

Arcade Games

Cosmic Invaders

Here is a complete implementation of 'space invaders. Even though it is written in BASIC, and uses MODE 7 graphics, you'll be amazed to see how faithful it is to the arcade original.

Full instructions are included in the game, which was written by Mark Bucknell. There are a number of options you can trigger, such as 'freeze/unfreeze' and turning the sound on or off. The SHIFT key is used to fire.

```
10 REM      ****      Cosmic invaders      ****
20 REM      ****              by              ****
30 REM      ****      M.N.Buckwell      ****
40 REM      ****              (C) 1984      ****
50 DIM Scores%(9),Names$(9),Space%(7,
5),Bombs%(10,1),Alien0$(1),Alien1$(1),Al
ien2$(1), Ch% 3
60 MODE 7:ON ERROR GOTO 3240
70 VDU 23,1,0;0;0;0;
80 ENVELOPE 1,129,0,0,0,100,0,0,127,-
2,0,-1,126,120
90 ENVELOPE 2,1,1,0,0,200,1,0,126,-10
0,0,-126,126,126
100 ENVELOPE 3,1,-16,-8,-4,5,10,20,70,
0,0,-63,70,63
110 REM *** COLOUR CHARACTERS ***
120 L$=CHR$(141):R$=CHR$(129):G$=CHR$(
130)
130 Y$=CHR$(131):B$=CHR$(132):M$=CHR$(
133)
140 C$=CHR$(134):FL$=CHR$(136):ST$=CHR
$(137)
150 REM *** SPECIAL CHARACTERS ***
```

```

160 BL$=CHR$(255):REM Block
170 B$=CHR$(8):REM Cursor Back
180 F$=CHR$(9):REM Cursor Forward
190 D$=CHR$(10):REM Cursor Down
200 U$=CHR$(11):REM Cursor Up
210 REM *** GRAPHIC COLOUR CHAR'S ***
220 GR$=CHR$(145):GG$=CHR$(146):GY$=CH
R$(147):GC$=CHR$(150)
230 REM ***** Space Invader A *****
240 Alien0$(0)=">"+BL$+"m "+D$+B$+B$+B
$+B$+"k/7 "+D$+B$+B$+B$+B$+"! " " "
250 Alien0$(1)=">"+BL$+"m "+D$+B$+B$+B
$+B$+"k/7 "+D$+B$+B$+B$+B$+" [ch163] "
260 REM ***** Space Invader B *****
270 Alien1$(0)="ypv "+D$+B$+B$+B$+B$+"
m"+BL$+"> "+D$+B$+B$+B$+B$+"! " " "
280 Alien1$(1)="ypv "+D$+B$+B$+B$+B$+"
m"+BL$+"> "+D$+B$+B$+B$+B$+" [ch163] "
290 REM ***** Space Invader C *****
300 Alien2$(0)="ypv "+D$+B$+B$+B$+B$+"
m/> "+D$+B$+B$+B$+B$+"! " " "
310 Alien2$(1)="ypv "+D$+B$+B$+B$+B$+"
m/> "+D$+B$+B$+B$+B$+" [ch163] "
320 REM ***** Blank Invader *****
330 Blank1$="[spc4]"+D$+B$+B$+B$+B$+"[
spc4]"+D$+B$+B$+B$+B$+"[spc4]"
340 Blank2$="[spc5]"+D$+B$+B$+B$+B$+B$
+"[spc5]"+D$+B$+B$+B$+B$+B$+"[spc5]"
350 REM ***** Blank Invader Side *****
360 Blank_Line$=F$+F$+F$+" "+D$+B$+" "
+D$+B$+" "
370 REM ***** Your Base *****
380 Base$=" p^p "
390 REM ***** Sheild *****
400 Sheild$=BL$+BL$+BL$+D$+B$+B$+B$+BL
$+BL$+BL$+D$+B$+B$+B$+BL$+BL$+BL$
410 REM ***** Mystery *****
420 Mystery$=" ns="
430 PROC_Initialize

```

```

440 REM Main Program
450 REPEAT:finish=FALSE
460     PROC_Title:PROC_Time_key
470     IF A$=" " GOTO 510
480     IF A$="Q" OR A$="q" GOTO 540
490     PROC_Scores:PROC_Time_key
500     IF A$="Q" OR A$="q" GOTO 540
510     IF A$=" " THEN PROC_Play
520     PROC_High_Scores:PROC_Time_key
530     IF A$=" " GOTO 510
540 UNTIL A$="Q" OR A$="q"
550 CLS:END
560 DEF PROC_Title:CLS
570 PRINTL$;Y$[spc10]Cosmic Invaders"

580 PRINTL$;G$[spc10]Cosmic Invaders"

590 PRINT'R$"Instructions for use :-"
600 PRINTC$"[spc5]The object of the ga
me is to"
610 PRINTC$"shoot the cosmic aliens fr
om the sky to";
620 PRINTC$"save the world."
630 PRINTM$"Press the following keys f
or use :-"
640 PRINT'"[spc5]Z - To move left"
650 PRINT'"[spc5]X - To move right"
660 PRINT'"[spc5]SHIFT - To fire"
670 PRINT'"[spc5]ESCAPE - To exit from
game"
680 PRINT'"[spc5]F/U - Freeze/Un-Freez
e"
690 PRINT'Y$"Option :-"
700 PRINT'"[spc5]S/O - Sound on/off"
710 PRINT'G$"Press"FL$M$"Space"ST$G$"t
o start or 'Q' to quit"
720 ENDPROC
730 DEF PROC_Play
740 Score%=0:Base%=3

```

```

750 Start%=0
760 REPEAT:PROC_Setup_game
770     REPEAT:PROC_Key_Test:PROC_Update
_Invasders:IF NOT finish THEN PROC_Bombs
780     UNTIL finish=TRUE:Start%=Start%+
1
790 UNTIL P%<>7:Base%=Base%-1
800 IF Score%>Scores%(9) THEN PROC_Sco
re
810 ENDPROC
820 DEF PROC_Bombs
830 LOCAL D%:IF Bomb%<MBomb% THEN PROC
_New_Bomb
840 IF Bomb%=0 GOTO 910
850 FOR D%=1 TO MBomb%
860     IF Bombs%(D%,1)<>0 THEN PROC_Loo
k_Bomb
870 NEXT
880 FOR D%=1 TO MBomb%
890     IF Bombs%(D%,1)=23 THEN PRINTTAB
(BBombs%(D%,0),Bombs%(D%,1));" ";:Bombs%
(D%,1)=0:Bombs%=Bombs%-1
900 NEXT
910 ENDPROC
920 DEF PROC_Look_Bomb
930 PRINTTAB(Bombs%(D%,0),Bombs%(D%,1)
+1);
940 IF (FN_Find_Char<>32) AND ((Bombs%
(D%,1)+1>Y%+(R%+1)*3) OR (Bombs%(D%,0)<X
%) OR (Bombs%(D%,0)>=X%+(P%-O%+1)*4)) TH
EN PROC_Bomb_Hit ELSE PROC_Move_Bomb
950 ENDPROC
960 DEF PROC_Bomb_Hit
970 Bomb%=Bomb%-1
980 IF Bombs%(D%,1)+1=23 THEN PROC_Bom
b_Hit_Base :GOTO1040
990 PRINTTAB(Bombs%(D%,0),Bombs%(D%,1)
+1);:C%=FN_Find_Char
1000 IF C%=ASC(BL$) THEN PRINT"|";:GOTO

```

```

1040
1010 IF C%=ASC("|") THEN PRINT"p";:GOTO
1040
1020 IF C%=ASC("p") THEN PRINT" ";:GOTO
1040
1030 PRINT" ";
1040 PRINTTAB(Bombs%(D%,0),Bombs%(D%,1)
);" ";:Bombs%(D%,1)=0
1050 ENDPROC
1060 DEF PROC_Bomb_Hit_Base
1070 SOUND &13,-10,6,10
1080 PRINTTAB(B%+1,23);"[spc3]";
1090 B%=1
1100 Base%=Base%-1
1110 PRINTTAB(0,24);STRING$(20," ");
1120 IF Base%=0 THEN finish=TRUE ELSE P
ROC_Display_Base:PROC_Info_line
1130 FOR I%=1 TO 10000:NEXT
1140 ENDPROC
1150 DEF PROC_Move_Bomb
1160 PRINTTAB(Bombs%(D%,0),Bombs%(D%,1)
);" ";
1170 Bombs%(D%,1)=Bombs%(D%,1)+1
1180 IF Bombs%(D%,1)=23 THEN Bombs%(D%,
1)=0:Bomb%=Bomb%-1:GOTO 1200
1190 PRINTTAB(Bombs%(D%,0),Bombs%(D%,1)
);"#";
1200 ENDPROC
1210 DEF PROC_New_Bomb
1220 D%=-1:REPEAT:D%=D%+1:UNTIL Bombs%(
D%,1)=0 OR D%=MBomb%
1230 REPEAT:T%=RND(P%-O%)+O%:UNTIL Spac
e%(T%,5)
1240 J%=5:REPEAT:J%=J%-1:UNTIL Space%(T
%,J%)=TRUE
1250 Bombs%(D%,0)=X%+(T%-O%)*4+1:Bombs%
(D%,1)=Y%+(J%)*3+2:Bomb%=Bomb%+1
1260 ENDPROC
1270 DEF PROC_Score

```

```

1280 CLS:S%=10:REPEAT:S%=S%-1:UNTIL Sco
re%<=Scores%(S%) OR S%=0
1290 IF S%=8 THEN GOTO 1310
1300 FOR I%=S%+1 TO 8:Scores%(I%+1)=Sco
res%(I%):Names$(I%+1)=Names$(I%):NEXT
1310 Scores%(S%+1)=Score%
1320 PRINT'''G$" You are at number ";S%
+2;" in the top 10''''M$"Enter your nam
e"C$;
1330 INPUT Name$:IF LEN(Name$)>15 THEN
Name$=LEFT$(Name$,15)
1340 Name$(S%+1)=Name$
1350 ENDPROC
1360 DEF PROC_Key_Test
1370 IF INKEY(-98) AND NOT INKEY(-67) T
HEN PROC_Move_Left
1380 IF INKEY(-67) AND NOT INKEY(-98) T
HEN PROC_Move_Right
1390 IF INKEY(-1) AND NOT Missile% THEN
PROC_Fire
1400 IF INKEY(-68) THEN REPEAT UNTIL IN
KEY(-54)
1410 *FX 15 1
1420 ENDPROC
1430 DEF PROC_Start_Mystery
1440 MystX%=1:Mystery%=TRUE
1450 PRINTTAB(MystX%,1);Mystery$;
1460 ENDPROC
1470 DEF PROC_Move_Mystery
1480 SOUND &13,3,20,5
1490 MystX%=MystX%+1
1500 PRINTTAB(MystX%,1);Mystery$;
1510 IF MystX% = 35 THEN Mystery%=FALSE
:PRINTTAB(MystX%,1);"[spc4]";
1520 ENDPROC
1530 DEF PROC_Fire
1540 SOUND &0012,2,0,10:Missile%=TRUE:M
issileX%=B%+2:MissileY%=22:PROC_Move_Mis
sile(L%)

```



```

1550 ENDPROC
1560 DEF PROC_Move_Missile(L%)
1570 LOCAL I%,K%
1580 PRINTTAB(MissileX%,MissileY%);
1590 IF FN_Find_Char <> 32 THEN PROC_Hi
t:GOTO 1650
1600 PRINTTAB(MissileX%,MissileY%+1);
1610 IF MissileY%+1 <> 23 THEN PRINT " "

1620 PRINTTAB(MissileX%,MissileY%);
1630 IF MissileY%<>1 THEN PRINT"^"
1640 IF MissileY%>1 THEN MissileY%=Miss
i
l
e
Y
%
-
1
ELSE Missile%=FALSE
1650 ENDPROC
1660 DEF PROC_Hit
1670 SOUND &0010,-15,6,5:IF MissileY% <
22 THEN PRINTTAB(MissileX%,MissileY%+1)
; " "
1680 PRINTTAB(MissileX%,MissileY%);:C%=
FN_Find_Char
1690 IF (MissileY%+1<=Y%+(R%+1)*3) AND
(MissileX%>=X%) AND (MissileX%<X%+(P%-O%
+1)*4) THEN PROC_Invader_Hit(L%):GOTO172
0
1700 IF MissileY%=1 THEN PROC_Mystery_H
i
t:GOTO 1740
1710 PROC_Base_Hit
1720 I%=7
1730 IF NOT FN_Is_Line THEN finish=TRUE

1740 ENDPROC
1750 DEF PROC_Mystery_Hit
1760 PRINTTAB(MystX%,1); "[spc5]";
1770 Score%=Score%+RND(30)*10
1780 PROC_Info_line
1790 ENDPROC
1800 DEF PROC_Base_Hit
1810 IF C%=ASC(BL$) THEN PRINT"/";:GOTO
1840

```

```

1820 IF C%=ASC("/") THEN PRINT"[ch163]"
;:GOTO 1840
1830 PRINT " ";
1840 Missile%=FALSE
1850 ENDPROC
1860 DEF PROC_Invader_Hit(L%)
1870 LOCAL I%,K%
1880 IF Left% THEN G%=-1 ELSE G%=1
1890 IF MissileX% > X%+(L%-O%+1)*4 THEN
I%=(MissileX%-X%+(O%*4)+G%)DIV4 ELSE I%
=(MissileX%-X%+O%*4)DIV4
1900 K%=(MissileY%-Y%) DIV 3:Space%(I%,
K%)=FALSE:J%=(I%-O%)*4:PROC_Blank(J%)
1910 IF NOT FN_Is_Line THEN Space%(I%,5
)=FALSE:PROC_Boundary_Update
1920 K%=(MissileY%-Y%) DIV 3
1930 IF NOT FN_Is_Row THEN Space%(7,K%)
=FALSE:PROC_Boundary_Update
1940 Missile%=FALSE:Score%=Score%+((3-(
(MissileY%-Y%)DIV3)DIV2)*20):PROC_Info_1
ine
1950 ENDPROC
1960 DEF PROC_Boundary_Update
1970 S%=O%:PROC_X_Boundary_Update:PROC_
Y_Boundary_Update:X%=X%+(O%-S%)*4
1980 ENDPROC
1990 DEF PROC_X_Boundary_Update
2000 IF Space%(O%,5) GOTO 2020
2010 REPEAT:O%=O%+1:UNTIL Space%(O%,5)
2020 IF P%=7 GOTO 2060
2030 IF Space%(P%,5) GOTO 2060
2040 REPEAT:P%=P%-1:IF P%<0 THEN finish
=TRUE:P%=7
2050 UNTIL Space%(P%,5)
2060 ENDPROC
2070 DEF PROC_Y_Boundary_Update
2080 IF Space%(7,Q%) GOTO 2100
2090 REPEAT:Q%=Q%+1:UNTIL Space%(7,Q%)
2100 IF R%=5 GOTO 2140

```

```

2110 IF Space%(7,R%) GOTO 2140
2120 REPEAT:R%=R%-1:IF R%<0 THEN finish
=TRUE:R%=5
2130 UNTIL Space%(7,R%)
2140 ENDPROC
2150 DEF PROC_Move_Left
2160 IF B%>1 THEN B%=B%-1
2170 PROC_Display_Base
2180 ENDPROC
2190 DEF PROC_Move_Right
2200 IF B%<34 THEN B%=B%+1
2210 PROC_Display_Base
2220 ENDPROC
2230 DEF PROC_Setup_game
2240 CLS
2250 O%=0:P%=6:Q%=0:R%=4:B%=1:Bomb%=0:M
Bomb%=Start%+6:FOR I%=0 TO 10:Bombs%(I%,
1)=0:NEXT
2260 Mystery%=FALSE:Missile%=FALSE:Miss
ileX%=0:MissileY%=0:finish=FALSE:VDU 23,
1,0;0;0;0;
2270 PROC_Info_line:PROC_Colour_setup:P
ROC_Invader_setup:PROC_Display_Sheilds:P
ROC_Display_Invaders:PROC_Display_Base
2280 ENDPROC
2290 DEF PROC_Display_Base
2300 PRINTTAB(B%,23);Base$
2310 ENDPROC
2320 DEF PROC_Display_Sheilds
2330 FOR I%= 0 TO 3:PRINTTAB(5+I%*9,20)
;Sheild$:NEXT
2340 ENDPROC
2350 DEF PROC_Invader_setup
2360 FOR I%= 0 TO 7:FOR J%= 0 TO 5:Spac
e%(I%,J%)=TRUE:NEXT:NEXT
2370 X%=10:Y%=Start%MOD4 + 2:Left%=TRUE

2380 ENDPROC
2390 DEF PROC_Display_Invaders

```

```

2400 SOUND 1,-15,1,1:I%=O%:M%=ABS((X%+I
%*4)MOD 2)
2410 REPEAT:PROC_Key_Test:J%=(I%-O%)*4
2420   FOR K%= Q% TO R%:IF Space%(I%,K%
) THEN PROC_Invader ELSE PROC_Blank(J%)
2430   NEXT
2440   IF Missile% THEN PROC_Move_Missi
le(I%)
2450   IF Mystery% THEN PROC_Move_Myste
ry ELSE IF RND(100)=1 THEN PROC_Start_My
stery
2460 I%=I%+1:UNTIL I%>P% OR finish=TRUE

2470 ENDPROC
2480 DEF PROC_Update_Invaders
2490 IF Left% THEN X%=X%-1 ELSE X%=X%+1

2500 IF NOT Left% THEN PRINTTAB(X%-1,Y%
+Q%*3);STRING$((R%-Q%+1)*3," "+D$+B$);
2510 PROC_Display_Invaders
2520 IF X%<=1 THEN Left%=FALSE:PROC_Mov
e_Down
2530 IF X%+(P%-O%+1)*4>=39 THEN Left%=T
RUE:PROC_Move_Down
2540 ENDPROC
2550 DEF PROC_Move_Down
2560 PRINTTAB(X%,Y%+Q%*3);STRING$((P%-O
%+1)*4," ");:Y%=Y%+1:IF Y%-(4-R%)*3=9THE
N finish=TRUE
2570 ENDPROC
2580 DEF FN_Is_Line
2590 L%=0:FOR K%=0 TO 4:IF NOT Space%(I
%,K%) THEN L%=L%+1
2600   NEXT:IF L%= 5 THEN =FALSE ELSE
=TR
UE
2610 DEF FN_Is_Row
2620 L%=0:FOR I%=0 TO 6:IF NOT Space%(I
%,K%) THEN L%=L%+1

```

```

2630 NEXT:IF L%= 7 THEN =FALSE ELSE =TRUE
2640 DEF PROC_Invader
2650 PRINTTAB(X%+J%,Y%+K%*3);
2660 ON K%+1 GOTO 2670,2670,2680,2680,2
690,2700
2670 PRINT Alien2$(M%):GOTO 2710
2680 PRINT Alien1$(M%):GOTO 2710
2690 PRINT Alien0$(M%):GOTO 2710
2700 finish=TRUE
2710 ENDPROC
2720 DEF PROC_Blank(J%)
2730 IF NOT Left% THEN J%=J%-1
2740 PRINTTAB(X%+J%,Y%+K%*3);:IF Left%
THEN PRINTBlank1$ ELSEPRINTBlank2$
2750 ENDPROC
2760 DEF PROC_Colour_setup
2770 PRINTTAB(0,1);GR$;
2780 FOR I%= 2 TO 7:PRINTTAB(0,I%);GC$;
TAB(0,I%+6);GR$;TAB(0,I%+12);GY$;:NEXT
2790 FOR I%= 0 TO 2:PRINTTAB(0,I%+20);G
G$;:NEXT:PRINT'GR$;
2800 ENDPROC
2810 DEF PROC_Info_line
2820 PRINTTAB(0,0)G$"HS"M$;Scores%(0);
2830 PRINTTAB(20,0)G$"Score"M$;Score%;
2840 PRINTTAB(0,24);CHR$(150);
2850 FOR I%= 1 TO Base%:PRINTBase$;:NEX
T
2860 ENDPROC
2870 DEF PROC_Scores
2880 CLS
2890 PRINTCHR$(141)CHR$(130)"[spc3]Scor
es given :-"
2900 PRINTCHR$(141)CHR$(130)"[spc3]Scor
es given :-"
2910 FOR I%= 4 TO 24:PRINT TAB(0,I%);CH
R$(150);:NEXT
2920 PRINTTAB(5,6);Mystery$;TAB(20,6);C

```

```

HR$(133); "????";
2930 PRINTTAB(6,10);Alien2$(0);TAB(20,1
1);CHR$(133); "60"
2940 PRINTTAB(6,15);Alien1$(0);TAB(20,1
6);CHR$(133); "40"
2950 PRINTTAB(6,20);Alien0$(0);TAB(20,2
1);CHR$(133); "20"
2960 ENDPROC
2970 DEF PROC_High_Scores
2980 CLS
2990 PRINTCHR$(141)CHR$(130) "[spc10]Hig
h Scores"
3000 PRINTCHR$(141)CHR$(130) "[spc10]Hig
h Scores"
3010 FOR I%= 0 TO 9
3020     PRINT'CHR$(134);I%+1;TAB(4);Name
$(I%);TAB(30);Scores%(I%)
3030 NEXT
3040 ENDPROC
3050 DEF PROC_Initialize
3060 *FX210 0
3070 FOR I%= 0 TO 9
3080     Names$(I%)="Mark beat you !!!"
3090     Scores%(I%)=10000-(I%*1000)
3100 NEXT
3110 ENDPROC
3120 DEF PROC_Time_key
3130 TIME=0
3140 REPEAT
3150     A$=INKEY$(0)
3160     IF A$="s" OR A$="S" THEN *FX 210
0
3170     IF A$="o" OR A$="O" THEN *FX 210
1
3180     UNTIL A$="Q" OR A$="q" OR A$=" "
OR TIME = 1000
3190 ENDPROC
3200 DEF FN_Find_Char
3210 A%=&87

```

```
3220    !Ch%=USR(&FFF4)
3230    = Ch%?1
3240    REM **** Error Routine ****
3250    IF ERR = 17 GOTO 440
3260    REPORT:PRINT" in line ";ERL
3270    END
```

Cosmos

In this exciting game you are flying over a moving landscape which produces a very good 3-D effect. You have to try and shoot as many alien birds as possible within a limited time. Full instructions are within the game.

You move your sights around the galaxy until you spot one of the birds. The birds will try to evade your lethal weaponry.

```
10 REM COSMOS
30 MODE 7
40 PROC_INSTR
50 MODE 2
60 PROC_INIT
70 REPEAT
80     PROC_NEWGAME
90     PROC_PLAY
100 UNTIL NOT FN_ANOTHER
110 MODE 7
120 END
130 DEF PROC_INIT
140 VDU 23,0,10,32,0;0;0;
150 VDU 23,128,&31,&4B,&85,&87,&06,&03
, &04,&08
160 VDU 23,129,&BC,&D2,&A1,&E1,&60,&C0
, &20,&10
170 ENVELOPE 1,1,-16,-8,-4,5,10,20,126
,0,0,-63,126,63
180 ENVELOPE 2,1,-2,4,-2,2,2,2,126,0,0
,-126,126,100
190 HIGH=0:H$=" "
200 FOR W%=0 TO 15
210     VDU 19,W%+(W%=7),0,0,0;
220 NEXT
```



```

230 VDU 19,7,2,0,0,0
240 COLOUR 1:W%=14
250 ENDPROC
260 DEF PROC_ALTCOLS
270 SOUND &13,-10,3,10
280 *FX 19
290 VDU 19,W%,4,0,0;
300 W%=W%-1:IF W%<0 THEN W%=14
310 IF W%=7 THEN W%=6
320 VDU 19,W%,6,0,0;
330 ENDPROC
340 DEF PROC_LANSCAPE
350 COLOUR 7:COLOUR 143:CLS
360 PROC_SHIP
370 PRINT TAB(0,0);"Energy 1000 Hits 0
"
380 offset1=128:C%=14
390 MOVE 0,512:DRAW 1279,512
400 FOR col=5 TO 1 STEP -0.05
410     PROC_ALTCOLS
420     GCOL 0,128+C%
430     offset2=offset1:offset1=636/col
440     VDU 24,0;636-offset1;1279;636-of
fset2;16,26
450     C%=C%-1:IF C%=7 THEN C%=6
460     IF C%=-1 THEN C%=14
470 NEXT
480 GCOL 3,8
490 MOVE 640,508:MOVE 500,4:PLOT 85,78
0,4
500 PROC_SHIP
510 GCOL 3,8
520 ENDPROC
530 DEF PROC_SHIP
540 GCOL 0,7:GCOL 0,135
550 MOVE 0,50:MOVE 0,0:PLOT 85,100,100

560 MOVE 1279,50:MOVE 1279,0:PLOT 85,1
179,100

```

```

570 VDU 24,0;0;1279;26;16,26
580 PX%=9:PY%=13
590 PROC_SIGHT
600 ENDPROC
610 DEF PROC_NEWGAME
620 E%=1000:H%=0:B%=0
630 PX%=9:PY%=13:SX%=PX%:SY%=PY%
640 BX%=RND(30)-5:BY%=RND(5)
650 GCOL 4,0
660 PROC_LANSCAPE
670 PRINT TAB(0,10);"Press RETURN to p
lay"
680 *FX15
690 REPEAT UNTIL GET=13
700 REPEAT UNTIL NOT INKEY(-74)
710 PRINT TAB(0,10);SPC(20)
720 ENDPROC
730 DEF PROC_FIRE
740 VDU 19,15,4,0,0;
750 TX%=64*(SX%+1):TY%=1007-(32*SY%)
760 MOVE 100,100:DRAW TX%,TY%:DRAW 117
9,100
770 SOUND &11,1,20,5
780 MOVE 100,100:DRAW TX%,TY%:DRAW 117
9,100
790 VDU 19,15,0;0;
800 E%=E%-5
810 IF SX%=BX% AND SY%=BY% THEN PROC_H
IT
820 PRINT TAB(7,0);E%;" ";
830 ENDPROC
840 DEF PROC_HIT
850 VDU 19,15,1,0,0,0
860 SOUND &12,2,2,5
870 PROC_SIGHT
880 VDU 19,15,0;0;
890 H%=H%+1:B%=B%+1
900 IF B%=10 THEN B%=0:E%=E%+100
910 PRINT TAB(17,0);H%

```

```

920 BX%=RND(30):BY%=-RND(5)
930 ENDPROC
940 DEF PROC_BOMB
950 MOVE BX%*64+64,1000-(BY%*32):DRAW6
40,512
960 VDU 23,0,13,1,0;0;0;
970 SOUND &12,1,20,5
980 MOVE BX%*64+64,1000-(BY%*32):DRAW6
40,512
990 E%=E%-10
1000 PRINT TAB(7,0);E%;" "
1010 VDU 23,0,13,0,0;0;0;
1020 ENDPROC
1030 DEF PROC_SIGHT
1040 PRINT TAB(PX%,PY%);
1050 IF PY%=15 THEN PRINT"__" ELSE PRIN
T"  "
1060 PRINT TAB(SX%,SY%);"[]"
1070 PX%=SX%:PY%=SY%
1080 ENDPROC
1090 DEF PROC_BIRD
1100 IF BX%<0 OR BX%>18 OR BY%<1 OR BY%
>15 THEN VDU 19,7,2,0,0,0:GOTO 1130
1110 IF NOT(BX%<14 AND BX%>5 AND BY%>10
AND BY%<16) THEN VDU 19,7,3,0,0,0
1120 IF BY%=15 THEN PRINT TAB(BX%,BY%);
"__" ELSE PRINT TAB(BX%,BY%);" "
1130 IF ABS(BX%-SX%)<3 AND ABS(BY%-SY%)
<2 THEN BX%=BX%+SGN(BX%-SX%):BY%=BY%+SGN
(BY%-SY%)
1140 IF RND(3)<3 THEN BX%=BX%+(BX%>8)-(
BX%<12) ELSE BX%=BX%+RND(3)-2
1150 IF RND(3)<3 THEN BY%=BY%+(BY%>12)-
(BY%<15) ELSE BY%=BY%+RND(3)-2
1160 IF BY%>15 THEN BY%=15
1170 IF BX%>18 OR BX%<0 THEN ENDPROC
1180 IF BY%<1 THEN ENDPROC
1190 PRINT TAB(BX%,BY%);"[fs0][fs1]"
1200 IF BX%<14 AND BX%>5 AND BY%>10 AND

```

```

BY%<16 THEN VDU 19,7,1,0,0;
1210 IF BX%<12 AND BX%>7 AND BY%>12 AND
BY%<15 THEN PROC_BOMB
1220 ENDPROC
1230 DEF PROC_PLAY
1240 REPEAT
1250     PROC_ALTCOLS
1260     SX%=SX%+(INKEY(-98) AND SX%>0)-(
INKEY(-67) AND SX%<18)
1270     SY%=SY%-(INKEY(-105) AND SY%<15)
+(INKEY(-73) AND SY%>1)
1280     PROC_SIGHT
1290     IF INKEY(-74) PROC_FIRE ELSE PRO
C_DELAY
1300     PROC_BIRD
1310     E%=E%+1
1320     PRINT TAB(7,0);E%
1330 UNTIL E%<=0
1340 *FX 15
1350 ENDPROC
1360 DEF PROC_DELAY
1370 TIME=0:REPEAT UNTIL TIME>3
1380 ENDPROC
1390 DEF PROC_INSTR
1400 PRINT TAB(12,1);CHR$141;"[fs1]C[fs
2]O[fs3]S[fs4]M[fs5]O[fs6]S"
1410 PRINT TAB(12,2);CHR$141;"[fs1]C[fs
2]O[fs3]S[fs4]M[fs5]O[fs6]S"
1420 PRINT'" "[fs6]The aim of the game
is to zap as many"
1430 PRINT"[fs6]bird like aliens as pos
sible using as"
1440 PRINT"[fs6]little energy as possib
le. The aliens"
1450 PRINT"[fs6]will try to fire at yo
ur Main Energy"
1460 PRINT"[fs6]Reserves down the middl
e of your Ship."
1470 PRINT"[fs6]If they succeed it will

```

```

drain 10 Units"
1480 PRINT"[fs6]from your reserves. Each time you fire"
1490 PRINT"[fs6]it will take 5 Units of energy but the"
1500 PRINT"[fs6]energy reserves are being built up all"
1510 PRINT"[fs6]the time."
1520 PRINT'"[fs1]The controls-"
1530 PRINT"[fs3]Z - left X - right"
1540 PRINT"[fs3]/ - down : - up[spc5]RETURN - fire"
1550 PRINT'"[fs2]An extra 100 Units of energy is awarded[fs2]every 10 aliens hit.[fs7]"
1560 PRINT'TAB(5);"[fs1]Press[fs8]SPACE BAR[fs9]to continue";
1570 *FX 15
1580 REPEAT UNTIL GET=32
1590 ENDPROC
1600 DEF FN_ANOTHER
1610 *FX 202 32
1620 PRINT TAB(7,0);"0 "
1630 VDU 28,0,14,19,2,12
1640 PRINT" You have run outof energy and have hit";
1650 PRINT ;H%;" aliens."'
1660 FOR W=1 TO 100
1670 REPEAT UNTIL INKEY(-147)
1680 NEXT
1690 IF HIGH<=H% THEN HIGH=H%:INPUT"This is a high scoreso please enter yourname : "HS$
1700 PRINT'"High score ";H%
1710 PRINT "by ";HS$
1720 PRINT'"Another go ?";
1730 REPEAT KEY$=GET$
1740 UNTIL KEY$="Y" OR KEY$="y" OR KEY$="N" OR KEY$="n"

```

```
1750 VDU 26,12
1760 IF KEY$="Y" OR KEY$="y" THEN =TRUE
ELSE =FALSE
```

Meteoroids

Here's a neat little program in which you must pilot your V - Wing spacecraft through a field of asteroids. You use the ' X' and ' Z' keys to move yourself right and left. At the end of the game (when you hit an asteroid) you'll be told the number of ' light years' you have managed to traverse.

```
10 REM METEORIDS
20
30 MODE 7
40 REPEAT PROC_NEWGAME
50     PROC_PLAY
60     UNTIL NOT FN_ANOTHER
70 MODE 7
80 END
90
100 DEF PROC_NEWGAME
110 VDU 23,0,10,32,0;0;0;
120 X%=20:D%=0
130 HIT=FALSE
140 CLS
150 PROC_METEORS
160 PRINT TAB(X%,0); "V"
170 PRINT TAB(10,15); "[f s1]PRESS ANY K
EY TO PLAY";
180 *FX 15 1
190 A=GET
200 PRINT TAB(10,15); SPC(22);
210 ENDPROC
220 DEF PROC_METEORS
230 PRINT TAB(0,24) 'CHR$(128+RND(7));
240 FOR A%=1 TO 4
250     PRINT TAB(RND(38),24); "*" ;
```

```

260     NEXT
270 ENDPROC
280 DEF PROC_PLAY
290 REPEAT PROC_METEORS
300     X%=X%+( (INKEY-98 AND X%>1)-(INKE
Y-67 AND X%<38))
310     PRINT TAB(X%,0);
320     A%=135:IF(USR(&FFF4)AND&FFFF)DIV
&100<>32 THEN HIT=TRUE
330     *FX 19
340     PRINT"V";
350     D%=D%+1
360     PROC_DELAY(D%)
370     UNTIL HIT
380 PRINT TAB(X%-4,0);"[fs8][fs1]B[fs2
]A[fs3]N[fs4]G[fs9]"
390 ENDPROC
400 DEF PROC_DELAY(d%)
410 FOR W%=1 TO 500-d%
420     NEXT
430 ENDPROC
440 DEF FN_ANOTHER
450 PRINT"' '[fs1]YOU[fs2]TRAVELLED[f
s3]";D%;"[fs7]LIGHT[fs6]YEARS"
460 PRINT"' '[fs8][fs6]ANOTHER GO?[fs9]
";
470 *FX 15 1
480 REPEAT KEY$=GET$
490     UNTIL KEY$="Y" OR KEY$="N"
500 IF KEY$="Y" THEN =TRUE ELSE =FALSE

```


Light Cycles

In this fast-moving game you have to ride your light cycle around the screen avoiding the computer's cycle and making sure you don't run over the trail left by either your own bike or the computer's. The aim of the game is to enclose the computer so that it cannot move. It, of course, is trying to do the same thing to you.

You control your cycle as follows:

: to move up
/ to move down
Z to move left
X to move right

If you hold down the SHIFT key, your cycle will move twice as fast, which gives you the chance of riding through trails. However, this option should not be tried lightly because it is programmed to work only around 50 per cent of the time.

Light Cycles was written by Neal Cavalier-Smith.

```
10 REM LIGHT CYCLES
20 REM By Neal Cavalier-Smith.
30 MODEL
40 PROCsetup
50 PROCboard
60 REPEAT
70   PROCmove
80   UNTIL FALSE
90 DEFPROCsetup
100 F%=FALSE
110 N$="HUMANOID": REM or your name!
120 SC%=0
```

```

130 VDU19,1,4,0,0,0
140 VDU23,0,10,32,0;0;0;
150 VDU19,2,1,0,0,0
160 VDU19,3,3,0,0,0
170 VDU23,130,7,63,79,187,191,219,102,
56
180 VDU23,131,240,220,254,98,89,217,22
6,12
190 VDU23,132,60,60,126,126,60,0,0,0
200 VDU23,133,0,24,60,126,126,126,126,
60
210 DIMN(2)
220 N(1)=32
230 N(2)=-32
240 DIMC$(2,2)
250 C$(0,1)=CHR$(8)+CHR$(8)+CHR$(130)+
CHR$(131)
260 C$(1,0)=CHR$(132)+CHR$(11)+CHR$(8)
+CHR$(133)
270 C$(1,2)=CHR$(132)+CHR$(10)+CHR$(8)
+CHR$(133)
280 C$(2,1)=CHR$(9)+CHR$(130)+CHR$(131
)
290 ENDPROC
300 DEFPROCboard
310 VDU 26,12,24,0;0;1279;908;
320 COLOUR129:GCOL0,0
330 CLS
340 FORX=4TO1279 STEP 181
350     MOVEX,4
360     DRAWX,904
370     MOVE4,X
380     DRAW1271,X
390     NEXT
400 COLOUR2
410 PRINTTAB(0,2);"SCORE..."
420 PRINTTAB(0,1);N$
430 COLOUR3
440 PRINTTAB(23,1);"COMPUTER"

```

```

450 VDU5
460 S=0
470 X%=640
480 Y%=512
490 T%=640
500 U%=512
510 M%=0
520 N%=0
530 R%=32
540 S%=0
550 ENDPROC
560 DEFPROCmove
570 PROCread
580 PROCplay1
590 IFINKEY(-1) THENR%=2*R%:S%=S%*2
600 ENDPROC
610 DEFPROCcomp
620 GCOL0,3
630 IFPOINT(T%+M%,U%+N%)=-1THENPROCcal
C
640 MOVET%,U%
650 T%=T%+M%
660 U%=U%+N%
670 DRAWT%,U%
680 GCOL3,2
690 MOVET%-16,U%+16
700 PRINTC$(1+SGN(M%),1+SGN(N%))
710 MOVET%-16,U%+16
720 PRINTC$(1+SGN(M%),1+SGN(N%))
730 ENDPROC
740 DEFPROCplay1
750 SC%=SC%+1
760 VDU 4:PRINT TAB(10,2);SC%;" ";:VD
U 5
770 GCOL0,2
780 MOVEX%,Y%
790 X%=X%+R%
800 Y%=Y%+S%
810 P%=POINT(X%,Y%)

```

```

820  IFP%=-1ORP%>1THENPROCblam
830  DRAWX%,Y%
840  GCOL3,3
850  MOVEX%-16,Y%+16
860  PRINTC$(1+SGN(R%),1+SGN(S%))
870  PROCcomp
880  PROCcalc
890  GCOL3,3
900  MOVEX%-16,Y%+16
910  PRINTC$(1+SGN(R%),1+SGN(S%))
920  ENDPROC
930  DEFPROCcalc
940  IFPOINT(T%+M%,U%+N%)=1THENENDPROC
950  TRY%=0
960  N=N(RND(2))
970  TRY%=TRY%+1
980  P%=POINT(T%+N,U%)
990  IFP%=-1THEN1010
1000 IFP%>1THEN1010ELSEM%=N:N%=0:ENDPRO
C
1010 P%=POINT(T%,U%+N)
1020 IFP%=-1 AND TRY%<10THEN960
1030 IFP%>1THEN1040ELSEN%=N:M%=0:ENDPRO
C
1040 IF TRY%<10 THEN 960
1050 F%=TRUE
1060 PROCblam
1070 ENDPROC
1080 DEFPROCread
1090 IFINKEY(-147)THENENDPROC
1100 R%=32*(INKEY(-98)-INKEY(-67))
1110 IFINKEY(-1)THENR%=2*R%
1120 IFR%THENS%=0:ENDPROC
1130 S%=32*(INKEY(-105)-INKEY(-73))
1140 ENDPROC
1150 DEFPROCblam
1160 CLG
1170 VDU 4
1180 PRINT TAB(0,10);

```

```
1190 IF F% THEN SOUND 0,-10,6,10 ELSE S
OUND 0,-10,4,10
1200 IF F% THEN PRINT''' "You won Wretch
!''' ELSE PRINT''' "Ha Ha I beat you'''
1210 PRINT''' "Another Go?";
1220 REPEAT K$=GET$
1230 UNTIL K$="Y" OR K$="N"
1240 IF K$="Y" THEN RUN
1250 VDU 26,12,4
1260 END
```

Ghost Lane

In this game, based loosely on the popular arcade game in which you have to get a frog across a busy highway, you have to help a number of ghosts (who have wandered out of yet another popular arcade game) across four lanes of traffic.

```
10 REM Ghost Lane
20
30 MODE 5
40 PROC_INIT
50 REPEAT
60     PROC_NEW_GAME
70     PROC_PLAY
80     UNTIL NOT FN_ANOTHER
90 END
100 DEF PROC_INIT
110 VDU 23,0,10,32,0;0;0;
120 VDU 23,128,56,124,214,214,254,254,
170,170
130 VDU 23,129,0,238,68,255,255,68,238
,0
140 VDU 23,130,0,119,34,255,255,34,119
,0
150 VDU 23,131,145,82,0,192,3,0,74,137

160 VDU 23,132,-1,-1,-1,-1,-1,-1,-1,-1

170 VDU 19,3,2,0,0,0
180 ENDPROC
190 DEF PROC_NEW_GAME
200 A$="[fs1] [fs1][fs1] [fs1] [fs1] [
fs1] [fs1][fs1] [fs1] [fs1] "
210 B$="[fs1] [fs1][fs1] [fs1][spc3]
```

```

[fs1]   [fs1]   [fs1]   "
    220 C$="[fs4][fs4] [fs2] [fs2] [fs2]
[fs2]   [fs2]   [fs2][spc5]"
    230 D$="[fs2] [fs2][fs2] [fs2] [fs2][f
s2][spc3][fs4][fs4] [fs2][fs2][spc3]"
    240 COLOUR 131
    250 VDU 26,12,28,0,18,19,5
    260 COLOUR 128:VDU 12,26
    270 PA%=8:PD%=18
    280 PAE%=8:PDE%=18
    290 L%=5:S%=0
    300 COLOUR 131:COLOUR 1
    310 PRINT TAB(0,20);"LIVES  ";TAB(8,20
);"[fs0] [fs0] [fs0] [fs0]"
    320 ENDPROC
    330 DEF PROC_PLAY
    340 COLOUR 128
    350 REPEAT
    360     PAE%=PA%:PDE%=PD%
    370     IF INKEY(-67) THEN PA%=PA%+1:IF
PA%>19 THEN PA%=19
    380     IF INKEY(-98) THEN PA%=PA%-1:IF
PA%<0 THEN PA%=0
    390     IF INKEY(-105) THEN PD%=PD%+1:IF
PD%>18 THEN PD%=18
    400     IF INKEY(-73) THEN PD%=PD%-1:IF
PD%<4 THEN PD%=4
    410     COLOUR 1
    420     PRINT TAB(PAE%,PDE%);" ";TAB(PA%
,PDE%);"[fs0]";
    430     Y%=PA%+1
    440     COLOUR 2
    450     A$=RIGHT$(A$,1)+LEFT$(A$,19)
    460     B$=RIGHT$(B$,1)+LEFT$(B$,19)
    470     C$=MID$(C$,2)+LEFT$(C$,1)
    480     D$=MID$(D$,3)+LEFT$(C$,2)
    490     PRINT TAB(0,6);A$;TAB(0,9);B$;TA
B(0,13);C$;TAB(0,16);D$
    500     IF PD%=6 AND MID$(A$,Y%,1)<>" "

```

```

THEN PROC_HIT
  510   IF PD%=9 AND MID$(B$,Y%,1)<>" "
THEN PROC_HIT
  520   IF PD%=13 AND MID$(C$,Y%,1)<>" "
THEN PROC_HIT
  530   IF PD%=16 AND MID$(D$,Y%,1)<>" "
THEN PROC_HIT
  540   IF PD%=4 THEN PROC_HOME
  550   FOR D%=0 TO 50:NEXT
  560   UNTIL L%=0
  570 ENDPROC
  580 DEF PROC_HIT
  590 COLOUR 1
  600 PRINT TAB(PA%,PD%);" [fs3] "
  610 SOUND 0,-10,4,1
  620 TIME=0:REPEAT UNTIL TIME>100
  630 L%=L%-1
  640 PROC_NEXT_ATTEMPT
  650 ENDPROC
  660 DEF PROC_HOME
  670 S%=S%+1
  680 COLOUR 1:COLOUR 131
  690 PRINT TAB(0,0);STRING$(S%," [fs0] "
);TAB(9,0);"SAFELY HOME"
  700 PRINT TAB(2,2);"SCORE IS ";100*S%
  710 PROC_NEXT_ATTEMPT
  720 ENDPROC
  730 DEF PROC_NEXT_ATTEMPT
  740 COLOUR 1
  750 PRINT TAB(PA%,PD%);" ";
  760 PA%=8:PD%=18
  770 COLOUR 131:COLOUR 1
  780 IF L%=0 THEN PRINT TAB(8,20);" " E
LSE PRINT TAB(8,20);STRING$(L%-1," [fs0]
");" ";
  790 COLOUR 128
  800 ENDPROC
  810 DEF FN_ANOTHER
  820 *FX 15 1

```



```
830 COLOUR 1:COLOUR 131
840 PRINT TAB(0,22);"There are no more
ofyou left!"
850 PRINT'"Another Go ?";
860 REPEAT K$=GET$
870 UNTIL K$="Y" OR K$="N"
880 =(K$="Y")
```

Bomb Raid

In this program, based on one by Gwyn Dewey, you are in control of an aeroplane which has to make an emergency landing. Unfortunately, in order to land you must destroy the buildings which are in your way.

There are full instructions within the program. Here' s the listing so you can throw your weight (and a few bombs) about.

```
10 REM BOMB RAID
20
30 MODE 5
40 PROCINIT
50 REPEAT
60     PROCINST
70     REPEAT
80         PROCBUILD
90         PROCPLAY
100        UNTIL FLAG%
110        UNTIL NOT FNAGAIN
120 MODE 7
130 END
140 DEF PROCINIT
150 DIM B%(20),S(20),COL%(1,5)
160 DATA 0,1,1,4,0,5,0,4,1,5,4,5
170 FOR AC=0 TO 5
180     FOR LC=0 TO 1
190         READ COL%(LC,AC)
200         NEXT LC
210     NEXT AC
220 VDU 19,2,6,0,0;19,3,2,0,0;
230 FOR LC=0 TO 1:VDU19,LC,COL%(LC,0),
0,0;:NEXT
240 VDU 23,0,10,32;0;0;0
```

```

250 PROCCHARS
260 HIGH%=0
270 ENDPROC
280 DEF PROCCHARS
290 VDU 23,128,254,146,146,254,254,146
,146,254
300 VDU 23,129,0,0,0,0,0,56,124,254
310 VDU 23,130,254,146,146,254,254,146
,146,254
320 VDU 23,131,254,146,146,254,254,146
,146,242
330 VDU 23,132,252,73,145,-1,253,249,3
2,248
340 VDU 23,133,193,224,240,-1,127,63,3
3,195
350 VDU 23,134,-1,-1,-1,-1,-1,-1,-1
360 VDU 23,135,40,56,16,56,56,56,56,16
370 ENDPROC
380 DEF PROCBUILD
390 COLOUR 0
400 WA%=WA%+1
410 FOR LC=0 TO 1
420     VDU19,LC,COL%(LC,(WA%-1) MOD 6),
0,0,0
430     NEXT
440 VDU 17,130,12,17,131,28,0,31,19,30
,12,26,17,3,17,130
450 FOR A%=3 TO 17
460     H%=RND(13)
470     COLOUR RND(2)-1
480     PRINT TAB(A%,29); "[fs3]"
490     FOR H=28 TO (28-H%) STEP-1
500         PRINT TAB(A%,H); "[fs2]"
510         NEXT H
520     COLOUR RND(2)-1
530     PRINT TAB(A%,H); "[fs1]"
540     B%(A%)=H%+2
550     S(A%)=150/(B%(A%)+1)
560     NEXT A%

```

```

570 COLOUR 1:PRINT TAB(0,0);"SC ";INT(
S+.9)
580 PRINT TAB(10,0);"HI ";HIGH%
590 N%=N%-50
600 ENDPROC
610 DEF PROCPLAY
620 F%=0:Y%=0:X1%=0:Y1%=1
630 COLOUR 1:COLOUR 130
640 REPEAT Y%=Y%+1
650     X%=0
660     REPEAT X%=X%+1
670         IF F%=1 THEN PROCBOMB:GOTO690
680         IF INKEY(-99) THEN X1%=X% MOD2
0:Y1%=Y%:F%=1:PROCBOMB
690     COLOUR 1:PRINT TAB(X%-1,Y%);"
[fs5][fs4]"
700     FOR E%=0 TO N%:NEXT
710     SOUND &10,-10,15,1:SOUND &11,0
,200,1
720     C%=FNCODE(X%+2,Y%)
730     IF C%>128 AND C%<134 THEN FLAG
%=TRUE ELSE FLAG%=FALSE
740     UNTIL X%>19 OR FLAG%
750     PRINT TAB(18,Y%);" ";
760     UNTIL (X%>19 AND Y%>27) OR FLAG%
770 IF FLAG% THEN PROCCRASH:ENDPROC
780 PROCTUNE
790 S=S+250
800 ENDPROC
810 DEF PROCBOMB
820 COLOUR 1
830 Y1%=Y1%+1
840 IF Y1%>29-B%(X1%) THEN S=S+FNSCORE
850 IF Y1%>29 THEN PROCMISS:GOTO 900
860 PRINT TAB(X1%,Y1%);" [fs7]"
870 SOUND 18,-10,255-Y1%,1
880 COLOUR 2
890 PRINT TAB(X1%,Y1%-1);" "
900 ENDPROC

```

```

910 DEF PROCMISS
920 F%=0:COLOUR 2
930 PRINT TAB(X1%,Y1%-1);" "
940 S(X1%)=0:B%(X1%)=0
950 ENDPROC
960 DEF FNCODE(X%,Y%)
970 LOCAL A%,C%:VDU 31,X%,Y%
980 A%=135
990 =(USR(&FFF4) AND &FFFF)DIV &100
1000
1010 DEF PROCCRASH
1020 PRINT TAB(X%-1,Y%);"[spc3]"
1030 IF F%=1 THEN PROCBOMB:FOR W=0 TO 1
00:NEXT:GOTO 1030
1040 X%=X%*64:Y%=1024-(Y%*40)
1050 *FX 9,5
1060 *FX 10,5
1070 VDU 19,2,9,0,0;
1080 SOUND 0,-15,15,100
1090 SOUND 1,0,200,100
1100 FOR W%=0 TO 5000:NEXT
1110 VDU 19,2,6,0,0;
1120 FOR Z%=0 TO 250
1130   GCOL 0,RND(4)-1
1140   PLOT 69,X%+RND(320),Y%+RND(200)
1150   NEXT
1160 IF (S+.9)>HIGH% THEN HIGH%=(S+.9)
1170 ENDPROC
1180 DEF PROCTUNE
1190 RESTORE 1280
1200 FOR N=1 TO 9
1210   READ P,D
1220   SOUND 17,-15,P,D
1230   SOUND 18,-15,P+48,D
1240   SOUND 19,-15,P+96,D
1250   FOR X=0 TO 400:NEXT
1260   NEXT
1270 ENDPROC
1280 DATA 73,20,93,20,73,3,73,7,65,3,61

```

```

, 3, 65, 7, 73, 7, 61, 10
1290 DEF FNSCORE
1300 PRINT TAB(3, 0); INT(S+S(X1%)+.9);
1310 SOUND 16, -15, 5, 10
1320 =S(X1%)
1330 DEF FNAGAIN
1340 COLOUR 1
1350 PRINT TAB(0, 2); " WOULD YOU LIKE TO
"
1360 PRINT " PLAY AGAIN (Y/N) ?";
1370 REPEAT AN$=GET$
1380 UNTIL AN$="Y" OR AN$="y" OR AN$=
"N" OR AN$="n"
1390 IF AN$="Y" OR AN$="y" THEN =TRUE E
LSE =FALSE
1400 DEF PROCINST
1410 VDU 17, 131, 17, 1, 12
1420 PRINT "[spc3]BOMBER COMMAND " ' '
1430 COLOUR 0:PRINT " DROP BOMBS ON TOW
N"
1440 PRINT "BY PRESSING SPC BAR."
1450 PRINT "ONLY ONE BOMB IN AIR"
1460 PRINT "AT ANY ONE TIME." ' '
1470 PRINT " PLANE MOVES DOWN AT"
1480 PRINT "EACH PASS OVER TOWN." '
1490 PRINT "GAME ENDS WHEN PLANE"
1500 PRINT "HITS BUILDING." ' '
1510 PRINT "150 PTS PER BUILDING";
1520 PRINT TAB(9, 24); "&"
1530 PRINT "250 PTS ON LANDING." '
1540 *FX 15 1
1550 PRINT TAB(3); "PRESS SPACE BAR"
1560 PRINT TAB(6); "TO PLAY"
1570 A=GET
1580 S=0:N%=300:WA%=0
1590 ENDPROC

```


Zombie

Now you can take on the challenge of Zombie in this exciting version of the game based on a program by Raymond Blake.

You are stranded on an island of cannibal zombies. Your only hope of escape is to lure them into the pot-holes scattered about the island. The trees on the island hinder the progress of both man and zombie, and falling into a pot-hole means eternal entrapment.

The game is made more difficult by the fact that you can move in only four directions. The zombies can not only move up, down, right and left, but can also move diagonally.

You need not panic, however, as the zombies are pretty short-sighted and this gives you a distinct edge with your 20/20 vision.

```
10 ON ERROR GOTO 1240
20 REM ZOMBIE
30
40 MODE 1
50 PROC_INIT
60 REPEAT
70     PROC_NEW_GAME
80     PROC_PLAY
90     UNTIL NOT FN_ANOTHER
100 CLS
110 END
120
130 DEF PROC_INIT
140 VDU 23,0,10,32,0;0;0;
150 VDU 23,128,0,&3C,&5A,&FF,&C3,&42,&
3C,0
160 VDU 23,129,8,&7E,&90,&3C,&24,&24,&
```



```

24, &33
170 VDU 23, 130, &38, &44, &82, &8A, &EE, &28
, &28, &44
180 VDU 19, 3, 2, 0, 0, 0
190 DIM A%(6), B%(6)
200 *FX 4 1
210 ENDPROC
220 DEF PROC_NEW_GAME
230 CLS
240 P%=10:Q%=15
250 PROC_TREES
260 PROC_HOLES
270 PROC_ZOMBIES
280 ENDPROC
290 DEF PROC_TREES
300 COLOUR 3
310 FOR I%=1 TO 40
320 PRINT TAB(RND(39)-1, RND(31)-1); "[f
s2]";
330 NEXT I%
340 ENDPROC
350 DEF PROC_HOLES
360 COLOUR 2
370 FOR I%=1 TO 12
380 PRINT TAB(RND(39)-1, RND(31)-1); "o"
;
390 NEXT
400 ENDPROC
410 DEF PROC_ZOMBIES
420 COLOUR 1
430 FOR I%=1 TO 6
440 REPEAT
450 X%=RND(36):Y%=RND(30)
460 UNTIL FNC(X%,Y%)=32
470 A%(I%)=Y%:B%(I%)=X%
480 PRINT TAB(B%(I%), A%(I%)); "[fs0]";
490 NEXT
500 COLOUR 2
510 REPEAT Q%=Q%+1

```

```

520     UNTIL FNC(Q%,P%)=32
530 PRINT TAB(Q%,P%); "[fs1] ";
540 ENDPROC
550 DEF FNC(X%,Y%)
560 VDU 31,X%,Y%
570 A%=135
580 =(USR(&FFF4)AND&FFFF)DIV&100
590 DEF PROC_PLAY
600 REPEAT
610     DIE=FALSE
620     *FX 15 1
630     REPEAT
640         A%=GET
650         UNTIL A%>135 AND A%<140
660         M%=( (A%=138) * (P%<30) ) - ( (A%=139) *
(P%>0) )
670         N%=( (A%=137) * (Q%<38) ) - ( (A%=136) *
(Q%>0) )
680         PRINT TAB(Q%,P%); " ";
690         C%=FNC(Q%+N%,P%+M%)
700         IF C%=ASC"[fs0]" THEN DIE=TRUE
710         IF C%=ASC"o" THEN PROC_PIT
720         IF C%=32 THEN P%=P%+M%:Q%=Q%+N%
730         COLOUR 2
740         IF DIE THEN 900
750         PRINT TAB(Q%,P%); "[fs1] ";
760         COLOUR 1
770         FOR I%=1 TO 6
780             IF A%(I%)=99 THEN 840
790             A1%=SGN(P%-A%(I%)):B1%=SGN(Q%-B%
(I%))
800             PRINT TAB(B%(I%),A%(I%)); " ";
810             IF FNC(B%(I%)+B1%,A%(I%)+A1%)=32
THEN A%(I%)=A%(I%)+A1%:B%(I%)=B%(I%)+B1
% ELSE PROC_ZOMBY
820             PRINT TAB(B%(I%),A%(I%)); "[fs0]"
;
830             IF A%(I%)=P% AND B%(I%)=Q% THEN
DIE=TRUE:I%=6

```

```

840     NEXT I%
850     SOUND &11,-10,10,2
860     G%=0
870     FOR I%=1 TO 6
880     IF A%(I%)=99 THEN G%=G%+1
890     NEXT I%
900     UNTIL G%=6 OR DIE
910 IF DIE THEN PRINT TAB(Q%,P%); " ";
920 ENDPROC
930 DEF PROC_PIT
940 DIE=TRUE
950 PRINT TAB(0,31);"Whoops! You just
found a pit!";
960 FOR S%=101 TO 5 STEP-4
970 SOUND 2,-10,S%,2
980 NEXT
990 SOUND &110,-10,5,5:SOUND &102,-10,
5,3
1000 ENDPROC
1010 DEF PROC_ZOMBY
1020 C%=FNC(B%(I%)+B1%,A%(I%)+A1%)
1030 IF C%=ASC"[fs0]" THEN ENDPROC
1040 IF C%=ASC"[fs2]" THEN ENDPROC
1050 IF C%=ASC"[fs1]" THEN DIE=TRUE:END
PROC
1060 PRINT TAB(0,31);"Whoops! There goe
s a Zombie.";
1070 COLOUR 2:PRINT TAB(B%(I%)+B1%,A%(I
%)+A1%);"o";
1080 FOR S%=101 TO 5 STEP-4
1090 SOUND 2,-10,S%,2
1100 NEXT
1110 SOUND &110,-10,4,5:SOUND &102,-10,
5,5
1120 PRINT TAB(0,31);SPC(29);
1130 A%(I%)=99
1140 COLOUR 1
1150 ENDPROC
1160 DEF FN_ANOTHER

```

```

1170 COLOUR 2
1180 IF DIE THEN PRINT TAB(0,3);"The zo
mbies claim another victim !" ELSE PRINT
TAB(0,3);"You have escaped the zombies!
"
1190 PRINT'"Another Game ?";
1200 REPEAT K$=GET$
1210 UNTIL K$="Y" OR K$="N"
1220 IF K$="N" THEN *FX 4
1230 =(K$="Y")
1240 MODE 6
1250 *FX 4
1260 REPORT:PRINT" at line ";ERL
1270 END

```

Two-Wall Smash

This game, patterned after one written by Mike O' Neill, is a two-wall version of the popular arcade game which is often called ' Breakout' .

The aim of the game is to destroy all the bricks in the walls by controlling the ball with your bat. The bat moves horizontally at the bottom of the screen. The ball moves at two different angles, and there are three bat sizes. The bat gets smaller as the game progresses - after a certain number of hits, when the ball reaches the second wall and when the ball hits the top of the screen.

The colour of the brick determines how much it is worth. The high score displayed on the screen, next to the current score. The number of the ball in play is also shown. The high score is updated at the end of each game.

```
10 REM TWO-WALL SMASH
20
30 MODE 7
40 PROC_INIT
50 REPEAT
60     PROC_NEWGAME
70     REPEAT ball=ball-1
80         PROC_NEWBALL
90         PROC_PLAY
100     UNTIL ball=0
110     UNTIL NOT FN_ANOTHER
120 END
130
140 DEF PROC_INIT
150 VDU 23,0,10,32,0;0;0;
160 brick$="||4"
170 big$="//"
```

```

180 mid$="//"
190 sml$="/"
200 HIGH=0
210 PROC_WALLS
220 PRINT TAB(9,9);"[fs1][fs8]WALL SMA
SH[fs9]"
230 PRINT TAB(0,15);"[fs3]Press any ke
y to play";
240 *FX 15 1
250 dummy=GET
260 VDU 26
270 ENDPROC
280 DEF PROC_NEWGAME
290 S%=0:bat$=big$
300 CLS
310 PROC_WALLS
320 ball=3:B%=20
330 ENDPROC
340 DEF PROC_WALLS
350 PRINT TAB(0,0);CHR$141;"[fs1]SCORE
:[fs6]0";TAB(20,0);"[fs2]HIGH :[fs6]";H
IGH
360 PRINT TAB(0,1);CHR$141;"[fs1]SCORE
:[fs6]0";TAB(20,1);"[fs2]HIGH :[fs6]";H
IGH
370 PRINT TAB(0,2);"[fc7]";STRING$(38,
"p");"0";
380 FOR Y%=3 TO 23
390 PRINT TAB(0,Y%);"[fc7]5";TAB(38,Y%
);"[fc7]5";
400 NEXT
410 PRINT TAB(2,22);"[fc6]";
420 PROC_WALL(4)
430 PROC_WALL(10)
440 ENDPROC
450 DEF PROC_WALL(ROW%)
460 VDU 26
470 FOR Y%=0 TO 3
480 PRINT TAB(3,Y%+ROW%);CHR$(145+Y%);

```

```

STRING$(11,brick$);
490 NEXT
500 W%=88
510 VDU 28,3,23,36,0
520 ENDPROC
530 DEF PROC_NEWBALL
540 PRINT TAB(B%,22);SPC(LEN(bat$));
550 X%=RND(33):Y%=21
560 DX%=(X%>21)*2)+1
570 DY%=-1:PRINT TAB(X%,Y%);"O";
580 B%=20:PRINT TAB(B%,22);bat$;
590 PRINT TAB(0,23);"[fs1]Balls :[fs6]
";ball;
600 ENDPROC
610 DEF PROC_PLAY
620 REPEAT
630     PROC_MOVE(INKEY-98,INKEY-67)
640     PROC_BALL
650     FOR delay=1 TO (1500-S%)DIV10:NE
XT
660     PROC_MOVE(INKEY-98,INKEY-67)
670     UNTIL Y%>21
680 PRINT TAB(X%,Y%);"O";
690 SOUND 0,-10,6,10
700 FOR delay=1 TO 2000:NEXT
710 PRINT TAB(X%,Y%);" ";
720 ENDPROC
730 DEF PROC_MOVE(L%,R%)
740 b%=B%
750 IF L% AND B%>1 THEN B%=B%-1
760 IF R% AND B%<(34-LEN(bat$)) THEN B
%=B%+1
770 IF B%<>b% PRINT TAB(b%,22);SPC(LEN
(bat$));
780 PRINT TAB(B%,22);bat$;
790 ENDPROC
800 DEF PROC_BALL
810 IF X%=1 THEN DX%=1
820 IF X%=33 THEN DX%=-1

```

```

830 IF Y%=3 THEN DY%=1
840 IF Y%=21 AND (X%+1)>=B% AND (X%-1)
<(B%+LEN(bat$)) PROC_return ELSE PRINT T
AB(X%,Y%); "O";
850 VDU 31,X%,Y%,32
860 X%=X%+DX%:Y%=Y%+DY%
870 PROC_TEST
880 ENDPROC
890 DEF PROC_TEST
900 VDU 31,X%,Y%
910 A%=135:M%=(USR(&FFF4)AND&FFFF)DIV&
100
920 IF Y%>20 THEN ENDPROC
930 IF M%<>32 THEN PROC_HIT ELSE PRINT
"O";
940 ENDPROC
950 DEF PROC_return
960 PRINT TAB(X%,Y%); "O";
970 SOUND &12,-10,200,2
980 SOUND 2,-10,200,1
990 REPEAT UNTIL ADVAL(-7)>14
1000 DY%=-1
1010 ENDPROC
1020 DEF PROC_HIT
1030 IF Y%>20 THEN ENDPROC
1040 DY%=-DY%
1050 SOUND &11,-10,100,1
1060 IF X%MOD3=0 THEN PRINT TAB(X%-2,Y%
); " O";:IFDX%<0THENDX%=1
1070 IF X%MOD3=1 THEN PRINT TAB(X%,Y%);
"O ";:IFDX%>0THENDX%=-1
1080 IF X%MOD3=2 THEN PRINT TAB(X%-1,Y%
); " O ";
1090 W%=W%-1
1100 IF Y%>8 THEN P%=4-((Y%+2)MOD4) ELS
E P%=14-(Y%MOD4)
1110 S%=S%+P%
1120 PRINT TAB(7,0);S%;
1130 PRINT TAB(7,1);S%;

```



```

1140 IF W%=0 THEN PROC_EXTRABALL
1150 ENDPROC
1160 DEF PROC_EXTRABALL
1170 PRINT TAB(X%,Y%); " "
1180 PRINT TAB(B%,22);SPC(LEN(bat$));
1190 PROC_WALL(4)
1200 PROC_WALL(10)
1210 PROC_NEWBALL
1220 IF bat$=mid$ THEN bat$=sml$
1230 IF bat$=big$ THEN bat$=mid$
1240 ENDPROC
1250 DEF FN_ANOTHER
1260 IF S%>HIGH THEN PRINT TAB(25,0);S%
;TAB(25,1);S%
1270 VDU 28,3,20,36,4,12
1280 PRINT TAB(12,5);"[fs1]GAME OVER[fs
7]"
1290 PRINT''
1300 IF S%>HIGH THEN HIGH=S%:PRINT TAB(
7);"[fs8][fs3]A NEW HIGH SCORE![fs9]"
1310 PRINT'"[fs2]Another game Y/N ?";
1320 REPEAT KEY$=GET$
1330 UNTIL KEY$="Y" OR KEY$="N"
1340 VDU 26,12
1350 IF KEY$="Y" THEN =TRUE ELSE =FALSE

```

Sharp Shooter

In this fun program, based on an idea from Ant Hurriion, you control the man on the right of the cacti on your screen. You must shoot it out with the computer gunman.

The winner is the marksman who first gains 100 points. There are several levels of difficulty, and you' rable to choose the level at which you' ll start play at the start of the game.

```
10 REM Sharp Shooter
20
30 MODE 7
40 PROC_INIT
50 PROC_INSTR
60 REPEAT MODE 5
70     PROC_NEWGAME
80     PROC_PLAY
90     UNTIL NOT FN_ANOTHER
100 MODE 7
110 END
120
130 DEF PROC_INIT
140 VDU 23,0,10,32,0;0;0;
150 VDU 23,128,0,3,7,3,3,7,15,27
160 VDU 23,129,0,192,224,192,192,224,2
40,216
170 VDU 23,130,7,3,1,3,7,12,24,48
180 VDU 23,131,224,192,128,192,224,48,
24,12
190 VDU 23,132,0,0,0,1,3,1,113,31
200 VDU 23,133,0,0,0,128,192,128,128,1
28
210 VDU 23,134,1,1,1,15,9,9,25,0
220 VDU 23,135,128,128,128,128,128,136
```

```

,248,0
  230 VDU 23,136,0,0,0,1,3,1,1,1
  240 VDU 23,137,0,0,0,128,192,128,142,2
48
  250 VDU 23,144,1,1,1,1,1,9,15,0
  260 VDU 23,145,128,128,128,240,144,144
,152,0
  270 VDU 23,146,48,48,52,52,60,24,24,24
  280 VDU 23,147,0,0,0,56,56,56,0,0
  290 VDU 23,148,0,192,97,51,25,77,127,0
  300 VDU 23,149,0,3,134,204,152,178,254
,0
  310 ENVELOPE 1,1,-1,-2,-4,1,1,1,126,0,
0,-10,126,50
  320 ENDPROC
  330 DEF PROC_INSTR
  340 PRINT TAB(12,1);"[fs1]SHARP SHOOTE
R[fs7]"
  350 PRINT'"[fs6]You control the Gunman
on the right of"
  360 PRINT"[fs6]the screen and must sho
ot the computer"
  370 PRINT"[fs6]gunman on the left of t
he screen.'"
  380 PRINT'"[fs2]To move UP press[fs3
]A"
  390 PRINT'"[fs2]To move DOWN press[fs3
]Z"
  400 PRINT'"[fs2]To FIRE press[fs3]RETU
RN"
  410 PRINT'"[fs6]The first to 100 poin
ts wins!"
  420 PRINT'"[fs6]Difficulty[fs3]1-9[fs
6](9 easy!)[fs3]";
  430 REPEAT D%=GET-ASC"0"
  440 UNTIL D%>0 AND D%<10
  450 PRINT'"[fs1]GOOD LUCK ";:IF D%<5
THEN PRINT"SUCKER!"
  460 DU=INKEY(200)

```

```

470  ENDPROC
480  DEF  PROC_NEWGAME
490  VDU  19,1,2,0,0,0
500  VDU  19,3,6,0,0,0
510  VDU  23,0,10,32,0;0;0;
520  S1%=0:S2%=0
530  PROC_LANDSCAPE
540  ENDPROC
550  DEF  PROC_LANDSCAPE
560  VDU  24,0;639;1279;1023; 18,0,131,1
6
570  VDU  24,0;0;1279;639; 18,0,130, 16,
26, 5
580  GCOL 0,1
590  FOR CACT%=0 TO 30
600  X%=RND(40)*32:Y%=RND(21)*32
610  MOVE X%,Y%:VDU 146
620  NEXT CACT%
630  VDU 4, 17,131, 17,0
640  PRINT TAB(0,0);"YOU:";S1%;TAB(10,0
); "ME :";S2%
650  Y1%=RND(20)+10:Y2%=RND(20)+10
660  y1%=Y1%:y2%=Y2%
670  PROC_DBL(2,Y1%,"[fs0][fs1]", "[fs2]
[fs3]")
680  PROC_DBL(17,Y2%,"[fs0][fs1]", "[fs2]
[fs3]")
690  COLOUR 131
700  PRINT TAB(8,10);"READY!"
710  *FX 15 1
720  DU=INKEY(200)
730  PRINT TAB(8,10);"[spc6]"
740  ENDPROC
750  DEF  PROC_PLAY
760  REPEAT
770      IF y1%<>Y1% THEN PROC_DBL(2,y1%,
" ", " "):PROC_DBL(2,Y1%,"[fs0][fs1]", "[
fs2][fs3]")
780      IF y2%<>Y2% THEN PROC_DBL(17,y2%

```

```

, " ", " "):PROC_DBL(17,Y2%,"[fs0][fs1]"
,"[fs2][fs3]")
790 y1%=Y1%:y2%=Y2%
800 Y1%=Y1%-(RND(3)=1 AND Y1%<30)+(R
ND(3)=1 AND Y1%>10)
810 Y2%=Y2%-(INKEY(-98) AND Y2%<30)+
(INKEY(-66) AND Y2%>10)
820 IF INKEY(-74) AND (RND(0)<.3+(D%
/10)) THEN PROC_UFIRE
830 IF RND(10)>D% THEN PROC_FIRE
840 IF RND(5)=1 THEN COLOUR 1:COLOUR
130:PRINT TAB(RND(20)-1,11+RND(19));"[f
s2]";
850 UNTIL S1%=100 OR S2%=100
860 ENDPROC
870 DEF FN_ANOTHER
880 PRINT TAB(8,10);
890 IF S1%=100 THEN PRINT"I WON!" ELSE
PRINT"YOU WON!"
900 COLOUR 130
910 PRINT''' "Do you want another"
920 PRINT"go ";:IF S2%=100 THEN PRINT"
sharpshooter ?"; ELSE PRINT"?";
930 REPEAT K$=GET$
940 UNTIL K$="Y" OR K$="N"
950 IF K$="Y" THEN =TRUE ELSE =FALSE
960 DEF PROC_DBL(X%,Y%,T$,B$)
970 COLOUR 0
980 IF Y%<12 THEN COLOUR 131 ELSE COLO
UR 130
990 PRINT TAB(X%,Y%);T$
1000 IF Y%<11 THEN COLOUR 131 ELSE COLO
UR 130
1010 PRINT TAB(X%,Y%+1);B$;
1020 ENDPROC
1030 DEF PROC_FIRE
1040 PROC_DBL(2,y1%,"[fs8][fs9]", "[fs0]
[fs1]")
1050 SOUND &10,-10,4,2

```

```

1060 FOR B%=4 TO 16
1070 IF B%<>4 THEN PROC_DBL(B%-1,y1%,"
[fs3]"",") ELSE PROC_DBL(B%,y1%,"[fs3]",
"")
1080 NEXT
1090 PROC_DBL(16,y1%," ",")
1100 IF y1%=y2% OR y1%=y2%+1 THEN PROC_
UDIE
1110 ENDPROC
1120 DEF PROC_UFIRE
1130 PROC_DBL(17,y2%,"[fs4][fs5]", "[fs6
][fs7]")
1140 SOUND &10,-10,5,2
1150 FOR B%=15 TO 2 STEP -1
1160 IF B%<>15 THEN PROC_DBL(B%+1,y2%,"
[fs3]"",") ELSE PROC_DBL(B%,y2%,"[fs3]",
"")
1170 NEXT
1180 PROC_DBL(3,y2%," ",")
1190 IF y2%=y1% OR y2%=y1%+1 THEN PROC_
IDIE
1200 ENDPROC
1210 DEF PROC_UDIE
1220 PROC_DBL(17,y2%," ", "[fs4][fs5]")
1230 SOUND 1,1,20,1
1240 S1%=S1%+10
1250 COLOUR 0:COLOUR 131
1260 PRINT TAB(4,0);S1%
1270 REPEAT UNTIL INKEY(-147)
1280 *FX 15 1
1290 D=INKEY(200)
1300 IF S1%<>100 THEN PROC_LANDSCAPE
1310 ENDPROC
1320 DEF PROC_IDIE
1330 PROC_DBL(2,y1%," ", "[fs4][fs5]")
1340 SOUND 1,1,20,1
1350 S2%=S2%+10
1360 COLOUR 0:COLOUR 131
1370 PRINT TAB(14,0);S2%

```

```
1380 REPEAT UNTIL INKEY(-147)
1390 *FX 15 1
1400 D=INKEY(200)
1410 IF S2%<>100 THEN PROC_LANDSCAPE
1420 ENDPROC
```

Racer

In this game, on a racing track designed by Scott Vincent, you must drive your car around the circuit as quickly as you can without crashing. The faster you go, the more your score increases.

The game ends when your car skids off the track and mounts the grass. Obviously, at higher speeds the car tends to skid more when turning. If you go too fast, you' llose control. The car starts off at around ten miles an hour.

```
10 REM Racer
20
30 MODE 4
40 PROC_INIT
50 REPEAT
60     PROC_NEWGAME
70     PROC_RACE
80     UNTIL NOT FN_ANOTHER
90 MODE 7
100
110 END
120 DEF PROC_INIT
130 VDU 19,7,2,0,0,0
140 VDU 23,0,10,32,0;0;0;
150 VDU 23,128,0,24,24,60,60,189,255,2
53
160 VDU 23,129,194,233,126,124,60,56,1
60,64
170 VDU 23,130,7,2,30,127,127,30,2,7
180 VDU 23,131,64,160,56,60,124,126,24
1,194
190 VDU 23,132,153,255,189,60,60,24,24
,0
```



```

200 VDU 23,133,2,5,28,60,62,126,143,67
210 VDU 23,134,224,64,120,254,254,120,
64,224
220 VDU 23,135,67,143,126,62,60,28,5,2
230 VDU 23,136,-1,-1,-1,-1,-1,-1,-1,-1
240 H%=0
250 DIM X(2),Y(2)
260 ENDPROC
270 DEF PROC_TRACK
280 FOR Y%=1 TO 30
290 PRINT TAB(0,Y%);STRING$(40,"[fs8]"
);
300 NEXT
310 VDU 28,3,28,36,4
320 RESTORE
330 FOR A%=1 TO 41
340 READ Y%,X%,L%
350 PRINT TAB(X%,Y%);STRING$(L%," ");
360 NEXT
370 VDU 26
380 PRINT TAB(0,0);"Score :";S%;"[spc6
]";TAB(20,0);"High :";H%
390 ENDPROC
400
410 DATA 3,15,11,4,14,14,5,14,15,6,4,8
,6,14,6,6,22,8,7,3,10,7,15,5,7,23,7,8,2,
12,8,16,5,8,24,6
420 DATA 9,1,13,9,17,5,9,24,7,10,1,6,1
0,9,6,10,17,5,10,25,7,11,1,6,11,10,12,11
,24,7
430 DATA 12 ,1,7,12,11,10,12,24,7,13,1
,8,13,11,10,13,23,8,14,1,9,14,14,6,14,22
,8,15,2,9,15,21,9
440 DATA 16,2,12,16,18,11,17,3,25,18,5
,22,19,6,8,19,18,7,20,7,5,20,19,4
450
460 DEF PROC_NEWGAME
470 S%=0
480 PROC_TRACK

```

```

490 X(1)=640:Y(1)=512
500 X(2)=0:Y(2)=0.1
510 V%=10:A$="[fs0]"
520 PRINT TAB(10,2);"Speed =[spc5]m.p.
h."
530 PRINT TAB(18,2);V%;
540 ENDPROC
550 DEF PROC_RACE
560 CRASH=FALSE
570 SOUND 0,-10,7,255
580 REPEAT
590     VDU 5
600     MOVE X(1),Y(1):GCOL0,0:VDU136
610     X(1)=X(1)-X(2)
620     Y(1)=Y(1)+Y(2)
630     IF FN_HIT THEN CRASH=TRUE
640     MOVE X(1),Y(1):GCOL 0,1:PRINT A$
650     VDU 4
660     S%=S%+(V%DIV10)
670     PRINT TAB(7,0);S%
680     SOUND &11,0,V%,1
690     IF INKEY(-74) V%=V%-10*(V%<150)
700     IF INKEY(-90) V%=V%+10*(V%>10)
710     PROC_MOVE
720     PRINT TAB(18,2);V%;" ";
730     UNTIL CRASH
740 ENDPROC
750 DEF PROC_MOVE
760 PROC_DELAY(V%)
770 O=V%/5
780 X(2)=SGN(X(2))*O
790 Y(2)=SGN(Y(2))*O
800 IF INKEY(-65) PROC_LEFT
810 IF INKEY(-2) PROC_RIGHT
820 ENDPROC
830 DEF PROC_DELAY(T%)
840 REMFORW%=1 TO 500-(10*T%):NEXT
850 ENDPROC
860 DEF PROC_LEFT

```

```

870 A$=CHR$(128+((ASC(A$)+1)MOD8))
880 IF A$<"[fs4]" THEN X(2)=0 ELSE X(2)
)=0
890 IF A$<"[fs2]" OR A$>"[fs6]" THEN Y
(2)=0 ELSE Y(2)=0
900 IF A$="[fs4]" OR A$="[fs0]" THEN X
(2)=0
910 IF A$="[fs2]" OR A$="[fs6]" THEN Y
(2)=0
920 ENDPROC
930 DEF PROC_RIGHT
940 A$=CHR$(128+((ASC(A$)-1)MOD8))
950 IF A$<"[fs4]" THEN X(2)=0 ELSE X(2)
)=0
960 IF A$<"[fs2]" OR A$>"[fs6]" THEN Y
(2)=0 ELSE Y(2)=0
970 IF A$="[fs4]" OR A$="[fs0]" THEN X
(2)=0
980 IF A$="[fs2]" OR A$="[fs6]" THEN Y
(2)=0
990 ENDPROC
1000 DEF FN_HIT
1010 IF POINT(X(1),Y(1))<>0 THEN =TRUE
1020 IF POINT(X(1)+30,Y(1)-30)<>0 THEN
=TRUE
1030 =FALSE
1040 DEF FN_ANOTHER
1050 SOUND &10,-10,255,255
1060 IF S%>H% THEN H%=S%
1070 PRINT TAB(26,0);H%
1080 PRINT TAB(10,10);"You hit somethin
g!"
1090 REPEAT UNTIL INKEY(-147)
1100 PRINT TAB(10,21);"Another go ?";
1110 SOUND &10,0,0,0
1120 *FX 15 1
1130 *FX 202 32
1140 REPEAT K$=GET$
1150 UNTIL K$="Y" OR K$="N"

```

```
1160 IF K$="Y" THEN =TRUE ELSE =FALSE
```

Ski Run

In this program you have to steer your skier down the ski-slope, dodging the trees as you go. Your speed increases as you progress down the slope. If you move off the side, you'll reappear on the other side of the screen. If you drop off the bottom of the world, you'll reappear on the top of the screen. The 'Z' key moves you left, and the '/' moves you right. The skier automatically moves down unless he is moving across the slope. The time for the completed run will be shown at the end of the game.

```
10 REM Ski Run
20
30 MODE 1
40 PROC_INIT
50 REPEAT
60   PROC_NEW_GAME
70   PROC_PLAY
80   UNTIL NOT FN_ANOTHER
90 END
100
110 DEF PROC_INIT
120 DIM M% 3
130 L%=-98:R%=-105
140 VDU 23,0,10,32,0;0;0;
150 VDU 23,128,&18,&18,&10,&3C,&50,&08
, &11, &FE
160 VDU 23,129,&18,&18,&08,&3C,&0A,&10
, &88, &7F
170 VDU 23,130,&1C,&5D,&3E,&08,&2A,&36
, &22, &22
180 VDU 23,131,60,118,227,201,221,201,
201,201
```

```

190 VDU 23,132,&08,&1C,&1C,&3E,&7F,&41
,0,0
200 VDU 23,133,0,0,0,0,0,8,8,8
210 VDU 23,134,&80,&80,&80,&80,&80,&80
,&80,&80
220 VDU 23,135,1,1,1,1,1,1,1,1
230 VDU 19,2,2,0,0,0
240 ENVELOPE 1,1,-1,-2,-4,1,1,1,126,0,
0,-10,126,50
250 Tree$=CHR$(5)+CHR$(18)+CHR$(0)+CHR
$(2)+"[fs4]"+CHR$(8)+CHR$(18)+CHR$(18)+C
HR$(0)+"[fs5]"+CHR$(4)
260 COLOUR 0:COLOUR 131:CLS
270 ENDPROC
280
290 DEF PROC_NEW_GAME
300 PROC_LANDSCAPE
310 X%=2:Y%=2
320 PRINT TAB(X%,Y%);"[fs2]";
330 Crash%=FALSE:Finish%=FALSE
340 ENDPROC
350
360 DEF PROC_LANDSCAPE
370 LOCAL X%,Y%
380 CLS:COLOUR 3:COLOUR 129
390 GCOL 0,1
400 MOVE 0,928:DRAW 0,1023
410 PRINT TAB(0,0);"START"
420 MOVE 159,928:DRAW 159,1023
430 MOVE 1088,0:DRAW 1088,96:DRAW 1279
,96:DRAW 1279,0
440 PRINT TAB(34,29);"FINISH";
450 FOR TREE=1 TO 150
460 REPEAT
470 X%=32*(RND(40)-1):Y%=31+(32*(RND
(32)-1))
480 UNTIL (X%>160 OR Y%<863) AND (X%
<1056 OR Y%>128)
490 MOVE X%,Y%

```

```

500 PRINT Tree$;
510 NEXT
520 COLOUR 1:COLOUR 131
530 PRINT TAB(2,2);
540 ENDPROC
550
560 DEF PROC_PLAY
570 *FX 15 1
580 wait%=GET
590 TIME=0
600 VDU 31,X%,Y%,32,8
610 Man$="[fs2]"
620 REPEAT
630     IF FN_CLEAR THEN PRINT Man$; ELS
E PROC_CRASH:GOTO 750
640     PROC_DELAY
650     Man$="[fs2]"
660     D%=1
670     IF INKEY(L%) THEN X%=X%-1:D%=0:Man$="[fs1]"
680     IF INKEY(R%) AND NOT INKEY(L%) THEN X%=X%+1:D%=0:Man$="[fs0]"
690     Y%=Y%+D%
700     IF Y%=32 THEN Y%=0:IF X%<5 THEN Y%=2
710     IF X%=-1 THEN X%=39
720     IF X%=40 THEN X%=0
730     IF X%>33 AND Y%>28 THEN PROC_FINISH
740     VDU 8,32,31,X%,Y%
750     UNTIL Crash% OR Finish%
760 Time%=TIME
770 ENDPROC
780
790 DEF FN_CLEAR
800 A%=135:!M%=USR&FFF4
810 IF M%?1=0 THEN =FALSE ELSE =TRUE
820
830 DEF PROC_DELAY

```

```

840 T%=TIME+(32-Y%)
850 REPEAT UNTIL TIME>T%
860 SOUND &11,-10,Y%DIV4,3
870 ENDPROC
880
890 DEF PROC_CRASH
900 IF X%>33 AND Y%>28 THEN PROC_FINIS
H:ENDPROC
910 COLOUR 0
920 PRINT TAB(X%,Y%);"[fs3]";
930 Crash%=TRUE
940 SOUND &11,1,20,1
950 ENDPROC
960
970 DEF PROC_FINISH
980 Finish%=TRUE
990 SOUND &11,1,175,2
1000 PRINT "[fs2]";
1010 ENDPROC
1020
1030 DEF FN_ANOTHER
1040 *FX 15 1
1050 COLOUR 3
1060 IF Y%>5 AND Y%<15 THEN VDU 28,11,2
4,29,16 ELSE VDU 28,11,14,29,6
1070 COLOUR 129:CLS
1080 IF Finish% THEN PRINT"You have fin
ished""in ";Time%/100;" seconds." ELSE
PRINT" You have Crashed!"
1090 COLOUR 2
1100 PRINT"" Another go ? ";
1110 REPEAT K$=GET$
1120 IF K$>"Z" THEN K$=CHR$(ASC(K$)-3
2)
1130 UNTIL K$="Y" OR K$="N"
1140 VDU 26
1150 COLOUR 0:COLOUR 131:CLS
1160 =(K$="Y")

```