

ELBUG

SUPPLEMENT TO BEEBUG

VOLUME 3 NUMBER 7

JUNE 1986

WORDSEARCH

Master of the BEEBUG game, Jonathan Temple turns his hand to the Electron with a puzzle to tax the most ingenious of ELBUG minds.

Wordsearch puzzles continually crop up in magazines. The basis for these puzzles is a large grid of seemingly random letters within which are hidden several words written vertically, horizontally and diagonally, and often backwards. Your task is to find the words, usually from a list supplied - hence the name 'Wordsearch'.

The Wordsearch program supplied here will allow you to create these puzzles with your own list of words. Once this has been done you may print the Wordsearch out, attempt to solve it, or (useful for cheats) reveal the positions of the words in the grid. Who knows - you may even manage to sell your completed wordsearch to one of the puzzle magazines!

When you want to create a wordsearch you must first type in your list of words. To do this, choose option 1 from the menu, and enter as many words as required (up to a maximum of 32). Press Escape when you have finished. The word list will be automatically sorted into alphabetical order, and a new wordsearch puzzle generated. This may take some time,

depending on the number and length of the words you have typed in.

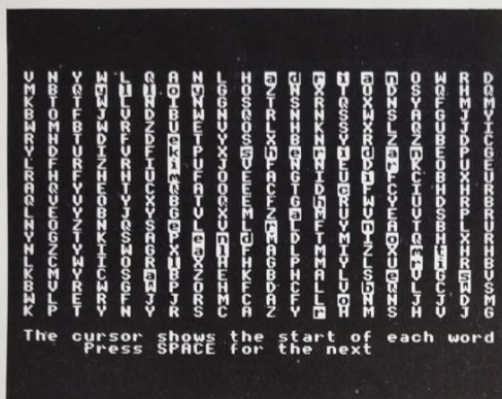
Any mistakes can be edited with option 2. Type the number of the word to be altered or added, and then type in the new word. There will be a short pause as the new list of words is re-sorted and then you can edit another word or press Escape to generate a new puzzle.

Option 3 displays the complete puzzle and allows you to try solving the wordsearch. The word you have to find is displayed, and you should move to the start of the word using the cursor keys and press Return. You will then be prompted to move to the end of the word, again using the cursor keys followed by Return. When (if?) you finally discover all the words you will be greeted with a quick tune! Note that if you are unable to find the current word, pressing the space bar will allow you to move onto the next.

Option 4 displays the solution to the puzzle. The grid is displayed and the first word highlighted in inverse, lower

G CASSETTE ELBUG CASSETTE ELBUG CASSETTE ELBUG CASSETTE ELBUG CASSETTE ELBUG CASSETTE E

This month's ELBUG magazine cassette includes several of the programs from this month's BEEBUG - the Computer Simulation program, the utility to pass arrays to procedures, the three sideways RAM programs, the six Workshop examples, and the First Course ADVAL examples - plus the Wordsearch puzzle and Sum Invaders game from this month's ELBUG supplement. All this for just £3.00 plus 50p post and packing from the St. Albans address.



case characters. Press the space bar to move onto the next.

If you have a printer and a suitable interface, option 5 allows you to print out the wordsearch with a title and the list of words. As listed, the program does this in emphasized print with an Epson-compatible printer. However, the codes in line 2020 to produce this can easily be changed to suit another printer.

A new wordsearch, using the same list of words, can be generated using the re-shuffle option.

The program should be fairly straightforward to enter. As always, save the program before you attempt to run it. At least while debugging the program, always use option 7 to leave it as this will reset the cursor keys and the Escape key to their normal functions.

PROGRAM NOTES

The program is quite long and so you may wish to omit the 'solve' and 'reveal' features which take up much of the listing, and just stick with the puzzle generating and printing sections. To do this, miss out lines 150, 160 and 1380-1870. In addition, some users may find the alphabetical sort feature a nuisance because groups of words (e.g. names) may be split across the word list. To remedy this you should miss out line 1360.

The procedures and functions used are:
 init Initializes arrays etc.
 enter Procedure for entering the word list.

p edit Procedure for editing the word list.
p sort Procedure for sorting the word list.
p solve Procedure for solving the wordsearch.
p tune Plays the congratulations tune.
p reveal Procedure for revealing the words in the wordsearch.
p highlight Highlights the current word.
p display Displays the wordsearch grid and prints if necessary.
p on Switches printer on for PROCdisplay.
p extra Prints title, underlining for PROCdisplay.
p list Prints word list for PROCdisplay.
p generate Procedure for generating a new wordsearch.
p interlock Used by PROCgenerate to allow words to interlock.
p inverse Highlights character at cursor in inverse lower case.
p menu Prints menu and returns option chosen.
p inside Tests for position inside grid.
p start Tests for position at start of word.
p finish Tests for position at finish of word.
p input Returns input string.
p key Returns key press, moves cursor if necessary.

```

10 REM PROGRAM WORDSEARCH
20 REM VERSION E0.2
30 REM AUTHOR J. Temple
40 REM ELBUG JUNE 1986
50 REM PROGRAM SUBJECT TO COPYRIGHT
60 :
100 MODE 6
110 PROCinit
120 REPEAT:C%=FNmenu
130 IF C%=1 PROCcenter
140 IF C%=2 PROCedit
150 IF C%=3 PROCsolve
160 IF C%=4 PROCreveal
170 IF C%=5 PROCdisplay(2)
180 IF C%=6 PROCgenerate
190 UNTIL C%=7
200 *FX 4,0
210 *FX 229,0
220 END

```

```

230 :
1000 DEFPROCcenter
1010 PRINT TAB(8,20);"Enter up to 32 wo
rds"TAB(7);"with 2-10 letters each."TAB
AB(7);"Press ESCAPE to finish";
1020 W%=0:REPEAT W%=W%+1
1030 IF W%<17 PRINT TAB(0-(W%<10),W%);
ELSE PRINT TAB(20,W%-16);
1040 PRINT;W%;" ";
1050 W$(W%)=FNinput(2,10,65,122)
1060 UNTIL E% OR W%=32
1070 N%=W%:IF W%<32 N%=W%-1
1080 IF N%>1 PROCsort
1090 VDU 28,0,24,39,18,12
1100 PROCgenerate:VDU 26
1110 ENDPROC
1120 DEFPROCedit
1130 PRINT TAB(7,20);"Press ESCAPE to f
inish"
1140 REPEAT:FOR W%=1 TO N%
1150 IF W%<17 PRINT TAB(0-(W%<10),W%);
ELSE PRINT TAB(20,W%-16);
1160 PRINT ;W%;" ";W$(W%);SPC(10-LEN(W$
(W%)));
1170 NEXT:W%=FALSE
1180 REPEAT VDU 28,0,24,39,23,12,26
1190 PRINT TAB(0,23);"Edit word number
(1-"N%+1+(N%=32);" ) ?";
1200 VDU B%:W%=VAL(FNinput(1,2,48,57))
1210 B%=7
1220 UNTIL E% OR (W%>0 AND W%<33 AND W%
<=N%+1)
1230 IF E% GOTO 1300
1240 VDU 28,0,24,39,23,12,26
1250 PRINT TAB(0,23);"New word for numb
er ";W%;" :";
1260 L$=FNinput(2,10,65,122)
1270 IF L$>"" W$(W%)=L$
1280 IF W%>N% N%=W%
1290 IF N%>1 PROCsort
1300 UNTIL E%
1310 VDU 28,0,24,39,20,12,30
1320 PROCgenerate:VDU 26:ENDPROC
1330 DEFPROCsort
1340 FOR J%=1 TO N%-1
1350 FOR K%=J%+1 TO N%
1360 IF W$(J%)>W$(K%) T$=W$(J%):W$(J%)=
W$(K%):W$(K%)=T$
1370 NEXT,:ENDPROC
1380 DEFPROCsolve
1390 PROCdisplay(1):W%=0:F%=0
1400 FOR J%=1 TO N%:F%(J%)=0:NEXT
1410 REPEAT:REPEAT W%=W%+1
1420 UNTIL F%(W%)=FALSE OR W%>N%
1430 IF W%>N% W%=1
1440 PRINT TAB(0,22);"Now move to the s
tart of ";W$(W%);SPC(9);TAB(0,23);"using
the cursor keys and press RETURN""Pres
s SPACE to move onto next word";
1450 VDU 30,10
1460 REPEAT:REPEAT K%=FNkey
1470 UNTIL K%=13 OR K%=27 OR K%=32
1480 IF K%=13 VDU 7
1490 UNTIL FNstart(POS DIV 2+1,VPOS) OR
K%=27 OR K%=32
1500 IF K%>13 GOTO 1670
1510 J%=POS:K%=VPOS:PROCinverse
1520 VDU 28,0,24,39,22,12
1530 PRINT TAB(14);"WELL DONE"TAB(3);"
Now move to the end of the word";
1540 VDU 26,31,J%,K%
1550 REPEAT:REPEAT K%=FNkey
1560 UNTIL K%=13 OR K%=27 OR K%=32
1570 IF K%=13 VDU 7
1580 UNTIL FNfinish(POS DIV 2+1,VPOS) O
R K%=27 OR K%=32
1590 IF K%>13 GOTO 1670
1600 PROChighlight(W%)
1610 VDU 28,0,24,39,22,12:F%=F%+1
1620 F$="You've found the whole word"
1630 IF F%=N% F$="WELL DONE - that was
the last one!"
1640 PRINT TAB((37-LEN(F$))/2);F$TAB(
13);"Press SPACE";
1650 F%(W%)=TRUE:IF F%=N% PROCTune
1660 REPEAT UNTIL GET=32:VDU 12,26
1670 UNTIL K%=27 OR F%=N%:ENDPROC
1680 DEFPROCtune
1690 N=81:RESTORE 2920
1700 FOR T=1 TO 10:READ D,L:N=N+D
1710 SOUND 1,1,N,L
1720 NEXT:ENDPROC
1730 DEFPROCreveal
1740 PROCdisplay(1)
1750 PRINT TAB(0,22);"The cursor shows
the start of each word"TAB(5);"Press SP
ACE for the next";
1760 FOR W%=1 TO N%
1770 PROChighlight(W%)
1780 REPEAT UNTIL GET=32:NEXT:ENDPROC
1790 DEFPROChighlight(W%)
1800 G%=X%(W%):H%=Y%(W%)
1810 J%=V%(W%)-G%:K%=W%(W%)-H%
1820 REPEAT
1830 VDU 31,(G%-1)*2,H%:PROCinverse
1840 L%=G%:M%=H%
1850 G%=G%+SGN(J%):H%=H%+SGN(K%)
1860 UNTIL FNfinish(L%,M%)
1870 VDU 31,X%(W%)*2-2,Y%(W%):ENDPROC
1880 DEFPROCdisplay(F%)
1890 F%=0:IF F%=2 PROCcon
1900 IF E% GOTO 2110
1910 FOR K%=1 TO 20:FOR J%=1 TO 20
1920 PRINT TAB(J%*2-2,K%);
1930 IF F%=1 PRINT ;
1940 PRINT GS(J%,K%);
1950 IF F%=2 PRINT " ";
1960 NEXT:IF F%=2 PROCextra
1970 NEXT:VDU 6,3:ENDPROC
1980 DEFPROCcon

```

```

1990 PRINT "Title of wordsearch""";
2000 N$=FNinput(0,26,32,126)
2010 PRINT TAB(4); "Please wait while wo
rdsearch" TAB(13); "is printed"
2020 VDU 2,1,27,1,69,21:REM emphasised
2030 ENDPROC
2040 DEFPROCextra
2050 IF K%=1 PRINT SPC(5);N$;
2060 IF K%=2 PRINT SPC(5);STRING$(LEN(N
$),"*");
2070 IF K%>3 PROClist
2080 PRINT:ENDPROC
2090 DEFPROClist
2100 J%=(N% DIV 2)+(N% MOD 2)
2110 IF (K%-3)>J% ENDPROC
2120 PRINT " ";W$(K%-3);
2130 IF (K%-3)<=(N% DIV 2) PRINT STRING
$(15-LEN(W$(K%-3))," ");W$(J%+K%-3);
2140 ENDPROC
2150 DEFPROCgenerate
2160 PRINT TAB(1); "Please wait while a
new wordsearch" TAB(12); "is generated"
2170 FOR J%=1 TO 20:FOR K%=1 TO 20
2180 G$(J%,K%)="" :NEXT,
2190 FOR L%=1 TO N%
2200 W$=W$(L%):REPEAT R%=RND(8)
2210 J%=J%(R%):K%=K%(R%):REPEAT
2220 G%=RND(20):H%=RND(20)
2230 L$=G$(G%,H%)
2240 IF L$>"" PROCinterlock
2250 UNTIL FNinside(G%,H%)
2260 P%=1:Q%=G%:R%=H%:REPEAT
2270 F%=FNinside(G%,H%)
2280 IF F% THEN F%=(G$(G%,H%)="" OR G$(
G%,H%)=MID$(W$,P%,1))
2290 G%=G%+J%:H%=H%+K%:P%=P%+1
2300 UNTIL F%=0 OR P%>LEN(W$)
2310 UNTIL F%
2320 X%(L%)=Q%:Y%(L%)=R%
2330 FOR U%=1 TO LEN(W$)
2340 G$(Q%,R%)=MID$(W$,U%,1)
2350 Q%=Q%+J%:R%=R%+K%:NEXT
2360 V%(L%)=Q%-J%:W%(L%)=R%-K%:NEXT
2370 FOR J%=1 TO 20:FOR K%=1 TO 20
2380 IF G$(J%,K%)="" G$(J%,K%)=CHR$(RND
(26)+64)
2390 NEXT,:ENDPROC
2400 DEFPROCinterlock
2410 P%=0:REPEAT P%=P%+1
2420 UNTIL P%>LEN(W$) OR MID$(W$,P%,1)=
G$(G%,H%)
2430 IF P%<=LEN(W$) G%=G%-(P%-1)*J%:H%=
H%-(P%-1)*K%
2440 ENDPROC
2450 DEFNFinside(G%,H%)=G%>0 AND G%<21
AND H%>0 AND H%<21
2460 DEFNFstart(G%,H%)=(G%=X%(W%) AND H
%=Y%(W%))
2470 DEFNFfinish(G%,H%)=(G%=V%(W%) AND
H%=W%(W%))

```

```

2480 DEFNFinput(P%,Q%,R%,S%)
2490 L$="" :REPEAT L%=LEN(L$):G%=GET
2500 IF G%=27 E%=TRUE:G%=13:IF L%=0 GOT
O 2580
2510 IF G%=127 IF L%>0 L$=LEFT$(L$,L%-1
):GOTO 2570
2520 IF G%=13 IF L%>P% GOTO 2570
2530 E%=FALSE
2540 IF G%<32 OR LEN(L$)=Q% OR (G%>90 A
ND G%<97) THEN G%=7
2550 IF G%>96 IF G%<123 G%=G%-32
2560 IF G%>R% AND G%<=S% L$=L$+CHR$(G%
) ELSE G%=7
2570 A%=G%:CALL &FFE3
2580 UNTIL G%=13:=L$
2590 DEFNFkey:K%=GET
2600 IF K%=136 VDU 8,8 ELSE IF K%=137 V
DU 9,9 ELSE IF K%=138 AND VPOS<20 VDU 10
ELSE IF K%=139 AND VPOS>1 VDU 11
2610 =K%
2620 DEFPROCinverse
2630 A%=135:REPEAT
2640 Z%=(USR(&FFF4)AND&FF00)/256
2650 VDU 17,0,17,129:UNTIL Z%
2660 VDU Z% OR 32,17,1,17,128:ENDPROC
2670 DEFNFmenu
2680 CLS
2690 PRINT TAB(9); "ELECTRON WORDSEARCH"
2700 FOR M%=1 TO 7
2710 PRINT TAB(0,2*M%+4);M%;" " :M$(M%)
2720 NEXT:B%=FALSE:REPEAT
2730 PRINT TAB(0,19) "Enter option (1-
7) ?";
2740 REPEAT VDU B%:K%=GET AND &CF
2750 B%=7:UNTIL K%>0 AND K%<8
2760 IF N%=0 IF K%>1 IF K%<7 PRINT "Y
ou must enter some words first!";
2770 UNTIL K%=1 OR K%=7 OR N%>0
2780 CLS:=K%
2790 DEFPROCinit
2800 DIM G$(20,20),W$(32),M$(7),X$(32),
Y$(32),V$(32),W$(32),F$(33),J$(8),K$(8)
2810 N%=0:FOR L%=1 TO 8
2820 READ J%(L%),K%(L%):NEXT
2830 FOR L%=1 TO 7:READ M$(L%):NEXT
2840 ENVELOPE 1,1,0,0,0,0,0,0,126,-1,0,
-5,126,0
2850 *FX 4,1
2860 *FX 229,1
2870 VDU 19,1,2;0;:ENDPROC
2880 ENDPROC
2890 DATA -1,1,0,1,1,1,-1,0
2900 DATA 1,0,-1,-1,0,-1,1,-1
2910 DATA "Enter new words","Edit curre
nt words","Solve current grid","Reveal w
ords in grid","Print out grid","Re-shuff
le grid","Exit program"
2920 DATA 0,4,8,4,8,4,4,4,8,8
2930 DATA -12,8,4,8,-12,8,8,8,-16,8

```

Your Guide to BEEBUG

No alterations are required to the programs appropriate to the Electron from this month's BEEBUG. However, for those ELBUG members interested in BBC micro programs, a few general comments on converting Beeb programs to the Electron may prove useful.

Beeb programs sometimes fail to run properly on the Electron because of the Beeb's mode 7 and extra sound facilities. However, conversion rarely proves very difficult.

The Beeb's mode 7 display is of the same format as the mode 6 display that ELBUG members know but with a full range of colours and crude graphics. Mode 7 is not available on the Electron. However, the Electron defaults to mode 6 when it encounters a MODE 7 command. So BBC micro programs that include a mode 7 display will therefore run on the Electron but often with a corrupted (and monochrome) display.

The Beeb has three sound channels each with full control over the volume of notes. Electron Basic includes all the SOUND and ENVELOPE parameters that are used in the BBC micro even though several have no effect. This means that Beeb programs making full use of sound will still run on the Electron unaltered, though not, of course, producing the full sound effect.

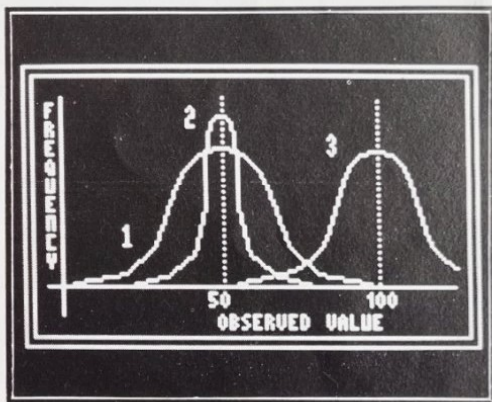
The other major difference between the Electron and the BBC micro is speed. The Electron is much slower, and particularly slow in modes 0, 1, and 2. BBC micro programs often benefit from a change of mode (admittedly, with a loss of some colours) when running them on the Electron. When changes are needed they will be given in this ELBUG supplement.

COMPUTER SIMULATION (Part 1)

If you want to use your Electron for more than just playing games then computer simulation is one avenue open to you. Although this may seem a daunting subject it can be quite easy to achieve some useful results with just a little know-how. That is where this article comes in useful.

The whole article is applicable to the

Electron and the programs work without alteration too.



PASSING ARRAYS TO PROCEDURES

Although BBC Basic is a very powerful and flexible computer language, it does have limitations. This article will help to rectify one of these shortcomings.

This utility will allow you to pass the whole contents of an array to a procedure 'by reference' in the same way as individual parameter values may be passed at present.

The whole article is applicable to ELBUG members.

TRACING WORDWISE PLUS

Unfortunately the Wordwise Plus word processor requires a mode 7 display and so it is not available for the Electron.

SOFTWARE FOR SIDWAYS RAM (Part 1)

The sideways ROM/RAM system employed in the BBC micro and Electron is a very powerful device. This article, the first of a series, will explain how to make the most of this system.

To make use of the program accompanying this article you will need

either a sideways ROM/RAM board, such as the Slogger ROM board, or a Plus-1 interface along with a sideways RAM cartridge such as that supplied by Advanced Computer Products.

BEEBUG FILER GOES GRAPHIC

This disc-based program is designed to add powerful graphics to the database program, Filer, published previously in BEEBUG. You will need a Cumana or similar disc interface similar to the Beeb's DFS. The program will not work with the Plus 3 and ADFS for lack of space.

TALKING HEADS

This excellent program to generate conversations will not operate on an Electron as it requires the large amounts of memory only available with a mode 7 display on the BBC micro.

WORKSHOP

Polishing your Programs

Everyone's Basic programs can benefit from a little polishing. This useful article will help you to put the finishing touches on your own work. This can have the practical benefit of speeding the programs up too.

As the Electron uses the same Basic as the BBC micro, this article and the six programs accompanying it, are perfectly compatible with the Electron.

FIRST COURSE

Using the ADVAL function

Not all uses of the ADVAL function are dependent upon having an analogue to digital (A to D) convertor. Thus all Electron owners may find something of interest.

However, if you have a Plus-1 (or other) interface connected to your Electron - which adds a joystick socket to your system - then the whole article will be relevant to your computer.

HARDWARE ANIMATION

As this uses the hardware scrolling specific to the BBC micro, the technique described in this article is not applicable to the Electron.

THE MASTER SERIES

Opening Up The Private RAM

This article on the use of the private RAM of the Master 128 is the first of a series of articles on Acorns new machines. The feature described here is not available on the Electron or on the BBC micro model B/B+.

THE WIGMORE MOUSE

This rodent rival to the AMX mouse is not available for the Electron.

MACRO ASSEMBLERS

Unfortunately only one of the two macro assemblers reviewed here will operate with the Electron. This is the Star macro assembler and it is recommended to ELBUG members interested in serious assembler work.

MOVIE MAKER

Movie Maker is used to create animated mode 7 displays and as such it will not operate on the Electron.

SOUNDS UNNATURAL

The Computer Concepts text-to-speech ROM requires the BBC micro's speech processor chip, which cannot be fitted to an Electron. For different reasons, Superior Software's system will also fail to work on an Electron.

LIGHT FANTASTIC

The excellent Colourspace light synthesizer will not operate on the Electron.

SUPERCHARGING THE MASTER

Although the Turbo co-processor for the Master is not available for the Electron or the BBC micro model B/B+, ELBUG members thinking of upgrading to Acorn's latest machine will be interested in this powerful combination.

HINTS

Of this month's hints and tips, the following are applicable to Electron owners:

MASKING
HALF PROCEDURES
DECODING USR
HIGH PRECISION ANALOGUE

SUM INVADERS

When you've tested your power over words with 'Wordsearch', try out your numeric capabilities with this ingenious game from Craig Brown.

This program will test your powers of mental arithmetic and provide some arcade fun, all at the same time.

The fearsome sum invaders have appeared in the skies above the Earth and you are in charge of the only laser base that can defeat them (of course!). The invaders appear in groups of four spaceships. Each ship displays a possible answer to a sum at the bottom of the screen. You must use the laser base to hit the right ship to ward off the invasion. If you miss the ships altogether then you can try again. However, if you hit the wrong ship the whole squadron will descend with terrible consequences - no score.

Just to add to the difficulty (the sum invaders are a particularly nasty bunch) there is a limit to the time you can spend thinking about your answer. This is set to four seconds. You must think fast and accurately to save the Earth.

There is a choice of four categories of sum - addition, subtraction, multiplication, and division. Ten sums are given for each category and then the chance to choose another category.

PROGRAM NOTES

The program is well structured so you shouldn't have too much difficulty following it. The length of time allowed to answer each question is preset in line 1010. This can be increased if your mental maths is a bit rusty.

The number of questions given before a change of category is allowed is specified by the FOR-NEXT loop in line 140 and again this can be changed if you want.

```
10 REM PROGRAM SUM INVADERS
20 REM VERSION E0.2
30 REM AUTHOR CRAIG BROWN
40 REM ELBUG JUNE 1986
50 REM PROGRAM SUBJECT TO COPYRIGHT
60 :
70 MODE 4
100 DIM S$(2,3),A$(3),R$(3)
```

```
110 PROCinit
120 REPEAT
130 PROCchoose
140 FOR GO%=1 TO 10:PROCCalc
150 REPEAT:PROCbase:UNTIL F%
160 NEXT GO%
210 UNTIL FALSE
220 END
230 :
1000 DEF PROCinit
1010 EASE%=400
1020 VDU23,1,0;0;0;0;0;:FX9,2
1030 VDU19,0,4;0;:FX10,2
1040 BS=CHR$240:TT%=0:SC%=-1
1050 FOR I%=0 TO 3
1060 S$(0,I%)=CHR$224+CHR$225+CHR$226+CHR$227+CHR$228
1070 S$(2,I%)=CHR$231+CHR$232+CHR$233+CHR$234+CHR$235
1080 NEXT I%
1090 VDU23,224,0,7,15,31,60,56,112,112
1100 VDU23,225,0,255,255,255,0,0,0,0
1110 VDU23,226,255,255,255,255,0,0,0,0
1120 VDU23,227,0,255,255,255,0,0,0,0
1130 VDU23,228,0,224,240,248,60,28,14,14
4
1140 VDU23,229,224,224,224,224,224,224,240,112,112
1150 VDU23,230,7,7,7,7,15,14,14
1160 VDU23,231,56,60,31,15,7,6,12,62
1170 VDU23,232,0,0,255,255,255,30,0,0
1180 VDU23,233,0,0,255,255,255,0,0,0
1190 VDU23,234,0,0,255,255,255,120,0,0
1200 VDU23,235,28,60,248,240,224,96,48,124
1210 VDU23,240,24,24,24,24,24,126,255,255
55
1220 ENDPROC
1230 :
1240 DEF PROCbase
1250 PROCprint(X%," ")
1260 IF INKEY(-98) X%=X%-1-(X%<2)
1270 IF INKEY(-67) X%=X%+1+(X%>37)
1280 IF INKEY(-1) PROCfire
1290 IF NOT F% PROCprint(X%,BS)
1300 IF TIME>EASE% AND NOT F% PROCclose:
F%=TRUE
1310 ENDPROC
1320 :
1330 DEF PROCprint(X%,P$)
1340 *FX19
1350 PRINT TAB(X%,25);P$
1360 ENDPROC
```

```

1370 :
1380 DEF PROCfire
1390 PRINT TAB(X%,25);B$
1400 FOR I%=0 TO 3:GCOL0,1:*FX19
1410 MOVE X%*32+16,240:DRAW X%*32+16,80
0
1420 GCOL0,0:*FX19
1430 MOVE X%*32+16,240:DRAW X%*32+16,80
0
1440 NEXT
1450 IF X%>1 AND X%<7 PROCscore(0)
1460 IF X%>11 AND X%<17 PROCscore(1)
1470 IF X%>22 AND X%<28 PROCscore(2)
1480 IF X%>32 AND X%<38 PROCscore(3)
1490 ENDPROC
1500 :
1510 DEF PROCdescend
1520 VDU26:X%=19:PROCprint(X%,B$)
1530 VDU28,0,20,39,0
1540 FOR AA%=0 TO 3
1550 A$=STR$(A%(AA%)):S$(1,AA%)=CHR$229
+STRINGS(3-LEN(A$)," ")A$+CHR$230
1560 NEXT AA%
1570 FOR L%=2 TO -2 STEP -1
1580 IF L%>-1 PRINT TAB(2)S$(L%,0)TAB(1
2)S$(L%,1)TAB(23)S$(L%,2)TAB(33)S$(L%,3)
1590 VDU30,11:*FX19
1600 NEXT L%
1610 TT%=TT%+10
1620 IF SC%=-1 SC%=0
1630 VDU26:ENDPROC
1640 :
1650 DEF PROCchoose
1660 CLS:GCOL0,1:MOVE0,180:DRAW1279,180
1670 PRINT TAB(14,2)"SUM INVADERS"
1680 PRINT TAB(12,10)"Z" - Left"
1690 PRINT TAB(12,12)"X" - Right"
"
1700 PRINTTAB(10,14)"Shift - Fire"
"
1710 VDU28,8,30,39,27
1720 IF SC%=-1 THEN 1750
1730 PRINT"" Score = ";SC%;" out of "
;TT%
1740 T%=TIME+200:REPEAT UNTIL TIME>T%
1750 PRINT"" Select: +, -, *, or /"
1760 REPEAT G$=GET$:UNTIL INSTR("+*/*",
G$):ENDPROC
1770 :
1780 DEF PROCcalc
1790 VDU28,0,25,39,0,12,28,8,30,39,27,1
2
1800 T%=TIME+100:REPEAT UNTIL TIME>T%
1810 IFG$<>"+" THEN 1840
1820 B%=RND(993):C%=RND(994-B%)
1830 A%=B%+C%:GOTO1940
1840 IFG$<>"-" THEN 1870
1850 B%=RND(994):C%=RND(B%)
1860 A%=B%-C%:GOTO1940
1870 IFG$<>"*" THEN 1910
1880 B%=RND(500):C%=RND(10)+1
1890 A%=B%*C%:IF A%>994 THEN 1880
1900 GOTO1940
1910 IFG$<>"/" THEN 1750
1920 A%=RND(994):C%=RND(10)+1
1930 B%=A%*C%:IF B%>994 THEN 1920
1940 FOR J%=0 TO 3:A%(J%)=0:NEXT J%
1950 FOR I%=0 TO 3
1960 A=A%+RND(8)-4:IF A=A% THEN 1960
1970 D%=FALSE:FOR J%=0 TO 3:IF A=A%(J%)
D%=TRUE:NEXT ELSE NEXT
1980 IF D% THEN 1960
1990 A%(I%)=A
2000 NEXT I%
2010 AN%=RND(4)-1:A%(AN%)=A%
2020 PRINT "SPC(7);STR$(B%);" ";G$;" "
;STR$(C%)
2030 T%=TIME+100:REPEAT UNTIL TIME>T%
2040 PROCdescend:F%=FALSE:TIME=0
2050 ENDPROC
2060 :
2070 DEF PROCscore(S%)
2080 IF A%(S%)<>A% PROCclose:ENDPROC
2090 VDU19,0,8;0;
2100 FOR P%=10 TO 200 STEP 10
2110 SOUND1,-15,P%,1:NEXT
2120 VDU19,0,4;0;
2130 VDU28,0,25,39,0,12
2140 PRINT TAB(15,8)"Well done"
2150 PRINT TAB(12,14);B%;" ";G$;" ";C%;
" = ";A%
2160 T%=TIME+100:REPEAT UNTIL TIME>T%
2170 VDU26:SC%=SC%+10:F%=TRUE
2180 ENDPROC
2190 :
2200 DEF PROCclose
2210 VDU28,0,25,39,0
2220 FOR P%=200 TO 10 STEP -10
2230 SOUND 1,-15,P%,1:VDU30,11
2240 NEXT
2250 PRINT TAB(17,10)"NO WAY!"
2260 PRINT TAB(13,14);B%;" ";G$;" ";C%;
" = ";A%
2270 SOUND0,-15,6,15
2280 T%=TIME+100:REPEAT UNTIL TIME>T%
2290 VDU26:F%=TRUE:ENDPROC

```

ELBUG BEEBUG ELBUG BEEBUG ELBUG BEEBUG ELBUG BEEBUG ELBUG BEEBUG ELBUG BEE

This is the penultimate issue of the ELBUG supplement. details will be included with the next (final) issue of the special arrangements we have made for transfer to membership of BEEBUG for those who wish to do so.