINT "CHARACTER REVERSAL : 0800080014141400000000001414 : 08081C0020202020221C002212 : 22A122C0000001E221E122200000
PROGRAM":"": by": 3E143E1414001C2A0A1C2B2A1C") : 0A060A12220002020202023E") : 03C021C201E0000003E08080808"
R.TWYNING (C) 1985" : :: CALL WAIT : :: CALL WAIT : ):: CALL WAIT
30 PRINT "":"SHOULD BE A GOO : 210 CALL CHAR(37,"0006261008 : 310 CALL CHAR(77,"0022362A2A : 410 CALL CHAR(117,"000000222
D AUSTRALIAN PROGRAM." : 04323000040A0A042A122C001010 : 222222002226262A323222003E22 : 222221C000000222214140800000
40 B$="BOTH" ! THIS STRING C : 08000000000100B0404040B10") : 22222223E001E22221E020202") : 022222A2A140000002214081422"
AN BE ALTERED TO UPD OR upd : :: CALL WAIT : :: CALL WAIT : ):: CALL WAIT
TO JUST TURN THE CHARACTERS : 220 CALL CHAR(41,"0004081010 : 320 CALL CHAR(81,"001C222222 : 420 CALL CHAR(121,"000000221
UPSIDE DOWN OR IT CAN BE AL : 100B040000140B3E081400000008 : 2A122C001E22221E0A1222001C22 : 4080B080000003E100B043E00180
TERED TO BTF OR btf JUST TO : 083E08080000000000000C0804") : 021C20221C003E080808080808") : 404020404180008080800080808"
: :: CALL WAIT : :: CALL WAIT : ):: CALL WAIT
50 ! TURN THEM BACK TO FRONT : 230 CALL CHAR(45,"0000003E00 : 330 CALL CHAR(85,"0022222222 : 430 CALL CHAR(125,"000C10102
60 IF B$="UPD" THEN CALL UPD : 0000000000000000000C0C000020 : 22221C0022222214140808002222 : 010100C0000042A10000000")::
70 IF B$="upd" THEN CALL UPD : 100B040200001C2222222221C") : 222A2A2A140022221408142222") : CALL WAIT
80 IF B$="BOTH" THEN 120 : :: CALL WAIT : :: CALL WAIT : 440 SUBEND
90 IF B$="BTF" THEN CALL BTF : 240 CALL CHAR(49,"00080C0808 : 340 CALL CHAR(89,"0022221408 :
100 IF B$="btf" THEN CALL BT : 08081C001C22201008043E001C22 : 080808003E20100804023E001C04 : 450 SUB UPD
F : 201820221C00101814123E1010") : 040404041C0000020408102000") : 460 FOR D=32 TO 126
110 IF B$="both" THEN 120 : :: CALL WAIT : :: CALL WAIT : 470 CALL CHARPAT(D,W$)
120 CALL WAIT :: CALL BTF :: 250 CALL CHAR(53,"003E021E20 : 350 CALL CHAR(93,"0038202020 : 480 B$=W$ :: C$=B$ :: W$=""
CALL UPD : 20221C001804021E22221C003E20 : 202038000000814220000000000000 : 490 L=LEN(B$)
130 A=0 :: FOR R=1 TO 24 :: 100B040404001C22221C22221C") : 000000003E0000040810000000") : 500 DIM A$(100)
FOR C=1 TO 32 :: A=A+1 :: CA : :: CALL WAIT : :: CALL WAIT : 510 FOR A=1 TO L
LL GCHAR(R,C,X):: SCR$(A)=CH : 260 CALL CHAR(57,"001C22223C : 360 CALL CHAR(97,"0000001C22 : 520 A$(A)=SEG$(C$,A,1)
R$(X):: NEXT C :: NEXT R : 20100C00000C0C000C0C0000000C : 3E22220000001E241C241E000000 : 530 NEXT A
140 A=769 :: FOR R=1 TO 24 : 0C000C080400100B0402040B10") : 3C0202023C0000001E2424241E") : 540 FOR A=L TO 1 STEP -1
: FOR C=1 TO 32 :: A=A-1 :: : :: CALL WAIT : :: CALL WAIT : 550 W$=W$&A$(A)
CALL HCHAR(R,C,ASC(SCR$(A))) : 270 CALL CHAR(61,"0000003E00 : 370 CALL CHAR(101,"0000003E0 : 560 NEXT A
: NEXT C :: NEXT R :: FOR D : 3E00000004081020100B04001C22 : 21E023E0000003E021E020200000 : 570 W$=""
=1 TO 10000 :: NEXT D : 20100B00008001C223A2A3A021C") : 03C023A223C00000022223E2222" : 580 W$=W$&A$(15)&A$(16)&A$(1
: :: CALL WAIT : ):: CALL WAIT : 3)&A$(14)&A$(11)&A$(12)&A$(9
150 SUB WAIT : 280 CALL CHAR(65,"001C22223E : 380 CALL CHAR(105,"0000001C0 : )&A$(10)&A$(7)&A$(8)&A$(5)&A
160 FOR D=1 TO 500 : 222222001E24241C24241E001C22 : 80B081C000000101010120C00000 : $(6)&A$(3)&A$(4)&A$(1)&A$(2)
170 NEXT D : 020202221C001E24242424241E") : 024140C1424000000020202023E" : 590 CALL CHAR(D,W$)
180 SUBEND : :: CALL WAIT : ):: CALL WAIT : 600 NEXT D
: 290 CALL CHAR(69,"003E02021E : 390 CALL CHAR(109,"000000223 : 610 SUBEND
: 02023E003E02021E020202003C02 : 62A222200000022262A322200000 :
: 023A22221C00222223E222222") : 03E2222223E0000001E2221E0202" :
: :: CALL WAIT : ):: CALL WAIT :

```


NO.2

THE SECOND OF RICHARDS PROGRAMMES
IS AN ON SCREEN CALCULATOR WHICH
USES BOTH EXTENDED BASIC & JOYSTICKS
DID YOU KNOW THAT THE FIRST HAND HELD
CALCULATOR WAS INVENTED BY TEXAS INSTRUMENTS
IN 1967! OH! THE PLANE AT THE BOTTOM OF THE
PAGE....RICHARD STRIKES AGAIN ON GRAPHX.

```

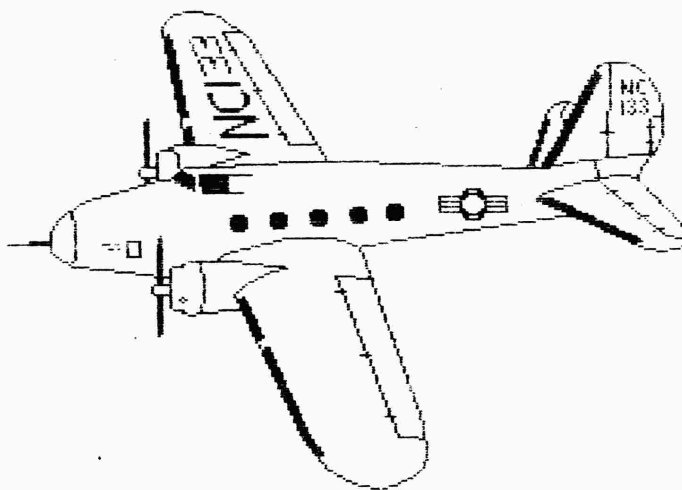
1 REM ***** : 30 PRINT " Z - CALCULATES V : 53 CALL SPRITE(#13,49,2,47,2 : 77 CALL COINC(#1,#10,5,H)::
2 REM $ : ALUE OF FIRST NUMBE : 11,#14,50,2,48,222,#15,51,2, : IF H=-1 THEN A$=A$+"5"
3 REM $ CALCULATOR $ : R AS A PERCENTAGE O : 46,237,#16,85,2,63,209) : 78 CALL COINC(#1,#11,5,H)::
4 REM $ : F SECOND NUMBER" : 54 CALL SPRITE(#17,48,2,62,2 : IF H=-1 THEN A$=A$+"6"
5 REM $ BY $ : :"" : 23,#18,46,2,62,237,#19,61,2, : 79 CALL COINC(#1,#12,5,H)::
6 REM $ : 31 PRINT " S - SAVES CURREN : 77,209,#20,43,2,76,223) : IF H=-1 THEN DISPLAY AT(20,1
7 REM $ RICHARD TWYNING. $ : T DISPLAY VALUE IN ME : 55 CALL SPRITE(#21,45,2,77,2 : )BEEP=" MEMORY=";M$
8 REM $ : MORY": "" : 36,#22,42,2,93,208,#23,47,2, : 80 CALL COINC(#1,#13,5,H)::
9 REM $ (C) 1985 $ : 32 PRINT " C - CLEARS DISPL : 92,222):: CALL CLEAR : IF H=-1 THEN A$=A$+"1"
10 REM $ : AY": "" : 56 REM : 81 CALL COINC(#1,#14,5,H)::
11 REM $ COPYRIGHT $ : 33 PRINT " D - DISPLAYS VAL : : IF H=-1 THEN A$=A$+"2"
12 REM $ : UE THAT IS CURRENTL : 57 REM ** MAIN LOOP ** : 82 CALL COINC(#1,#15,5,H)::
13 REM ***** : Y STORED IN MEMORY" : 58 REM : IF H=-1 THEN A$=A$+"3"
14 REM : :"" : 59 DISPLAY AT(8,4):OP$ :: CA : 83 CALL COINC(#1,#16,5,H)::
15 REM : 34 PRINT " U - USE VALUE CU : LL JOYST(1,X,Y):: CALL MOTIO : IF H=-1 THEN 166
: RRENTLY STORED IN M : N(#1,-Y#1.5,X#1.5):: CALL KE : 84 CALL COINC(#1,#17,5,H)::
16 REM DISPLAY INSTRUCTIONS : EMORY IN CALCULA : Y(1,K,S):: IF K=18 THEN CALL : IF H=-1 THEN A$=A$+"0"
17 REM : TION " :"" : SOUND(100,900,0):: GOTO 69 : 85 CALL COINC(#1,#18,5,H)::
18 REM : 35 FOR D=1 TO 1000 :: NEXT D : 60 IF A$="" THEN DISPLAY AT( : IF H=-1 THEN 184
19 CALL CLEAR :: DISPLAY AT( : :: DISPLAY AT(24,1):" PR : 10,2):B$ ELSE DISPLAY AT(10, : 86 CALL COINC(#1,#19,5,H)::
6,6):" CALCULATOR " : ESS ANY KEY" : 2):A$ : IF H=-1 THEN 95
20 DISPLAY AT(8,6):" : 36 CALL KEY(0,K,S):: CALL KE : 61 IF B$="" AND A$="" THEN D : 87 CALL COINC(#1,#20,5,H)::
BY " :"" R.TWYNING. : Y(1,K,S2):: IF S=0 AND S2=0 : ISPLAY AT(10,2):0 : IF H=-1 THEN IF OP=1 THEN OP
" : THEN 36 : 62 IF A$="" AND B$="" THEN D : $="+ ELSE GOSUB 149 :: OP$=
21 FOR D=1 TO 1000 :: NEXT D : 37 REM : ISPLAY AT(10,2):0 : "+"
22 INPUT " INSTRUCTIONS (Y/N : 38 REM : 63 GOTO 59 : 88 CALL COINC(#1,#21,5,H)::
)?":A$ :: IF A$="N" OR A$="n : 39 REM : 64 REM : IF H=-1 THEN IF OP=1 THEN OP
" THEN 42 : 40 REM : : $="- ELSE GOSUB 149 :: OP$=
23 CALL CLEAR :: PRINT " : 41 REM : 65 REM ** END OF MAIN LOOP $ : "-"
CALCULATOR" : : 89 CALL COINC(#1,#22,5,H)::
24 PRINT " ----- : 42 REM ** DEFINE VARIABLES : 66 REM : IF H=-1 THEN IF OP=1 THEN OP
--" :: PRINT " : THIS PROGRA : ** : : $="$ ELSE GOSUB 149 :: OP$=
M CAN BE USED AS A NORMAL CA : 43 OP$="" :: A$="" :: B$="" : 67 REM CHECK WHICH SPRITES A : "$"
LCULATOR BY USING THE JOYSTI : :: OP=0 :: M$="" : RE TOUCHING,AND DECIDE WHAT : 90 CALL COINC(#1,#23,5,H)::
CK TO MOVE A CURSOR ON T : 44 REM : TO DO : IF H=-1 THEN IF OP=1 THEN OP
HE SCREEN.:" : 45 REM : 68 REM : $="/ ELSE GOSUB 149 :: OP$=
25 PRINT "PRESS THE FIRE BUT : 46 REM : 69 CALL COINC(#1,#2,5,H):: I : "/"
TON TO SELECT THE NUMBERS : 47 REM : F H=-1 THEN 106 : 91 GOTO 59
OR MATHEMATICAL FUNCT : 48 REM : 70 CALL COINC(#1,#3,5,H):: I : 92 REM
IONS WHEN THE CURSOR IS COVE : 48 REM MAGNIFY & DISPLAY SPR : F H=-1 THEN B$=A$ :: OP$="Z" :
RING THEM" :: PRINT " : ITES, DEFINE GRAPHICS FOR CU : : A$="" : 93 REM CHECK WETHER USER
26 PRINT " PRESS ANY KEY" : RSOR AND CLEAR SCREEN : 71 CALL COINC(#1,#4,5,H):: I : HAS ENTERED %,+, -, / OR $,AND
27 CALL KEY(0,K,S):: CALL KE : 49 REM : F H=-1 THEN 176 : GOSUB TO CORRECT SUB-ROUTIN
Y(1,K,S2):: IF S=0 AND S2=0 : 50 CALL MAGNIFY(1):: CALL SP : 72 CALL COINC(#1,#5,5,H):: I : E
THEN 27 : RITE(#1,96,16,194,176,#2,69, : F H=-1 THEN A$=A$+"7" : 94 REM
28 PRINT " :"" :"" :: CALL : 2,1,210,#3,37,2,1,224,#4,83, : 73 CALL COINC(#1,#6,5,H):: I : 95 IF A$="" AND OP$="" THEN
CLEAR :: PRINT " LIST O : 2,1,237) : F H=-1 THEN A$=A$+"8" : 59
F COMMANDS " : ----- : 51 CALL SPRITE(#5,55,2,17,20 : 74 CALL COINC(#1,#7,5,H):: I : 96 IF OP$="Z" THEN GOSUB 113
-----": "" : 9,#6,56,2,17,223,#7,57,2,17, : F H=-1 THEN A$=A$+"9" : 97 IF OP$="+" THEN GOSUB 120
29 PRINT " E - ENDS PROGRAM : 238,#8,67,2,92,236):: CALL C : 75 CALL COINC(#1,#8,5,H):: I : 98 IF OP$="-" THEN GOSUB 127
": "" : HAR(96,"8060783C38140201") : F H=-1 THEN GOSUB 156 : 99 IF OP$="/" THEN GOSUB 134
: 52 CALL SPRITE(#9,52,2,32,21 : 76 CALL COINC(#1,#9,5,H):: I : 100 IF OP$="$" THEN GOSUB 14
: 0,#10,53,2,32,222,#11,54,2,3 : F H=-1 THEN A$=A$+"4" : 1
: 1,237,#12,68,2,108,207) : : 101 B$="" :: OP$=""

```

```

102 GOTO 59          : 130 REM          : 156 A$="" :: DISPLAY AT(24,1) : 176 M$=A$ :: DISPLAY AT(24,1
103 REM              : 131 REM          : )BEEP:" display cleared " :: )BEEP:" MEMORY SAVED " :: FO
:                      :                      : FOR D=1 TO 200 :: NEXT D :: R D=1 TO 200 :: NEXT D :: DI
104 REM ** END ROUTINE ** : 132 REM ** DIVISION ROUTINE : DISPLAY AT(24,1):" " : SPLAY AT(24,1):" " :: GOTO
105 REM              : ** : 157 RETURN          : 59
106 CALL DELSPRITE(ALL):: CA : 133 REM          : 158 REM          : 177 REM
LL CLEAR :: PRINT "":" " 6 : 134 A$=STR$(VAL(B$)/VAL(A$)) : 159 REM          : 178 REM
GOODBYE ! "":" ":" ":" : :: OP=0 : 160 REM          : 179 REM
107 END              : 135 RETURN          :
108 REM              : 136 REM          : 161 REM ** USE NUMBER CURREN : 180 REM ** ROUTINE TO STOP E
109 REM              : 137 REM          : TLY STORED IN MEMORY IN A CA : NTERING A NUMBER WITH MORE T
110 REM              : 138 REM          : LCULATION. ** : HAN ONE DECIMAL POINT **
:                      :                      : 162 REM          : 181 REM182 REM IF THE NUMBE
:                      :                      : 163 REM          : R DOESN'T CONTAIN A DECIMAL
111 REM ** PERCENTAGE ROUTIN : 139 REM ** MULTIPLICATION RO : 164 REM ** IF MEMORY IS EMPT : 183 REM
E ** : UTINE ** : Y PRINT AN ERROR. OTHERWISE : 184 IF POS(A$,".",1)=0 THEN
112 REM              : 140 REM          : MAKE THE DISPLAY VALUE EQUAL : A$=A$"." ELSE 186
113 A$=STR$((VAL(B$)/100)*VA : 141 A$=STR$(VAL(B$)*VAL(A$)) : TO THE VALUE IN MEMORY ** : 185 GOTO 59
L(A$)):: OP=0 : :: OP=0 : 165 REM          : 186 DISPLAY AT(23,1)BEEP:" Y
114 RETURN          : 142 RETURN          : 166 IF M$="" THEN 168 ELSE 1 : OU CAN ONLY HAVE ONE DE
115 REM              : 143 REM          : 70 : CIMAL POINT IN A NUMBER!" ::
116 REM              : 144 REM          : 167 GOTO 59 : FOR D=1 TO 500 :: NEXT D ::
117 REM              : 145 REM          : 168 DISPLAY AT(24,1):" * ERR : DISPLAY AT(23,1)BEEP:" "
:                      :                      : OR : MEMORY EMPTY " :: CALL : 187 DISPLAY AT(24,1)BEEP:"
118 REM ** ADDITION ROUTINE : 146 REM ** ROUTINE TO TRANSF : 169 GOTO 59 : " :: GOTO 59
** : ER FIRST NUMBER THAT IS ENTE : SOUND(200,217,0):: FOR D=1 T : 188 REM
119 REM              : RED SO THAT SECOND NUMBER CA : 0 500 :: NEXT D : 189 REM
120 A$=STR$(VAL(A$)+VAL(B$)) : N BE ENTERED, AND ALSO TO RE : 170 A$=M$ :: GOTO 59 : 190 REM
:: OP=0 : SET FLAG OP,TO SHOW THAT : 171 REM          : 191 REM
121 RETURN          : 147 ! MATHEMATICAL FUNCTION : 172 REM          :
122 REM              : HAS BEEN ENTERED : 173 REM          :
123 REM              : 148 REM          : 174 REM ** ROUTINE TO SAVE C : 192 REM **** * * ****
124 REM              : 149 B$=A$ :: A$="" :: OP=1 : 175 REM          : 193 REM * ** * *
: 150 RETURN          : : 176 REM          : 194 REM *** ** * *
125 REM ** SUBTRACTION ROUTI : 151 REM          : 177 REM          : 195 REM * ** * *
NE ** : 152 REM          : 178 REM          : 196 REM **** * * ****
126 REM              : 153 REM          : 179 REM          : 197 REM
127 A$=STR$(VAL(B$)-VAL(A$)) : 154 REM ** CLEAR DISPLAY ROU : 198 REM END
:: OP=0 : TINE ** :
128 RETURN          : 155 REM          :
129 REM              :

```



KIDS CORNER

EACH ISSUE WE ARE GOING TO TURN OVER TO
THE T.I. KIDS OUT THERE TWO PAGES OF SHORT
PROGRAMS. SO ADULTS AND BOFFINS OUT

By Order
ED.

▶ FOLLOW THIS

TOO EASY JUST FOLLOW ON SCREEN INSTRUCTIONS.

▶ SNAKE OUT

NOT SO EASY.
USE THE ARROW KEYS "E.S.D.X" TO MOVE YOUR BLUE SNAKE
AVOID YOURSELF, THE OTHER SNAKE, THE TREES AND THE SIDES
TRY AND MAKE THE OTHER SNAKE CRASH BY BLOCKING HIM OFF

FOLLOW THIS EXTENDED BASIC

```

10 CALL CLEAR :: CALL SCREEN : 140 CALL CHAR(120,"FFFFFFFF" : 310 ACCEPT AT(22,8)BEEP:B$ : 450 CALL HCHAR(11,10,104,5):
(16) : FFFFFFFF) : : FOR T=1 TO 100 :: NEXT T : : CALL HCHAR(12,10,104,5)::
20 DISPLAY AT(5,5):"FOLLOW T : 150 CALL COLOR(9,5,1,10,11,1 : : DISPLAY AT(22,8):" : : CALL HCHAR(13,10,104,5):: CA
HIS" :: DISPLAY AT(9,5):"BY" : ,11,13,1,12,8,1) : 320 IF G$=R$ THEN 360 : LL HCHAR(14,10,104,5):: CALL
:: DISPLAY AT(13,5):"PAUL T : 160 CALL COLOR(5,16,1,6,16,1 : 330 DISPLAY AT(23,7)BEEP:"IT : SOUND(250,330,0)
EMPLAR" : ,7,16,1,8,16,1,3,16,1,4,16,1 : WAS ";R$ :: FOR T=1 TO 400 : 460 FOR T=1 TO 40 :: NEXT T
30 FOR T=1 TO 400 :: NEXT T : ) : : NEXT T :: DISPLAY AT(23, : 470 CALL HCHAR(11,10,32,5)::
40 CALL CLEAR : 170 CALL SCREEN(14) : 7):"YOU ONLY MADE ";S : CALL HCHAR(12,10,32,5):: CA
50 DISPLAY AT(1,1):"WATCH TH : 180 AD=1 : 340 FOR T=1 TO 500 :: NEXT T : LL HCHAR(13,10,32,5):: CALL
E COLOURS AND WHERE" :: DISP : 190 DISPLAY AT(6,14):"1" :: : : CALL CLEAR : HCHAR(14,10,32,5):: GOTO 270
LAY AT(3,1):"THEY CAME FROM, : DISPLAY AT(12,7):"2" :: DISP : 350 AD=1 :: S=0 :: GOTO 190 : 480 CALL HCHAR(11,19,112,5):
----- : LAY AT(12,22):"3" :: DISPLAY : 360 DISPLAY AT(23,8):"CORREC : : CALL HCHAR(12,19,112,5)::
60 DISPLAY AT(5,1):"(EXAMPLE : AT(19,14):"4" : T" : CALL HCHAR(13,19,112,5):: CA
BLUE=1)" :: DISPLAY AT(7,1) : 200 CALL HCHAR(5,8,128,18):: : 370 FOR T=1 TO 200 :: NEXT T : LL HCHAR(14,19,112,5):: CALL
:"(GREEN=3) AND SO ON....." : CALL VCHAR(6,8,128,14):: CA : : DISPLAY AT(23,8):" : SOUND(250,523,0)
70 DISPLAY AT(9,1):"JUST PRE : LL VCHAR(6,25,128,14):: CALL : " : 490 FOR T=1 TO 40 :: NEXT T
SS THE RIGHT KEY" :: DISPLAY : HCHAR(20,8,128,18) : 380 S=S+1 : 500 CALL HCHAR(11,19,32,5)::
AT(11,1):"THEN PRESS ENTER" : 210 DISPLAY AT(2,8):"FOLLOW : 390 FOR T=1 TO 100 :: NEXT T : CALL HCHAR(12,19,32,5):: CA
80 DISPLAY AT(16,1):"PRESS A : THIS:" :: R$="" : 400 AD=AD+1 : LL HCHAR(13,19,32,5):: CALL
NY KEY TO START" : 220 FOR D=1 TO AD :: P=INT(R : 410 GOTO 210 : HCHAR(14,19,32,5):: GOTO 270
90 CALL KEY(0,K,S):: IF S=0 : ND*4)+1 : 420 CALL HCHAR(7,14,100,5):: : 510 CALL HCHAR(15,14,120,5):
THEN 90 : 230 IF P=1 THEN 420 : CALL HCHAR(8,14,100,5):: CA : : CALL HCHAR(16,14,120,5)::
100 CALL CLEAR :: RANDOMIZE : 240 IF P=2 THEN 450 : LL HCHAR(9,14,100,5):: CALL : CALL HCHAR(17,14,120,5):: CA
110 CALL CHAR(100,"FFFFFFFFF : 250 IF P=3 THEN 480 : HCHAR(10,14,100,5):: CALL SO : LL HCHAR(18,14,120,5):: CALL
FFFFFFFFF):: CALL CHAR(104,"F : 260 IF P=4 THEN 510 : UND(250,110,0) : SOUND(250,494,0)
FFFFFFFFFFFFFFFFF):: CALL CHA : 270 IF D/4<>INT(D/4)THEN 280 : 430 FOR T=1 TO 40 :: NEXT T : 520 FOR T=1 TO 40 :: NEXT T
R(112,"FFFFFFFFFFFFFFFFF") : 280 R$=R$&STR$(P) : 440 CALL HCHAR(7,14,32,5):: : 530 CALL HCHAR(15,14,32,5)::
120 S=0 : 290 NEXT D : CALL HCHAR(8,14,32,5):: CALL : CALL HCHAR(16,14,32,5):: CA
130 CALL CHAR(128,"F1FEF2FEF : 300 FOR L=1 TO 200 :: NEXT L : HCHAR(9,14,32,5):: CALL HCH : LL HCHAR(17,14,32,5):: CALL
3FEF4FE"):: CALL COLOR(13,2, : AR(10,14,32,5):: GOTO 270 : HCHAR(18,14,32,5):: GOTO 270
1) : : :

```



★ SNAKE OUT ★

▶ IN BASIC



```

10 RANDOMIZE
20 CALL SCREEN(2)
30 FOR I=1 TO 16
40 CALL COLOR(I,2,2)
50 NEXT I
60 YS=0
70 CS=0
80 FOR A=1 TO 3
90 PRINT TAB(6);"      xpxpx : 380 CALL CHAR(129,"3C7EFF999 : 710 IF D=32 THEN 860 : 1110 AA=2
      xpxpx : 9FFF663C") : 720 GOSUB 800 : 1120 IF D<>32 THEN 1180
      xpxpx : 390 CALL CHAR(136,"7EFFFF7E" : 730 CALL GCHAR(RR,CC-2,D) : 1130 AA=0
      ke out x : 7FFFF7E") : 740 IF D=32 THEN 890 : 1140 CALL HCHAR(R,C,129)
      xpxpxpxpxp"::: : 400 CALL CHAR(137,"3C7EFF999 : 750 GOSUB 800 : 1150 CALL SOUND(10,INT(RND*5
      100 NEXT A : 9FFF663C") : 760 CALL GCHAR(RR,CC+2,D) : 00)+110,0,INT(RND*500)+110,0
      110 PRINT " Edwin Armstro : 410 CALL CHAR(144,"183C7EFF1 : 770 IF D=32 THEN 920 : 1160 CALL HCHAR(RR,CC,137)
      ng 1985"::"xpxpx" PRESS ANY : 8181818") : 780 GOSUB 800 : 1170 GOTO 550
      KEY <xpxpx"::: : 420 R=INT(RND*19)+3 : 790 GOTO 670 : 1180 CALL SOUND(1000,-7,0)
      120 CALL CHAR(112,"FFFFFFF : 430 C=INT(RND*27)+3 : 800 H=H+1 : 1190 FOR X=1 TO 60
      FFFFFFFF") : 440 RR=INT(RND*19)+3 : 810 IF H<>4 THEN 820 ELSE 83 : 1200 IF AA=2 THEN 1300
      130 CALL CHAR(120,"FFFFFFF : 450 CC=INT(RND*27)+3 : 0 : 1210 CALL COLOR(13,7,16)
      FFFFFFFF") : 460 CALL CLEAR : 820 RETURN : 1220 CALL COLOR(13,2,16)
      140 FOR I=2 TO 11 : 470 CALL COLOR(1,16,16) : 830 CALL GCHAR(RR-1,CC,D) : 1230 NEXT X
      150 CALL COLOR(1,16,2) : 480 CALL HCHAR(23,1,120,130) : 840 IF D=32 THEN 1010 : 1240 CALL CLEAR
      160 NEXT I : 490 CALL VCHAR(2,31,120,94) : 850 GOSUB 960 : 1250 CALL COLOR(1,1,2)
      170 FOR I=2 TO 15 : 500 FOR J=1 TO 40 : 860 CALL GCHAR(RR+1,CC,D) : 1260 IF AA=2 THEN 1330
      180 CALL COLOR(12,I,2) : 510 CALL HCHAR(INT(RND*19)+3 : 870 IF D=32 THEN 1030 : 1270 PRINT "xxxxxxTHE COMPUT
      190 CALL KEY(0,K,S) : ,INT(RND*27)+3,144) : 880 GOSUB 960 : ER WONxxxxxx":::
      200 IF S<>0 THEN 240 : 520 NEXT J : 890 CALL GCHAR(RR,CC-1,D) : 1280 CS=CS+1
      210 CALL COLOR(13,1+1,2) : 530 CALL HCHAR(R,C,129) : 900 IF D=32 THEN 1050 : 1290 GOTO 1350
      220 NEXT I : 540 CALL HCHAR(RR,CC,137) : 910 GOSUB 960 : 1300 CALL COLOR(14,13,16)
      230 GOTO 170 : 550 CALL KEY(3,K,S) : 920 CALL GCHAR(RR,CC+1,D) : 1310 CALL COLOR(14,2,16)
      240 CALL CLEAR : 560 IF S=0 THEN 550 : 930 IF D=32 THEN 1070 : 1320 GOTO 1230
      300 IF S=0 THEN 290 : 570 IF (K=68)+(K=69)+(K=83)+ : 940 GOSUB 960 : 1330 PRINT "xxxxxxCONGRATULA
      310 CALL CLEAR : (K=88) THEN 580 ELSE 550 : 950 GOTO 830 : TIONSxxxxxx"::"xxxxxxxxxxYOU
      320 AA=0 : 580 CALL HCHAR(R,C,128) : 960 H=H+1 : 1340 YS=YS+1
      330 CALL COLOR(12,10,1) : 590 C=C+(K=68)*(C<31)-(K=83) : 970 IF H<>8 THEN 1000 : 1350 PRINT "SCORES"::"YOU="
      340 CALL COLOR(13,5,16) : *(C>2) : 980 AA=2 : ;YS::"COMPUTER=";CS;::"xxx
      350 CALL COLOR(14,14,16) : 600 R=R+(K=88)*(R<23)-(K=69) : 1000 RETURN : < PRESS REDO OR BACK >xxx"::
      360 CALL COLOR(15,13,16) : *(R>2) : 1010 RR=RR-1 : 1360 CALL KEY(0,K,S)
      370 CALL CHAR(128,"7EFFFF7E : 610 H=0 : 1020 GOTO 1080 : 1370 IF K=6 THEN 340
      7FFFF7E") : 620 CALL HCHAR(RR,CC,136) : 1030 RR=RR+1 : 1380 IF K=15 THEN 20 ELSE 13
      : 630 G=INT(RND*4) : 1040 GOTO 1080 : 60
      : 640 IF G=1 THEN 670 : 1050 CC=CC-1 : 1390 END
      : 650 IF G=2 THEN 700 : 1060 GOTO 1080 :
      : 660 IF G=3 THEN 730 ELSE 760 : 1070 CC=CC+1 :
      : 670 CALL GCHAR(RR-2,CC,D) : 1080 CALL GCHAR(R,C,D) :
      : 680 IF D=32 THEN 830 : 1090 IF D<>32 THEN 1180 :
      : 690 GOSUB 800 : 1100 CALL GCHAR(RR,CC,DD) :
      : 700 CALL GCHAR(RR+2,CC,D) :

```


ALPINER	92,186
ATTACK	388,800
BIGFOOT	23,270*
BLASTO	47,340
BUCK ROGERS	292,194
BURGERTIME	500,000
CAR WARS	56,250
CHISHOLM TRAIL	89,990
CONGO BONGO	250,000*
DEFENDER	3,720,000*
DONKEY KONG	1,741,300
FATHOM	5,881
HOPPER	32,500*
INVADERS	54,121
JAW BREAKER	6,435
JUNGLE HUNT	29,180
M*A*S*H	4,027
MICRO SURGEON	4,200
MOON MINE	131,720*
MOON PATROL	133,150
MOON SWEEPER	19,150
MS.PAC-MAN	151,610
MUNCHMAN	458,580*
MUNCHMOBILE	27,290
PAC-MAN	43,180
PARSEC	10,842,800
PICNIC PARANOIA	612,520*
POLE POSITION	110,500*
PROTECTOR II	125,764
SHAMUS	104,790*
SLYMOIDS	113,905
SNEGGIT	2,635
STAR TREK	660,110
SUPER DEMON ATTACK	6,375*
TOMBSTONE CITY	2,135,950*

S. LEACH	on	BIGFOOT
STEPHEN GRIGG	on	CONGO BONGO
JONATHAN LAMBERN	on	DEFENDER
MOGEN JENSEN	on	HOPPER
C. SINAR	on	MOON MINE
PAUL MORRIS	on	MUNCHMAN
SCOTT CLELLAND	on	PICNIC PARA
S. HARRIS	on	POLE POSITION
STEPHEN KING	on	SUPER DEMON
ANDREW WEBB	on	SHAMUS
M. LYFORD	on	TOMBSTONE CTY

SEND YOUR HIGH SCORES TO

The Editor

PARCO MAGAZINE

2 Devonshire Court. Heathpark.

Honiton. Devon. England EX14 8SB

SHAMUS
PROTECTOR II
MASH
STAR TREK
AMERICAN FOOTBALL

PARCO PACK II
PARCO GOLF
TI DRONE
MINI MINER
DRAGON SLAYER

PERSONAL RECORD KEEPING
EXTENDED BASIC
TERMINAL EMULATOR II
LIST FILE
ARTIST

MASTERING THE TI
33 EXTENDED BSAIC PROGS
TEXAS PROGRAM BOOK
GETTING STARTED WITH THE TI
DYNAMIC GAMES