



BOOK OF GAMES

WARNING
HAVE YOU SEEN
THIS MAN



HIGHLY DANGEROUS
WANTED FOR
TREACHERY



CONTENTS

**A message from
Ideas Central ...**

Hi...

Melissa Ravenflame, here, co-ordinator of IDEAs Central. Welcome to another superb *Computer + Video Games Book of Games* — 32 pages packed with great listings which you should find great fun. They will help improve your programming skills as well.

Otiss, Big Red and B-Con have given the listings a good going over so they should be free of bugs. But if you do have any problems don't hesitate to call the Bug Hunters.



Page 3 Treachery Amstrad

14 Eggs BBC B or Electron

17 Ski Master Spectrum

18 Reconnaissance QL

22 Operation Kristos BBC B

26 Car Race Atari

29 Skycapers Amstrad

31 Low Level CBM 64

Bluff and bamboozlement are the weapons you'll need to work your way through the world of espionage. Fun in the farmyard as the Farmer Giles has an egg-citing time.

There's no business like snow business as you cope with the ski slopes.

A special treat for QL owners Car and intelligence will be needed in this game. Mission Impossible? Time will tell with this alien adventure.

Squeals on wheels... so get into gear for race track action.

A taxing game. The Inland Revenue is out to get a lot of money.

Dares in the air as you fly close to the ground and destruction.

TREACHERY

IT IS THE YEAR 1984 AND THANKFULLY BIG BROTHER IS STILL JUST A DISTANT NIGHTMARE. YET, A MENACE JUST AS PERILOUS THREATENS THE WORLD....

THE SINISTER PROFESSOR SCHWEINSTEIN!



HEHEHEH!

AFTER MANY YEARS OF SECRET RESEARCH, SCHWEINSTEIN SUCCEEDED IN CREATING WHAT COULD BE THE ULTIMATE WEAPON....

A DEVICE HE CALLED THE MINDBOMB!

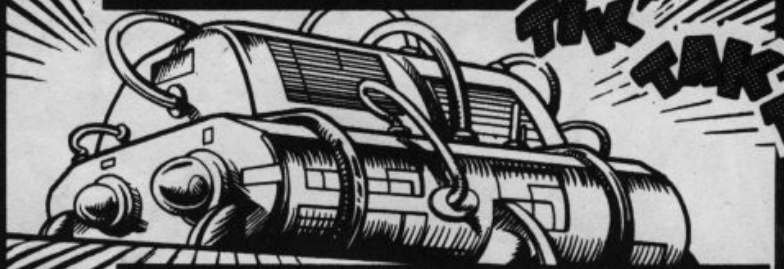
WHEN DETONATED, IT SIMPLY WIPE CLEAN THE MEMORIES OF ANYONE WITHIN TEN MILES OF THE EXPLOSION.

EVEN WHEN IT IS DOING NOTHING, IT HAS A TENDENCY TO CAUSE RANDOM OUTBREAKS OF AMNESIA FOR MILES AROUND!



SCHWEINSTEIN HAS ALSO PERFECTED A SUPERB DELIVERY SYSTEM FOR HIS WEAPON, THE MANDROID!

IN REALITY AN ANDROID, IT SO CLOSELY RESEMBLES A MAN THAT IT IS VIRTUALLY UNDETECTABLE.



CHORTLE!

WITH THE MINDBOMB CONCEALED IN THE MANDROID'S ABDOMEN, SCHWEINSTEIN IMAGINED HE COULD BLACKMAIL THE WORLD.



HOWEVER, JUST AS HIS CREATION WAS COMPLETE, AN ACCIDENT OCCURED AND SCHWEINSTEIN BECAME THE FIRST VICTIM OF HIS OWN WEAPON....



HE FORGOT EVERYTHING!

HE FORGOT ABOUT THE MANDROID
AND ITS HOMING BEACON....

DUNHHHHH?

HE FORGOT ABOUT THE MINDBOMB AND
ITS REMOTE FIRING BUTTON....

HE EVEN FORGOT ABOUT HIS TWO HOLLOW
TEETH, WHICH CONTAIN THE AMNESIA ANTIDOTE
AND THE ENTIRE HISTORY OF HIS RESEARCHES
ON MICRO-DISK.

HE ALSO FORGOT ABOUT THE BLACKMAIL LETTERS
HE SENT TO THE BRITISH AND THE SOVIETS!

00#!?+x!

?

NOW MIG AND THE KGB ARE SEARCHING
DESPERATELY FOR SCHWEINSTEIN, HIS FIRING
BUTTON AND HIS HOMING BEACON.

SCHWEINSTEIN IS A BROKEN MAN, ROAMING THE URBAN
WILDERNESS IN SEARCH OF HIS LOST IDENTITY....

SNIFF.
KOFF!

WHOEVER FINDS HIM FIRST WILL
HAVE PULLED OFF A TREMENDOUS
INTELLIGENCE COUP.

MEANWHILE, THE MANDROID IS
WANDERING ACROSS EUROPE!

THEY KNOW WHICH CITY IT IS IN AT
ANY ONE TIME BY THE INCREASED
INCIDENCE OF AMNESIA. BEYOND THAT,
THEY CAN GO NO FURTHER IN
TRACKING IT DOWN.

WHICHEVER SIDE GAINS CONTROL OF
THE HOMING BEACON AND FIRING
BUTTON WILL FIND IT EASY TO WIPE
OUT THE ENEMY'S INTELLIGENCE
NETWORK COMPLETELY....

LONDON AND MOSCOW ARE IN
DANGER OF LOSING THEIR MINDS!

TREACHERY

Treachery is a tricky business at the best of times and you can be sure that the "friend" you are playing will be doing his worst to double-cross you. Bluff and bamboozlement are the weapons to use. A knowing smile or a subtle twitch of an eyebrow as you read through your agents' reports can be enough to send your opponent into a frenzied panic. But beware! He won't start tearing his hair out. He'll just nod calmly at you as if to say: "Yes, you've fallen for my trap."

LESSON ONE: IT'S FOR YOU-HOO!

Sending, receiving and intercepting messages is the key to success in *Treachery*. No self-respecting spymaster forgets that every order he sends and every report he receives may have been intercepted en route by the enemy and a clever spymaster uses this fact to his advantage.

Only a real traitor will betray your messages to the enemy and that betrayal will result in a message reporting interception to enemy HQ. With any luck, one of your agents will himself intercept the message of interception and report it back to your HQ. You then discover the identity of the traitor in your midst and you also know that your intercepting agent can be completely trusted. If he really belonged to the enemy, he would never have sent in an interception report.

So, in the early stages, it's a good idea to send orders by the longest routes possible in the hope that they get intercepted. At the same time, you should call in reports by the shortest and safest routes back to HQ since they may carry information you definitely don't want betrayed to the enemy.

LESSON TWO: THE BEST MOLES STAY UNDERGROUND

Until you have some idea of which agents are to be trusted, the best action is no action. If you use your agents James Bond fashion and leave a trail of dead bodies across Europe, you are very likely to lose the game. Your own agents become prominent targets, you waste time that can be more profitably spent gathering information and you are quite likely to bump off someone who is not an enemy agent.

Don't forget that even enemy agents can be very useful so long as you know who they are. They will be gathering information too — information that you can intercept and use to your own advantage.

Relocating an agent in a different city should only be done of necessity — you waste valuable spying opportunities and you might, in fact, be moving an enemy double-agent to a better position!

Instead, the first few turns should be spent making

innocuous "searches" in the cities where you have agents. Soon you will build up a working knowledge of who to trust. Only then should you begin to think about new stations for your agents.

A vital task is to establish secure routes for your orders and reports by moving men you know you can rely on into gaps or suspect links in your communications.

Appropriately enough, Berlin, notorious for its wealth of spies, is a key centre for routing messages through. Without a man in Berlin, your flexibility in sending orders and calling in reports is severely limited and it is the one place where you might allow yourself the luxury of eliminating the opposition once and for all.

Another essential task is to make sure you have a trusted mole close to or in the enemy HQ, ready to intervene if the opposition finds Schweinstein first and makes a break for home.

LESSON THREE: A SNEAK IN TIME SAVES NINE

Above all, you must be sneaky. Not only is it vital in winning the game, it's also great fun! How sneaky you can be depends on the calibre of your opponent — it's no use employing tortuously subtle ruses against a blockhead — but it always offers rewarding opportunities for creative thinking.

One of my favourites is to send orders to an agent to kill your own Master Spy but to send them by an

incomplete route so that they never actually arrive! You make sure, however, that the route includes a traitor who will betray your orders to the enemy. Your opponent immediately gets the impression that you have discovered one of his triple agents and have decided to eliminate him. He never suspects for one moment that you would kill your own Master Spy and now believes he knows an agent he can trust!

LESSON FOUR: DON'T PANIC

Never lose your cool. Rushing into action with all guns blazing is a tempting recourse during a crisis but calm, logical thinking will usually find a quiet solution to the problem. Remember, it is Smiley, not Bond, who wins at *Treachery*.

TREACHERY

```

10 ' *****
20 REM ** TREACHERY **
30 ' *****
40 ' ** Converted From Spectrum **
50 ' ** & Commodore Listings in **
60 ' ** Computer & Video Games **
70 ' *****
80 CLEAR
90 MODE 1
100 WINDOW#1,11,37,5,22:PAPER#1,0
110 ON ERROR GOTO 7750
120 GOTO 3420
130 ' *****
140 REM ** Print Message Sheet **
150 ' *****
160 CLS#1
170 PLOT 158,336,2:DRAWR 438,0:DRAWR 0,-
294:DRAWR -438,0:DRAWR 0,294
180 PEN 1
190 RETURN
200 ' *****
210 REM ** Start of Turn Page **
220 ' *****
230 CLS:CLS#1:x=3:y=2:GOSUB 2940
240 PLOT 1,1,1:DRAW 1,399:DRAW 639,399:D
RAW 639,1:DRAW 1,1
250 GOSUB 160
260 FOR k=1 TO 3
270 LOCATE#1,4,1+k:PRINT#1, t$(k)
280 NEXT
290 LOCATE#1,5,5:PRINT#1,p$(p,1)
300 LOCATE#1,2,7:PRINT#1,"OPERATIONAL FI
LE"
310 LOCATE#1,2,9:PRINT#1,"REF: ";p$(p,2)
320 LOCATE#1,2,11:PRINT#1,"Heidelberg Sc
hweinstein"
330 LOCATE#1,2,13:PRINT#1,"GRADE 1 ACCES
S ONLY"
340 LOCATE#1,2,15:PRINT#1,"Enter Cleanan
ce Code"
350 LOCATE#1,2,17:PRINT#1,STRING$(6,210)
360 SOUND 7,200,25,5
370 c$="":FOR k=1 TO 6
380 k$=INKEY$:IF k$="" THEN 380
390 IF k$<"a" OR k$>"z" THEN GOTO 380
400 k$=CHR$(ASC(k$)-32):c$=c$+k$
410 LOCATE#1,1+k,17:PRINT#1,k$
420 IF INKEY$<>" " THEN 420
430 NEXT :IF day=3 THEN g$(p)=c$
440 IF c$<>g$(p) THEN LOCATE#1,9,17:PRIN
T#1,"IS INCORRECT":SOUND 7,1000,25,5,0,0
,7:FOR k=1 TO 1000:NEXT:LOCATE#1,9,17:PR
INT#1," "GOTO 350
450 RETURN
460 ' *****
470 REM ** End of Page **
480 ' *****
490 LOCATE#1,2,17:PRINT#1,CHR$(24);" KEY
COPY OR RETURN ";CHR$(24)
500 SOUND 7,200,25,5
510 IF INKEY(18)<>-1 THEN GOTO 550
520 IF INKEY(9)<>-1 THEN 540
530 GOTO 510
540 LOCATE#1,2,17:PRINT#1,"
":GOSUB 7620
550 RETURN
560 GOSUB 160
570 LOCATE#1,2,2:PRINT#1,p$(p,1);" DAY",
day
580 LOCATE#1,2,3:PRINT#1,"INCOMING REPOR

```

```

T"
590 LOCATE#1,2,5:PRINT#1,"PRESS KEY FOR
YOUR AGENT"
600 k$=INKEY$:IF k$="" THEN 600
610 IF k$="" THEN RETURN
620 IF k$>="a" AND k$<="z" THEN a9=ASC(k
$)-96:GOTO 650
630 IF k$>="0" AND k$<="9" THEN a9=ASC(k
$)-21:GOTO 650
640 GOTO 600
650 IF ASC(n$(a9,4))=0 THEN GOTO 610
660 orank=ASC(n$(a9,p+1))
670 erank=ASC(n$(a9,4-p))
680 IF orank>erank AND ASC(MID$(f$(1,a9)
,5,1))>0 THEN 720
690 LOCATE#1,2,7:PRINT#1,CHR$(24);" NO R
EPORT IS AVAILABLE ";CHR$(24)
700 FOR k=1 TO 500:NEXT
710 GOTO 560
720 IF ASC(MID$(f$(1,a9),2,1))=0 THEN 78
0
730 m1=ASC(MID$(f$(1,a9),2,1))
740 m5=ASC(MID$(f$(1,a9),3,1))
750 m6=ASC(MID$(f$(1,a9),4,1))
760 f$(1,a9)=LEFT$(f$(1,a9),1)+CHR$(0)+M
ID$(f$(1,a9),3)
770 GOTO 820
780 m1=17:IF ASC(LEFT$(f$(1,a9),1))>1 TH
EN m1=16
790 m5=ASC(MID$(f$(1,a9),6))
800 m6=0
810 f$(1,a9)=LEFT$(f$(1,a9),5)+MID$(f$(1
,a9),7,8)+RIGHT$(f$(1,a9),1)
820 m2=a9
830 m3=p
840 m4=day-1
850 f$(1,a9)=LEFT$(f$(1,a9),4)+CHR$(ASC(
MID$(f$(1,a9),5,1))-1)+MID$(f$(1,a9),6)
860 m$=CHR$(m1)+CHR$(m2)+CHR$(m3)+CHR$(m
4)+CHR$(m5)+CHR$(m6)
870 from=ASC(n$(a9,1))
880 type=1:IF p=2 THEN de=25 ELSE de=18
890 LOCATE#1,2,5:PRINT#1,"
"
900 LOCATE#1,2,5:PRINT#1,x$(a9);" TO ";M
ID$(p$(p,1),5):GOSUB 2360
910 IF k$="X" THEN GOTO 560
920 me=stack-1:md=3
930 GOSUB 160
940 LOCATE#1,2,2:PRINT#1,p$(p,1);" DAY",
day
950 LOCATE#1,2,3:PRINT#1,"INCOMING REPOR
T"
960 IF e=1 THEN LOCATE#1,2,5:PRINT#1,"RE
PORT HAS BEEN SENT BUT":LOCATE#1,2,6:PRI
NT#1,"IT HAS NOT BEEN RECEIVED":GOTO 490
970 y=4:GOSUB 1020
980 GOTO 490
990 ' *****
1000 REM ** Decode Message **
1010 ' *****
1020 m1=ASC(LEFT$(s$(md,me),1))
1030 m2=ASC(MID$(s$(md,me),2,1))
1040 m3=ASC(MID$(s$(md,me),3,1))
1050 m4=ASC(MID$(s$(md,me),4,1))
1060 m5=ASC(MID$(s$(md,me),5,1))
1070 m6=ASC(MID$(s$(md,me),6,1))
1080 LOCATE#1,2,y:PEN#1,3:PRINT#1,STRING
$(25,255):PEN#1,1:y=y+1
1090 IF m1<10 THEN LOCATE#1,2,y:PRINT#1,
MID$(p$(m3,1),5);" TO ";x$(m2)

```



```

1100 IF m1>9 THEN LOCATE#1,2,9:PRINT#1,x
$(m2);" TO ";MID$(P$(m3,1),5)
1110 y=y+1
1120 ON m1 GOSUB 1170,1180,1190,1200,122
0,1230,1250,1260,1270,1280,1300,1320
1130 IF m1>12 THEN ON (m1-12) GOSUB 1340
,1360,1380,1390,1420
1140 y=y+1
1150 LOCATE#1,2,9:PRINT#1,STRING
$(25,255):PEN#1,1
1160 RETURN
1170 LOCATE#1,2,9:PRINT#1,"Go to ";w$(m5
):RETURN
1180 LOCATE#1,2,9:PRINT#1,"Kill ";x$(m5)
:RETURN
1190 LOCATE#1,2,9:PRINT#1,"Search ";w$(m
5):RETURN
1200 LOCATE#1,2,9:PRINT#1,"Steal ";r$(m5
):y=y+1
1210 LOCATE#1,2,9:PRINT#1,"From agent ";
x$(m6):RETURN
1220 LOCATE#1,2,9:PRINT#1,"Hide ";r$(m5)
:RETURN
1230 LOCATE#1,2,9:PRINT#1,"Transfer ";r$(
m5):y=y+1
1240 LOCATE#1,2,9:PRINT#1,"To agent ";x$(
m5):RETURN
1250 LOCATE#1,2,9:PRINT#1,"Change status
to ";q$(m5):RETURN
1260 LOCATE#1,2,9:PRINT#1,"Switch HOMING
-BEACON ";o$(m5):RETURN
1270 LOCATE#1,2,9:PRINT#1,"Explode MIND-
BOMB":RETURN
1280 LOCATE#1,2,9:PRINT#1,r$(m5);" captu
red":y=y+1
1290 LOCATE#1,2,9:PRINT#1,"in ";w$(m6):R
ETURN
1300 LOCATE#1,2,9:PRINT#1,r$(m5);" locat
ed":y=y+1
1310 LOCATE#1,2,9:PRINT#1,"in ";w$(m6):R
ETURN
1320 LOCATE#1,2,9:PRINT#1,"SCHWEINSTEIN
was seen on":y=y+1
1330 LOCATE#1,2,9:PRINT#1,"DAY";m5;"in "
,w$(m6):RETURN
1340 LOCATE#1,2,9:PRINT#1,"I have got ";
r$(m5):y=y+1
1350 LOCATE#1,2,9:PRINT#1,"From ";x$(m6)
:RETURN
1360 LOCATE#1,2,9:PRINT#1,"I have not go
t":y=y+1
1370 LOCATE#1,2,9:PRINT#1,r$(m5):RETURN
1380 LOCATE#1,2,9:PRINT#1,"Goods have be
en stolen":RETURN
1390 LOCATE#1,2,9:PRINT#1,"Message stopp
ed on DAY";m4:y=y+1
1400 IF md=1 THEN LOCATE#1,2,9:PRINT#1,"
Contents not available":RETURN
1410 me=m5:md=md-1:GOSUB 1020:RETURN
1420 LOCATE#1,2,9:PRINT#1,"Message Passe
d on DAY";m4:y=y+1
1430 IF md=1 THEN LOCATE#1,2,9:PRINT#1,"
Contents not available":RETURN
1440 me=m5:md=md-1:GOSUB 1020:RETURN
1450 ' *****
1460 REM ** Give Orders **
1470 ' *****
1480 i$="":m6=0:GOSUB 160:LOCATE#1,2,2:P
RINT#1,P$(P,1);" DAY";day
1490 LOCATE#1,2,4:PRINT#1,"OUTGOING ORDE
R'S"

```

```

1500 LOCATE#1,2,6:PRINT#1,"PRESS KEY FOR
YOUR AGENT"
1510 k$=INKEY$:IF k$="" THEN 1510
1520 IF k$=" " THEN RETURN
1530 IF k$="a" AND k$<="z" THEN m2=ASC(
k$)-96:GOTO 1560
1540 IF k$="0" AND k$<="9" THEN m2=ASC(
k$)-21:GOTO 1560
1550 GOTO 1510
1560 IF ASC(n$(m2,4))=0 THEN 1510
1570 LOCATE#1,2,7:PRINT#1,MID$(P$(P,1),5
);" TO ";x$(m2)
1580 LOCATE#1,2,8:PRINT#1,"KEY 1 Go to
city"
1590 LOCATE#1,2,9:PRINT#1,"KEY 2 Kill a
gent"
1600 LOCATE#1,2,10:PRINT#1,"KEY 3 Searc
h city"
1610 LOCATE#1,2,11:PRINT#1,"KEY 4 Steal
object"
1620 LOCATE#1,2,12:PRINT#1,"KEY 5 Hide
object"
1630 LOCATE#1,2,13:PRINT#1,"KEY 6 Trans
fer object"
1640 LOCATE#1,2,14:PRINT#1,"KEY 7 Chang
e status"
1650 LOCATE#1,2,15:PRINT#1,"KEY 8 Switc
h BEACON"
1660 LOCATE#1,2,16:PRINT#1,"KEY 9 Expl
ode MIND-BOMB"
1670 k$=INKEY$:IF k$="" THEN 1670
1680 IF k$<"1" OR k$>"9" THEN 1670
1690 m1=VAL(k$)
1700 FOR k=2 TO 9
1710 LOCATE#1,2,7+k:PRINT#1,"
1720 NEXT
1730 LOCATE#1,2,8:PRINT#1,"
";LOCATE#1,2,8
1740 ON VAL(k$) GOSUB 1840,1850,1860,187
0,1880,1890,1900,1910,1920
1750 m3=P:m4=day
1760 m$=CHR$(m1)+CHR$(m2)+CHR$(m3)+CHR$(
m4)+CHR$(m5)+CHR$(m6)
1770 IF P=2 THEN from=25 ELSE from=18
1780 type=0:de=ASC(n$(m2,1)):GOSUB 2360
1790 IF k$="X" THEN GOTO 1480
1800 RETURN
1810 ' *****
1820 REM ** Parameter Selection **
1830 ' *****
1840 PRINT#1,"Go to ...":GOTO 1930
1850 PRINT#1,"Kill ...":GOSUB 2020:m5=a9
:LOCATE#1,7,8:PRINT#1,x$(m5):RETURN
1860 m5=ASC(n$(m2,1)):PRINT#1,"Search ";
w$(m5):RETURN
1870 PRINT#1,"Steal ...":GOSUB 2110:LOCAT
E#1,8,8:PRINT#1,r$(m5):LOCATE#1,2,9:PRI
NT#1,"from agent ...":GOSUB 2020:m6=a9:L
OCATE#1,13,9:PRINT#1,x$(m6):RETURN
1880 PRINT#1,"Hide ...":GOSUB 2110:LOCAT
E#1,7,8:PRINT#1,r$(m5):RETURN
1890 PRINT#1,"Transfer ...":GOSUB 2110:L
OCATE#1,11,8:PRINT#1,r$(m5):LOCATE#1,2,9
:PRINT#1,"to agent ...":GOSUB 2020:m6=a9
:LOCATE#1,11,9:PRINT#1,x$(m6):RETURN
1900 PRINT#1,"Change status to ...":GOSU
B 2200:LOCATE#1,19,8:PRINT#1,q$(m5):RETU
RN
1910 PRINT#1,"Switch HOMING-BEACON ...":
GOSUB 2270:LOCATE#1,23,8:PRINT#1,o$(m5):
RETURN

```

TREACHERY

```

1920 PRINT#1,"Explode MIND-BOMB":RETURN
1930 l=8:m5=ASC(n$(m2,1)):FOR k=1 TO 8
1940 k$=MID$(y$(m5),k,1):IF k$=" " THEN
l=k-1:k=8:GOTO 1960
1950 LOCATE#1,2,10+k:PRINT#1,"KEY";k/w$(
ASC(k$)-64)
1960 NEXT
1970 k$=INKEY$:IF k$="" THEN 1970
1980 IF k$<"1" OR k$>RIGHT$(STR$(1),1)
THEN 1970
1990 GOSUB 2320
2000 m5=ASC(MID$(y$(m5),VAL(k$),1))-64:LO
CATE#1,8,8:PRINT#1,w$(m5):RETURN
2010 REM ** Key in Agent **
2020 LOCATE#1,2,11:PRINT#1,"PRESS KEY FO
R AGENT"
2030 k$=INKEY$:IF k$="" THEN 2030
2040 IF k$>"a" AND k$<"z" THEN a9=ASC(
k$)-96:GOTO 2070
2050 IF k$>"0" AND k$<"9" THEN a9=ASC(
k$)-21:GOTO 2070
2060 GOTO 2030
2070 IF ASC(n$(a9,4))=0 THEN 2030
2080 l=1:GOSUB 2320
2090 RETURN
2100 REM ** Key in Object **
2110 FOR k=1 TO 3
2120 LOCATE#1,2,10+k:PRINT#1,"KEY";k/n$(
k)
2130 NEXT
2140 k$=INKEY$:IF k$="" THEN 2140
2150 IF k$<"1" OR k$>"3" THEN 2140
2160 m5=VAL(k$)
2170 l=3:GOSUB 2320
2180 RETURN
2190 REM ** Key in Status **
2200 FOR k=1 TO 3
2210 LOCATE#1,2,10+k:PRINT#1,"KEY";k/q$(
k)
2220 NEXT
2230 k$=INKEY$:IF k$="" THEN 2230
2240 IF k$<"1" OR k$>"3" THEN 2230
2250 GOTO 2160
2260 REM ** Key in ON/OFF **
2270 LOCATE#1,2,11:PRINT#1,"KEY 1 ON":LO
CATE#1,2,12:PRINT#1,"KEY 2 OFF"
2280 k$=INKEY$:IF k$="" THEN 2280
2290 IF k$<"1" OR k$>"2" THEN 2280
2300 m5=VAL(k$)
2310 l=2
2320 FOR k=1 TO l+1:LOCATE#1,2,9+k:PRINT
#1," "
NEXT:RETU
RN
2330 ' *****
2340 REM ** Message Routing **
2350 ' *****
2360 LOCATE#1,2,10:PRINT#1,"Choose route
for message"
2370 LOCATE#1,2,11:PRINT#1,"by Pressing
agent keys"
2380 LOCATE#1,2,13:PRINT#1,"Destination
of message"
2390 LOCATE#1,2,14:PRINT#1,"will end rou
te entry"
2400 LOCATE#1,2,16:PRINT#1,"SPACE will c
ancel."
2410 LOCATE#1,2,17:PRINT#1,"Longest rout
e = 8 agents"
2420 k$=INKEY$:IF k$="" THEN 2420
2430 IF (k$<"0" OR k$>"9") AND (k$<"a" O
R k$>"z") AND k$<>" " THEN GOTO 2420

```

```

2440 IF k$=" " THEN k$="X":RETURN
2450 l=8:GOSUB 2320
2460 a$="":n=0:GOTO 2480
2470 k$=INKEY$:IF k$="" THEN 2470
2480 IF k$=" " THEN l=8:GOSUB 2320:GOTO
2360
2490 IF k$>"0" AND k$<"9" THEN k=ASC(k
$)-21:GOTO 2520
2500 IF k$>"a" AND k$<"z" THEN k=ASC(k
$)-96:GOTO 2520
2510 GOTO 2470
2520 IF ASC(n$(k,4))=0 THEN 2470
2530 c=ASC(n$(k,1)):LOCATE#1,2,10+n:PRIN
T#1,x$(k)," ";w$(c)
2540 n=n+1:a$=a$+CHR$(k):IF c=de AND (ty
Pe=1 OR k=m2) THEN GOTO 2580
2550 IF n=8 THEN LOCATE#1,2,9:PRINT#1,CH
R$(24);" THIS ROUTE IS TOO LONG ";CHR$(2
4):FOR k=1 TO 500:NEXT:l=8:GOSUB 2320:LO
CATE#1,2,9:PRINT#1,"
":GOTO 2360
2560 IF INKEY$<>" " THEN 2560
2570 GOTO 2470
2580 l=8:GOSUB 2320:LOCATE#1,2,10:PRINT#
1,"MESSAGE TO BE SENT VIA":FOR k=0 TO 2:
LOCATE#1,2,12+k
2590 FOR j=1 TO 3
2600 IF k*3+j>n THEN GOTO 2630
2610 a9=ASC(MID$(a$,k*3+j,1))
2620 PRINT#1,x$(a9);" ";
2630 NEXT:NEXT
2640 SOUND 7,200,25,5
2650 LOCATE#1,2,17:PRINT#1,CHR$(24);" KE
Y CLEAR, COPY OR ENTER";CHR$(24)
2660 IF INKEY$(9)<>-1 THEN GOSUB 7620:GOT
O 2700
2670 IF INKEY$(18)<>-1 THEN 2700
2680 IF INKEY$(47)<>-1 THEN RETURN
2690 GOTO 2660
2700 LOCATE#1,2,17:PRINT#1,"THE MESSAGE
IS BEING SENT"
2710 s$(3,stack)=m$
2720 ' *****
2730 REM **Message Interception **
2740 ' *****
2750 e=0:a$=i$a$:n=LEN(a$):FOR k=1 TO n
2760 a9=ASC(MID$(a$,k,1)):tt=ASC(n$(a9,1
))
2770 IF e=1 OR VAL(MID$(z$(from),tt,1))>>
1 THEN e=1:GOTO 2840
2780 orank=ASC(n$(a9,p+1)):erank=ASC(n$(
a9,4-p))
2790 IF orank>erank THEN GOTO 2840
2800 im=ASC(MID$(f$(2,a9),5,1)):IF im>0
AND ASC(MID$(f$(2,a9),im+5,1))=stack THE
N GOTO 2840
2810 im=im+1:f$(2,a9)=LEFT$(f$(2,a9),4)+
CHR$(im)+MID$(f$(2,a9),6)
2820 f$(2,a9)=LEFT$(f$(2,a9),im+4)+CHR$(
stack)+MID$(f$(2,a9),im+6)
2830 IF ASC(n$(a9,4))<>1 THEN e=1:IF k=n
AND ASC(LEFT$(m$,1))<10 AND ASC(n$(a9,4
))=2 THEN e=0
2840 from=tt
2850 NEXT
2860 IF ASC(LEFT$(m$,1))>9 OR e=1 THEN 2
900
2870 IF orank=0 THEN 2900
2880 IF orank<erank AND ASC(n$(a9,5))<>0
THEN 2900
2890 n$(a9,5)=CHR$(stack)

```



```

2900 stack=stack+1:RETURN
2910 ' *****
2920 REM ** Draw Flag **
2930 ' *****
2940 IF P=2 THEN PEN 3
2950 FOR k=1 TO 14
2960 LOCATE x,y+k-1:PRINT u$(P,k)
2970 NEXT
2980 PEN 1:RETURN
2990 ' *****
3000 REM ** Player Tune **
3010 ' *****
3020 '
3030 '
3040 ' *****
3050 REM ** UDG Data **
3060 ' *****
3070 DATA 255,255,253,248,241,225,246,25
5
3080 DATA 255,255,239,199,152,63,127,255
3090 DATA 255,255,239,247,251,251,251,12
3
3100 DATA 187,211,231,199,27,253,255,255
3110 ' *****
3120 REM ** Various Strings **
3130 ' *****
3140 DATA M16 LONDON CONTROL
3150 DATA KGB MOSCOW CENTRAL
3160 DATA SCHW.315b/QZ
3170 DATA LIQ/ROBOTNIK/S-20
3180 DATA ON/OFF
3190 DATA ASLEEP,AWAKE,ACTIVE
3200 DATA SCHWEINSTEIN,FIRING BUTTON,HOM
ING BEACON
3210 DATA 4,3,3,4,3,2,3,2
3220 DATA 2,3,2,3,2,1,2,1
3230 DATA 2,1,2,1,1,2,1,2
3240 DATA 1,2,1,2,1,0,1,0
3250 DATA 1,0,1,0,1,0,1,0
3260 DATA 1,0,1,0,0,1,0,1
3270 DATA 0,1,0,1,0,1,0,1
3280 DATA 0,1,0,1,0,0,0,0
3290 DATA 0,0,0,0,0,0,0,0
3300 ' *****
3310 REM ** Cities & Agents **
3320 ' *****
3330 DATA AMSTERDAM,ALPHA,BELGRADE,BRAVO
,VIENNA,CHARLIE,PARIS,DELTA,OSLO,ECHO,LI
SBON,FOXTROT,MADRID,GOLF,TANGIER,HOTEL,W
ARSAW,IVAN,ROME,JULIET
3340 DATA ATHENS,KING,ISTANBUL,LIMA,BUCH
AREST,MIKE,SOFIA,NOBLE,PRAGUE,OSCAR,COPE
NHAGEN,PAPA,REYKJAVIK,QUIZ,LONDON,ROMEO,
BRUSSELS,SIERRA,BERLIN,TANGO
3350 DATA HELSINKI,UNCLE,BUDAPEST,VICTOR
,DUBLIN,WINTER,ZURICH,X-RAY,MOSCOW,YANKE
E,STOCKHOLM,ZULU
3360 DATA ZERO,ONE,TWO,THREE,FOUR,FIVE,S
IX,SEVEN,EIGHT,NINE
3370 DATA "PTSR ", "VMNKJC ", "QVBJXT
", "STXJGR ", "UZPRWQ ", "WRGH ", "RDJ
KHF ", "KFG ", "YVOTPZ ", "CBKGDY ",
"NLHGJB ", "YKNM ", "YLNBY ", "MLKB
", "IVCT ", "EZITAR ", "UEW ", "EP
ASDGFW", "ATDR ", "PIOCKDSA", "YZEQ
", "IYMBCO "
3380 DATA "QERF ", "TCJD ", "LMVIZU
", "UYIPE "
3390 ' *****
3400 REM ** Game Start **
3410 ' *****

```

```

3420 BORDER 13:INK 0,26:INK 1,0:INK 2,2:
INK 3,6
3430 CLS:PEN 1
3440 LOCATE 12,1:PRINT" T R E A C H E R Y
"
3450 LOCATE 4,3:PRINT" When the ";PEN 3:
PRINT" KGB ";PEN 1:PRINT" and ";PEN 2:PR
INT" M16 ";PEN 1:PRINT" wage a war of "
3460 LOCATE 3,4:PRINT" wits through the c
apitals of Europe,"
3470 LOCATE 11,5:PRINT" nothing is simple
!"
3480 OPENOUT "dummy"
3490 MEMORY HIMEM-1
3500 CLOSEOUT
3510 ' *****
3520 REM ** User Defined Graphics **
3530 ' *****
3540 BORDER 9
3550 FOR char=251 TO 254
3560 FOR ch=1 TO 8
3570 READ a(ch)
3580 NEXT
3590 SYMBOL char,a(1),a(2),a(3),a(4),a(5
),a(6),a(7),a(8)
3600 NEXT
3610 SYMBOL 255,0,0,0,255,255,0,0,0
3620 ' *****
3630 REM ** 'Top Secret' Data **
3640 ' *****
3650 t$(1)=CHR$(139)+CHR$(129)+CHR$(135)
+CHR$(133)+CHR$(135)+CHR$(133)+ " "+CHR$(
135)+CHR$(129)+CHR$(135)+CHR$(129)+CHR$(
135)+CHR$(129)+CHR$(135)+CHR$(133)+CHR$(
135)+CHR$(129)+CHR$(139)+CHR$(129)
3660 t$(2)=CHR$(138)+ " "+STRING$(2,133)+
CHR$(135)+CHR$(129)+ " "+CHR$(131)+CHR$(
133)+CHR$(135)+ " "+CHR$(133)+ " "+CHR$(13
5)+CHR$(132)+CHR$(135)+ " "+CHR$(138)
3670 t$(3)=CHR$(130)+ " "+CHR$(131)+CHR$(
129)+CHR$(129)+ " "+CHR$(131)+CHR$(129)
+CHR$(131)+CHR$(129)+CHR$(131)+CHR$(129)
+CHR$(129)+CHR$(129)+CHR$(131)+CHR$(129)
+CHR$(130)
3680 ' *****
3690 REM ** Construct Flag Strings **
3700 ' *****
3710 REM "...is CTRL & 0"
3720 DIM u$(2,14):u$(1,1)="3"+CHR$(213)+
CHR$(143)+CHR$(215)+"2"+CHR$(213)+STRIN
G$(3,143)+ " "+"3"+STRING$(3,143)+ " "+"2"+
STRING$(3,143)+CHR$(212)+"3"+CHR$(214)+C
HR$(143)+CHR$(212)
3730 u$(1,2)="2"+CHR$(215)+"3"+CHR$(213)
+CHR$(143)+CHR$(215)+"2"+CHR$(213)+STRIN
G$(2,143)+ " "+"3"+STRING$(3,143)+ " "+"2"
+STRING$(2,143)+CHR$(212)+"3"+CHR$(214)+
CHR$(143)+CHR$(212)+"2"+CHR$(214)
3740 u$(1,3)="2"+CHR$(143)+CHR$(215)+"3"
+CHR$(213)+CHR$(143)+CHR$(215)+"2"+CHR$(
213)+CHR$(143)+ " "+"3"+STRING$(3,143)+ "
"+"2"+CHR$(143)+CHR$(212)+"3"+CHR$(214)+
CHR$(143)+CHR$(212)+"2"+CHR$(214)+CHR$(1
43)
3750 u$(1,4)="2"+CHR$(143)+CHR$(143)+CHR
$(215)+"3"+CHR$(213)+CHR$(143)+CHR$(215)
+"2"+CHR$(213)+ " "+"3"+STRING$(3,143)+ "
"+"2"+CHR$(212)+"3"+CHR$(214)+CHR$(143)+
CHR$(212)+"2"+CHR$(214)+STRING$(2,143)
3760 u$(1,5)="2"+STRING$(3,143)+CHR$(215)
+"3"+CHR$(213)+CHR$(143)+CHR$(215)+ " "+

```

TREACHERY

```

STRING$(3,143)+" "+CHR$(214)+CHR$(143)+CHR$(212)+"2"+CHR$(214)+STRING$(3,143)
3770 u$(1,6)=STRING$(8,"")+CHR$(1,9)=u$(1,6)
3780 u$(1,7)="3"+STRING$(19,143):u$(1,8)=u$(1,7)
3790 u$(1,10)="2"+STRING$(3,143)+CHR$(212)+"3"+CHR$(214)+CHR$(143)+CHR$(212)+" "+
+STRING$(3,143)+" "+CHR$(213)+CHR$(143)+CHR$(215)+"2"+CHR$(213)+STRING$(3,143)
3800 u$(1,11)="2"+STRING$(2,143)+CHR$(212)+"3"+CHR$(214)+CHR$(143)+CHR$(212)+"2"+
+CHR$(214)+"3"+" "+STRING$(3,143)+" "+2
+CHR$(215)+"3"+CHR$(213)+CHR$(143)+CHR$(215)+"2"+CHR$(213)+STRING$(2,143)
3810 u$(1,12)="2"+CHR$(143)+CHR$(212)+"3"+CHR$(214)+CHR$(143)+CHR$(212)+"2"+CHR$(214)+CHR$(143)+
+CHR$(215)+"3"+" "+STRING$(3,143)+" "+2+CHR$(143)+CHR$(215)+"3"+CHR$(213)+CHR$(143)+CHR$(215)+
+CHR$(215)
3820 u$(1,13)="2"+CHR$(212)+"3"+CHR$(214)+CHR$(143)+CHR$(212)+"2"+CHR$(214)+STRING$(2,143)+
+CHR$(2,143)+"3"+" "+STRING$(3,143)+" "+2+STRING$(2,143)+CHR$(215)+"3"+CHR$(213)+CHR$(143)+CHR$(215)+
+CHR$(215)+"2"+CHR$(213)
3830 u$(1,14)="3"+CHR$(214)+CHR$(143)+CHR$(212)+"2"+CHR$(214)+STRING$(3,143)+"3"+
+" "+STRING$(3,143)+" "+2+STRING$(3,143)+CHR$(215)+"3"+CHR$(213)+CHR$(143)+CHR$(215)
3840 l$=STRING$(19,143)
3850 FOR l=1 TO 14
3860 u$(2,l)=l$
3870 NEXT
3880 MID$(u$(2,2),16,2)=CHR$(251)+CHR$(253)
3890 MID$(u$(2,3),16,2)=CHR$(252)+CHR$(254)
3900 REM a$="":b$="":c$="":d$="":e$="":f$="":g$="":h$="":i$="":j$="":k$="":l$="":m$="":n$="":
3910 ' *****
3920 REM ** Print Flags **
3930 ' *****
3940 P=2:x=20:y=7
3950 GOSUB 2940
3960 P=1:x=3:y=11
3970 GOSUB 2940
3980 ' *****
3990 REM ** Read Various Strings **
4000 ' *****
4010 DIM P$(2,2)
4020 READ P$(1,1),P$(2,1)
4030 READ P$(1,2),P$(2,2)
4040 DIM o$(2),q$(3),r$(3)
4050 DIM s$(3,16)
4060 READ o$(1),o$(2)
4070 READ q$(1),q$(2),q$(3)
4080 READ r$(1),r$(2),r$(3)
4090 DIM n$(36,5),f$(2,36)
4100 FOR k=1 TO 26:n$(k,1)=CHR$(k):NEXT k
4110 FOR k=27 TO 31:n$(k,1)=CHR$(18):NEXT k
4120 FOR k=32 TO 36:n$(k,1)=CHR$(25):NEXT k
4130 FOR k=1 TO 36:FOR j=2 TO 5:n$(k,j)=CHR$(32):NEXT j:NEXT k
4140 m$="":FOR k=1 TO 13:m$=m$+CHR$(0):NEXT k

```

```

4150 RANDOMIZE TIME:FOR k=1 TO 36
4160 READ a,b
4170 c=INT(RND*36)+1:IF ASC(n$(c,2))<>32 THEN GOTO 4170
4180 n$(c,2)=CHR$(a):n$(c,3)=CHR$(b)
4190 f$(1,c)=CHR$(1)+m$:n$(c,4)=CHR$(1)
4200 IF (a=1 AND b=0) OR (a=0 AND b=1) THEN f$(1,c)=CHR$(3)+m$:n$(c,4)=CHR$(3)
4210 f$(2,c)=f$(1,c)
4220 n$(c,5)=CHR$(0)
4230 NEXT
4240 DIM g$(2),o$(6,5),s(4)
4250 BORDER 20
4260 ' *****
4270 REM ** Read Cities & Agents **
4280 ' *****
4290 DIM w$(26),x$(36)
4300 FOR k=1 TO 26
4310 READ w$(k),x$(k)
4320 NEXT
4330 FOR k=27 TO 36
4340 READ x$(k)
4350 NEXT
4360 DIM y$(26)
4370 FOR k=1 TO 26
4380 READ y$(k)
4390 NEXT
4400 BORDER 6
4410 ' *****
4420 REM ** City to City Distance **
4430 ' *****
4440 DIM z$(26):GOSUB 7800
4450 GOSUB 8100:REM **Screen Copy M/C **
4460 ' *****
4470 REM ** Game Start **
4480 ' *****
4490 BORDER 24:zzz=FRE(" ")
4500 day=3
4510 BORDER 26
4520 stack=1
4530 P=1
4540 r=INT(RND*26)+1:IF r=18 OR r=25 THEN N 4540
4550 s(1)=r:schwa9=0
4560 FOR j=1 TO 5:GOSUB 6210:NEXT j
4570 r=INT(RND*26)+1:IF r=18 OR r=25 THEN N 4570
4580 btag=0:btloc=r
4590 r=INT(RND*26)+1:IF r=18 OR r=25 OR r=btloc THEN 4590
4600 bca9=0:bcloc=r:bcon=2
4610 manloc=INT(RND*26)+1
4620 exPloc=0
4630 GOTO 4980
4640 ' *****
4650 REM ** Print Agent Ranks **
4660 ' *****
4670 GOSUB 160
4680 LOCATE#1,2,2:PRINT#1,MID$(P$(P,1),1,4):"RANK & AGENT DAY":day
4690 PEN#1,3
4700 y=4:x=2
4710 FOR k=1 TO 36
4720 IF ASC(n$(k,4))=0 OR ASC(n$(k,P+1))=0 THEN GOTO 4770
4730 a=ASC(n$(k,P+1))
4740 LOCATE#1,x,y:PRINT#1,a;" "x$(k)
4750 y=y+1
4760 IF y=15 THEN y=4:x=14
4770 NEXT
4780 PEN#1,1

```



```

4790 GOTO 490
4800 ' *****
4810 REM ** List Agent Reports **
4820 ' *****
4830 GOSUB 160
4840 LOCATE#1,2,2:PRINT#1,MID$(P$(P,1),1,4);"AGENT REPORTS DAY";day
4850 y=4:x=2
4860 FOR k=1 TO 36
4870 IF ASC(n$(k,4))=0 THEN GOTO 4930
4880 orank=ASC(n$(k,P+1))
4890 erank=ASC(n$(k,4-P))
4900 IF orank<=erank OR ASC(MID$(f$(1,k),5,1))=0 THEN GOTO 4930
4910 LOCATE#1,x,y:PRINT#1,x$(k);" ";ASC(MID$(f$(1,k),5,1))
4920 y=y+1:IF y=15 THEN y=4:x=14
4930 NEXT
4940 GOTO 490
4950 ' *****
4960 REM ** Player Turn Reports **
4970 ' *****
4980 rePorts=0:GOSUB 230
4990 IF rePorts=5 THEN GOTO 5200
5000 GOSUB 160
5010 IF day=3 THEN GOTO 5200
5020 LOCATE#1,2,2:PRINT#1,P$(P,1);" DAY"
;day
5030 LOCATE#1,2,3:PRINT#1,"YESTERDAY'S FIELD REPORTS"
5040 LOCATE#1,2,6:PRINT#1,"KEY 1 List all agents"
5050 LOCATE#1,2,7:PRINT#1," and the ir rank"
5060 LOCATE#1,2,9:PRINT#1,"KEY 2 List agents with"
5070 LOCATE#1,2,10:PRINT#1," report s to send and"
5080 LOCATE#1,2,11:PRINT#1," how ma ny reports"
5090 LOCATE#1,2,13:PRINT#1,"KEY 3 Call i n a report"
5100 LOCATE#1,2,15:PRINT#1,"KEY 4 Finish with reports"
5110 LOCATE#1,2,16:PRINT#1," Give t oday's orders"
5120 k$=INKEY$:IF k$="" THEN GOTO 5120
5130 IF k$<"1" OR k$>"4" THEN GOTO 5120
5140 IF k$="1" THEN GOSUB 4670:GOTO 4990
5150 IF k$="2" THEN GOSUB 4830:GOTO 4990
5160 IF k$="3" THEN GOSUB 560:rePorts=re Ports+1:GOTO 4990
5170 ' *****
5180 REM ** Player Turn Orders **
5190 ' *****
5200 orders=0
5210 IF orders=3 THEN GOTO 5350
5220 GOSUB 160
5230 LOCATE#1,2,2:PRINT#1,P$(P,1);" DAY"
;day
5240 LOCATE#1,2,4:PRINT#1,"TODAY'S ORDER S"
5250 LOCATE#1,2,6:PRINT#1,"KEY 1 List al l agents"
5260 LOCATE#1,2,7:PRINT#1," and the ir rank"
5270 LOCATE#1,2,9:PRINT#1,"KEY 2 Give an order"
5280 LOCATE#1,2,11:PRINT#1,"KEY 3 Finish with orders"
5290 LOCATE#1,2,12:PRINT#1," Operat ions over"
5300 LOCATE#1,2,13:PRINT#1," until tomorrow"
5310 k$=INKEY$:IF k$="" THEN 5310
5320 IF k$<"1" OR k$>"3" THEN 5310
5330 IF k$="1" THEN GOSUB 4670:GOTO 5210
5340 IF k$="2" THEN GOSUB 1480:orders=or ders+1:GOTO 5210
5350 P=P+1:IF P<3 THEN GOTO 4980
5360 ' *****
5370 REM ** End of Day **
5380 ' *****
5390 GOSUB 5550
5400 day=day+1
5410 FOR k=1 TO 16
5420 s$(1,k)=s$(2,k)
5430 s$(2,k)=s$(3,k)
5440 NEXT:stack=1
5450 m$="" :FOR k=1 TO 13:m$=m$+CHR$(0):N EXT
5460 FOR k=1 TO 36
5470 f$(1,k)=f$(2,k)
5480 f$(2,k)=n$(k,4)+m$
5490 NEXT
5500 P=1
5510 GOTO 4980
5520 ' *****
5530 REM ** Execute Orders **
5540 ' *****
5550 CLS
5560 GOSUB 160
5570 LOCATE#1,2,2:PRINT#1,"END OF DAY";d ay;CHR$(8);"'s TREACHERY"
5580 GOSUB 6210
5590 FOR k=1 TO 6:FOR j=1 TO 5:o(k,j)=0: NEXT: NEXT
5600 no=0
5610 FOR k=1 TO 36
5620 o1=ASC(n$(k,5)):IF o1=0 THEN GOTO 5 700
5630 no=no+1
5640 o(no,1)=ASC(LEFT$(s$(3,o1),1))
5650 o(no,2)=ASC(MID$(s$(3,o1),2,1))
5660 o(no,3)=ASC(MID$(s$(3,o1),5,1))
5670 o(no,4)=ASC(MID$(s$(3,o1),6,1))
5680 o(no,5)=ASC(MID$(s$(3,o1),3,1))
5690 n$(k,5)=CHR$(0)
5700 NEXT
5710 y=4
5720 FOR o=1 TO no:IF o(o,1)=9 THEN GOSU B 6350
5730 NEXT
5740 FOR o=1 TO no:IF o(o,1)=1 THEN GOSU B 6520
5750 NEXT
5760 FOR o=1 TO no:IF o(o,1)=5 THEN GOSU B 6630
5770 NEXT
5780 FOR o=1 TO no:IF o(o,1)=6 THEN GOSU B 6750
5790 NEXT
5800 FOR o=1 TO no:IF o(o,1)=3 THEN GOSU B 6980
5810 NEXT
5820 FOR o=1 TO no:IF o(o,1)=4 THEN GOSU B 7210
5830 NEXT
5840 FOR o=1 TO no:IF o(o,1)=2 THEN GOSU B 7290
5850 NEXT
5860 FOR o=1 TO no:IF o(o,1)=8 THEN GOSU

```

TREACHERY

```

B 7440
5870 NEXT
5880 FOR o=1 TO no:IF o(o,1)=7 THEN GOSU
B 7520
5890 NEXT
5900 FOR k=1 TO 36
5910 IF ASC(MID$(f$(2,k),2,1))<>0 THEN f
$(2,k)=LEFT$(f$(2,k),4)+CHR$(ASC(MID$(f$
(2,k),5,1))+1)+MID$(f$(2,k),6)
5920 NEXT
5930 LOCATE#1,2,y:PRINT#1,"3 days ago SC
HWEINSTEIN":y=y+1
5940 LOCATE#1,2,y:PRINT#1,"Was seen in "
;w$(s(4)):y=y+1
5950 IF bcon=2 THEN GOTO 5970
5960 LOCATE#1,2,y:PRINT#1,"BEACON ON in
";w$(bcloc):y=y+1
5970 IF bta9=0 THEN GOTO 6000
5980 LOCATE#1,2,y:PRINT#1,"FIRING BUTTON
detected":y=y+1
5990 LOCATE#1,2,y:PRINT#1,"in ";w$(btloc
):y=y+1
6000 IF exPloD=1 THEN GOTO 6120
6010 d=VAL(MID$(z$(manloc),bcloc,1))
6020 IF d=0 AND bcon=1 THEN GOTO 6110
6030 n=0:FOR k=1 TO 8
6040 IF MID$(y$(manloc),k,1)<>" " THEN n
=n+1
6050 NEXT
6060 r=INT(RND*n)+1
6070 IF bcon=2 THEN manloc=ASC(MID$(y$(m
anloc),r,1))-64:GOTO 6110
6080 c=ASC(MID$(y$(manloc),r,1))-64
6090 IF VAL(MID$(z$(c),bcloc,1))>=d THEN
GOTO 6060
6100 manloc=c
6110 LOCATE#1,2,y:PRINT#1,"MANDROID in "
;w$(manloc):y=y+1
6120 IF s(1)<>18 AND s(1)<>25 THEN GOTO
490
6130 IF s(1)=18 THEN LOCATE#1,2,y:PRINT#
1,"SCHWEINSTEIN IN LONDON":y=y+2:LOCATE#
1,2,y:PRINT#1,"MI6 HAVE WON THE GAME":P=
1
6140 IF s(1)=25 THEN LOCATE#1,2,y:PRINT#
1,"SCHWEINSTEIN IN MOSCOW":y=y+2:LOCATE#
1,2,y:PRINT#1,"THE KGB HAVE WON THE GAME
":P=2
6150 REM
6160 GOTO 6160
6170 STOP
6180 ' *****
6190 REM ** Move Schweinstein **
6200 ' *****
6210 FOR k=4 TO 2 STEP-1
6220 s(k)=s(k-1)
6230 NEXT
6240 IF schwa9>0 THEN RETURN
6250 n=0:FOR k=1 TO 8
6260 IF MID$(y$(s(1)),k,1)<>" " THEN n=n+1
6270 NEXT
6280 r=INT(RND*n)+1
6290 c=ASC(MID$(y$(s(1)),r,1))-64
6300 IF c=s(3) OR c=18 OR c=25 THEN 6280
6310 s(1)=c:RETURN
6320 ' *****
6330 REM ** Explode Mindbomb **
6340 ' *****
6350 IF exPloD=1 THEN RETURN
6360 a9=o(o,2):IF bta9<>a9 THEN ob=2:GOT
0 6900

```

```

6370 LOCATE#1,2,y:PRINT#1,"MIND-BOMB exp
lodes in":y=y+1
6380 LOCATE#1,2,y:PRINT#1,w$(manloc):y=
y+1
6390 IF manloc=18 THEN y=y+1:LOCATE#1,2,
y:PRINT#1,"THE KGB HAVE WON THE GAME":P=
2:GOTO 6150
6400 IF manloc=25 THEN y=y+1:LOCATE#1,2,
y:PRINT#1,"MI6 HAVE WON THE GAME":P=1:GO
TO 6150
6410 FOR k=1 TO 36
6420 IF ASC(n$(k,1))<>manloc THEN 6470
6430 n$(k,4)=CHR$(0)
6440 IF bta9=k THEN bta9=0
6450 IF bca9=k THEN bca9=0
6460 IF schwa9=k THEN schwa9=0
6470 NEXT
6480 exPloD=1:RETURN
6490 ' *****
6500 REM ** Go to City **
6510 ' *****
6520 a9=o(o,2):c=o(o,3)
6530 IF ASC(n$(a9,4))=0 THEN RETURN
6540 n$(a9,1)=CHR$(c)
6550 IF bta9=a9 THEN btloc=c
6560 IF bca9=a9 THEN bcloc=c
6570 IF schwa9=a9 THEN s(1)=c
6580 LOCATE#1,2,y:PRINT#1,x$(a9);" go to
";w$(c)
6590 y=y+1:RETURN
6600 ' *****
6610 REM ** Hide Object **
6620 ' *****
6630 a9=o(o,2):ob=o(o,3)
6640 IF ASC(n$(a9,4))=0 THEN RETURN
6650 ON ob GOTO 6660,6680,6700
6660 IF schwa9<>a9 THEN GOTO 6900
6670 schwa9=0:RETURN
6680 IF bta9<>a9 THEN GOTO 6900
6690 bta9=0:RETURN
6700 IF bca9<>a9 THEN GOTO 6900
6710 bca9=0:RETURN
6720 ' *****
6730 REM ** Transfer Object **
6740 ' *****
6750 a9=o(o,2):ob=o(o,3):to1=o(o,4):IF a
9=to1 THEN RETURN
6760 IF ASC(n$(a9,4))=0 OR ASC(n$(to1,4)
)=0 THEN RETURN
6770 fc=ASC(n$(a9,1)):tc=ASC(n$(to1,1))
6780 IF MID$(z$(fc),tc,1)>"1" THEN RETUR
N
6790 ON ob GOTO 6800,6830,6860
6800 IF schwa9<>a9 THEN GOTO 6900
6810 schwa9=to1:s(1)=tc
6820 GOTO 6880
6830 IF bta9<>a9 THEN GOTO 6900
6840 bta9=to1:btloc=tc
6850 GOTO 6880
6860 IF bca9<>a9 THEN GOTO 6900
6870 bca9=to1:bcloc=tc
6880 f$(2,to1)=LEFT$(f$(2,to1),1)+CHR$(1
3)+CHR$(ob)+CHR$(a9)+MID$(f$(2,to1),5)
6890 RETURN
6900 IF o(o,1)=4 THEN GOTO 6930
6910 f$(2,a9)=LEFT$(f$(2,a9),1)+CHR$(14)
+CHR$(ob)+MID$(f$(2,a9),4)
6920 RETURN
6930 f$(2,to1)=LEFT$(f$(2,to1),1)+CHR$(1
4)+CHR$(ob)+MID$(f$(2,to1),4)
6940 RETURN

```



```

6950 ' *****
6960 REM ** Search City **
6970 ' *****
6980 a9=o(o,2):c=o(o,3)
6990 IF ASC(n$(a9,4))=0 THEN RETURN
7000 s1=0:FOR k=3 TO 2 STEP-1
7010 IF s(k)=c THEN s1=k
7020 NEXT
7030 IF s1=0 THEN GOTO 7050
7040 f$(2,a9)=LEFT$(f$(2,a9),1)+CHR$(12)+
CHR$(day+1-s1)+CHR$(c)+MID$(f$(2,a9),5)
7050 IF s(1)<>c THEN GOTO 7100
7060 IF schwa9=0 THEN schwa9=a9:f$(2,a9)=
LEFT$(f$(2,a9),1)+CHR$(10)+MID$(f$(2,a9),3):GOTO 7080
7070 f$(2,a9)=LEFT$(f$(2,a9),1)+CHR$(11)+
MID$(f$(2,a9),3)
7080 f$(2,a9)=LEFT$(f$(2,a9),2)+CHR$(1)+
CHR$(c)+MID$(f$(2,a9),5)
7090 RETURN
7100 IF btloc<>c OR bta9<>0 THEN GOTO 7130
7110 f$(2,a9)=LEFT$(f$(2,a9),1)+CHR$(10)+
CHR$(2)+CHR$(c)+MID$(f$(2,a9),5)
7120 bta9=a9
7130 IF bcloc<>c OR bcon=1 THEN RETURN
7140 IF bca9=0 THEN bca9=a9:f$(2,a9)=LEF
T$(f$(2,a9),1)+CHR$(10)+MID$(f$(2,a9),3)
7150 f$(2,a9)=LEFT$(f$(2,a9),1)+CHR$(11)+
MID$(f$(2,a9),3)
7160 f$(2,a9)=LEFT$(f$(2,a9),2)+CHR$(3)+
CHR$(c)+MID$(f$(2,a9),5)
7170 RETURN
7180 ' *****
7190 REM ** Steal Object **
7200 ' *****
7210 to1=o(o,2):ob=o(o,3):a9=o(o,4):IF a
9=to1 THEN RETURN
7220 GOSUB 6760
7230 IF ASC(MID$(f$(2,to1),2,1))<>13 THE
N RETURN
7240 f$(2,a9)=LEFT$(f$(2,a9),1)+CHR$(15)+
MID$(f$(2,a9),3)
7250 RETURN
7260 ' *****
7270 REM ** Kill Agent **
7280 ' *****
7290 a9=o(o,2):to1=o(o,3)
7300 IF a9=to1 THEN RETURN
7310 IF ASC(n$(a9,4))=0 OR ASC(n$(to1,4))=0 THEN RETURN
7320 c=ASC(n$(a9,1))
7330 IF ASC(n$(to1,1))<>c THEN RETURN
7340 n$(to1,4)=CHR$(0)
7350 IF bta9=to1 THEN bta9=0
7360 IF bca9=to1 THEN bca9=0
7370 IF schwa9=to1 THEN schwa9=0
7380 LOCATE#1,2,y:PRINT#1,x$(to1);" is e
liminated"
7390 y=y+1
7400 RETURN
7410 ' *****
7420 REM ** Switch Beacon **
7430 ' *****
7440 a9=o(o,2):on1=o(o,3)
7450 IF ASC(n$(a9,4))=0 THEN RETURN
7460 IF bca9<>a9 THEN ob=3:GOTO 6900
7470 bcon=on1
7480 RETURN
7490 ' *****

7500 REM ** Change Status **
7510 ' *****
7520 a9=o(o,2):p1=o(o,5)
7530 IF ASC(n$(a9,4))=0 THEN RETURN
7540 orank=ASC(n$(a9,p1+1))
7550 erank=ASC(n$(a9,4-p1))
7560 IF erank>orank THEN RETURN
7570 n$(a9,4)=CHR$(o(o,3))
7580 RETURN
7590 ' *****
7600 REM ** Hard Copy to Printer **
7610 ' *****
7620 PRINT#8,CHR$(13):a%=0
7630 FOR k2=5 TO 19
7640 FOR k1=11 TO 36
7650 LOCATE k1,k2
7660 CALL coPychar,@a%
7670 PRINT#8,CHR$(a%);
7680 NEXT
7690 PRINT#8
7700 NEXT
7710 RETURN
7720 ' *****
7730 REM ** Error Trapping **
7740 ' *****
7750 LOCATE 1,24:PRINT"Error",ERR;"in LI
NE";ERL
7760 END
7770 ' *****
7780 REM ** Z$(array) DATA **
7790 ' *****
7800 RESTORE 7810:FOR k=1 TO 26:READ z$(
k):NEXT:RETURN
7810 DATA "03222223233444213111332232"
7820 DATA "30124322211211234332314223"
7830 DATA "21023323212322124321314123"
7840 DATA "22202212212333223111332133"
7850 DATA "24320223233334311122131321"
7860 DATA "23322011322343422123341343"
7870 DATA "22212101311232323122332233"
7880 DATA "32323110421232433233432334"
7890 DATA "22222334033223113221213211"
7900 DATA "31113212301222234222423134"
7910 DATA "31223211310121334233323223"
7920 DATA "42333322221011333343224312"
7930 DATA "41233433222101233443214312"
7940 DATA "41234322321110344343324323"
7950 DATA "22123434123323024321314222"
7960 DATA "13221223133334202121222221"
7970 DATA "34431233344334420233131422"
7980 DATA "13311112222343312012231232"
7990 DATA "13212223223444223101332233"
8000 DATA "12112323123333113210323122"
8010 DATA "33331334243223321233022411"
8020 DATA "31133433122212123332204212"
8030 DATA "24421122333444421123240332"
8040 DATA "22113323212333224221423033"
8050 DATA "32232433132112222332113301"
8060 DATA "23331334143223212232122310"
8070 ' *****
8080 REM ** Screen CoPy Machine Code **
8090 ' *****
8100 RESTORE 8160
8110 MEMORY HIMEM-12
8120 coPychar=HIMEM+1
8130 FOR address=coPychar TO coPychar+10
8140 READ byte:POKE address,byte
8150 NEXT
8160 DATA &cd,&60,&bb,&dd,&6e,&00,&dd,&6
6,&01,&77,&c9
8170 RETURN

```

EGGS • EGGS

• BBC B or ELECTRON

Take control of the madcap Farmer Giles and make sure he doesn't end up with egg on his face.

Collecting his hens' freshly laid eggs is no easy task. If he is not alert the eggs end up smashed on the floor, making a meal for the farm pig.

Points are scored for every egg caught — and one lost for every egg dropped.

Oh yes, watch out for the rotten eggs. Don't let them hit you.

Control keys are Z — left, X — right and Return — jump.

• Paul Dyer

```
>LIST
10 MODE 6
20 VDU 23.1.0:0:0:0:
30 PROCInstr
40 MODE 5
50 VDU 23.1.0:0:0:0:
60 PROctt1
70 COLOUR 3
80 ENVELOPE 1.2.2.-3.1.40.20.20.0.0.
0.0.0.0
90 ENVELOPE 2.1.2.-2.1.40.20.40.0.0.
0.0.0.0
100 ENVELOPE 3.1.100.10.-110.10.100.1
10.0.0.0.0.0
110 ENVELOPE 4.3.4.0.0.50.0.0.0.0.0.0
.0.0
120 ENVELOPE 5.3.-4.0.0.50.0.0.0.0.0.0
0.0.0
130 DIM HI(5),HIS(5)
140 FOR H=1 TO 5:HI(H)=((6-H)*(6-H))*
10:HI(H)="FARMER GILES":NEXT
150 ON ERROR GOSUB 5920
160 PROCprh1
170 VDU 23.1.0:0:0:0:
180 SO=0:SC=0:HO=1:TEC=0:TED=0:PROC=
cr:PROChegr
190 PROCvar:PROCflg:PROCl:PROChens
200 LA=1:PROCflap:LA=5:PROCflap:EG=0
210 TIME=0
220 IF EG=1 THEN ED=18
230 IF RO=1:IF J=0 AND (ROG+1)=A THEN
RE=1:COLOUR 2:PRINT TAB(ROG-1.27) " ":T
AB(ROG.27):CHR#145:PROCend
240 IF RO=1:IF J=0 AND ROG=A THEN RE=
1:COLOUR 2:PRINT TAB(ROG-1.27) " ":TAB(R
OG.27):CHR#145:PROCend
250 IF EG=1 AND AS="1":IF (ED)=UP) AN
D (C=A) THEN RE=3:PROCend
260 IF EG=1 AND AS="r":IF (ED)=UP) AN
D (C=A-1) THEN RE=3:PROCend
270 IF INKEY(-98) THEN T=TIME:PROCl:T
IME=T:GOTO 290
280 IF INKEY(-67) THEN T=TIME:PROCr:T
IME=T
290 IF J=1:IF TIME>TJ+50 THEN T=TIME:
TIME=T:PROCdown:J=0
300 IF J=0:IF INKEY(-74) THEN T=TIME:
PROCjump:J=1:TIME=T
310 IF J=0 THEN IF (A)=PIA AND A<PIA
+2) OR (A+1)=PIA AND A+1<PIA+2):RE=5:PR
OCend
320 IF RO=1:IF J=0 AND (ROG=A OR ROG=
A+1) THEN RE=1:COLOUR 2:PRINT TAB(ROG.27
):" ":TAB(ROG.27):CHR#145:PROCend
330 IF J=0:IF INKEY(-74) THEN T=TIME:
PROCjump:J=1:TIME=T
340 IF J=0 THEN T=TIME:PROClpd:TIME=
T
350 IF TIME>6000 THEN PROCnuly
360 GCOL 0.0:TI=(TIME/10):MOVE 762-(T
I/2).890:DRAW 762-(TI/2).920
370 IF 300-(TIME/20)>45 AND 300-(TIME
/20)<50 THEN SOUND 1.3.230.8
380 IF EG=1:IF ED>18.9 THEN PRINT TAB
(C+1.ED-1) " "
390 IF EG=1:IF ED>20 THEN PRINT TAB(C
+1.ED-2) " "
400 IF EG=1:COLOUR 3:PRINT TAB(C+1.ED
):CHR#145
410 IF EG=0:EGG=RND(4):IF EGG=3 THEN
T=TIME:EG=1:PROCflap:TIME=T:GOTO 220
420 IF EG=0:FA=RND(8):IF FA=2 THEN T=
TIME:PROCflap:PROChens:TIME=T
430 IF (EG=1) AND (AS="1"):IF (A=C+1)
AND (ED)=UP+1 AND ED<UP+2) THEN T=TIME:
EGS=EGS+1:EG=0:SOUND 1.3.120.2:PRINT TAB
(C+1.UP+1) " ":PROCflap:TIME=T
440 IF (EG=1) AND (AS="r"):IF (A=C) A
ND (ED)=UP+1 AND ED<UP+2) THEN T=TIME:EG
S=EGS+1:EG=0:SOUND 1.3.120.2:PRINT TAB(C
+1.UP+1) " ":PROCflap:TIME=T
```

```
450 COLOUR 3:PRINT TAB(9-LEN (STR$(EG
S)).5):EGS
460 IF RO=0 THEN IF HO<4:REG=RND(6):I
F REG=1 THEN ROG=-1:RO=1
470 IF RO=0 THEN IF HO>3:REG=RND(4):I
F REG=1 THEN ROG=-1:RO=1
480 IF RO=1 THEN IF ROG=-1:ROG=ROG+1:
GOTO 500
490 IF RO=1 THEN PRINT TAB(ROG.27) " "
:ROG=ROG+1:IF ROG>19 THEN ROG=-1:RO=0
500 IF RO=1 THEN COLOUR 2:PRINT TAB(R
OG.27):CHR#145
510 COLOUR 2
520 IF RO=1:IF J=0 AND ROG=A THEN RE=
1:COLOUR 2:PRINT TAB(ROG-1.27):CHR#145:PR
OCend
530 IF J=0 THEN IF (A)=PIA AND A<PIA
+2) OR (A+1)=PIA AND A+1<PIA+2):RE=5:PR
OCend
540 COLOUR 3
550 IF INKEY(-56) THEN T=TIME:REPEAT:
PRINT TAB(14.7)"PAUSED":UNTIL INKEY(-55)
:TIME=T:PRINT TAB(14.7) " "
560 IF INKEY(-17) THEN PRINT TAB(0.7)
"QUITE":SO=1:FX 210.1
570 IF INKEY(-82) THEN PRINT TAB(0.7)
" ":SO=0:FX 210.0
580 PGO=RND(3):IF PGO=1:IF PGO=0 THEN
T=TIME:PROChpi:TIME=T
590 IF PG=1 AND BS="1" THEN T=TIME:PR
OCpif:TIME=T
600 IF PG=2 AND BS="r" THEN T=TIME:PR
OCpirt:TIME=T
610 IF J=0 THEN IF (A)=PIA AND A<PIA
+2) OR (A+1)=PIA AND A+1<PIA+2):RE=5:PR
OCend
620 COLOUR 3
630 IF EG=1 THEN ED=ED+ST:IF (ED)>29-S
T) THEN T=TIME:PROCamash:EG=0:EGD=EGD+1:
PRINT TAB(20-LEN (STR$(EGD)).5):EGD:PROC
slp:TIME=T
640 IF EG=1 THEN GOTO 230
650 GOTO 220
660 END
670 DEFPROctt1
680 VDU 5:GCOL 0.3:VDU 19.3.0:0:0:MOVE
0.32:PRINT "EGGS"
690 VDU 23.128.0.0.0.0.0.56.60.28
700 FOR L1=4 TO 32 STEP 4
710 FOR L2=0 TO 64 STEP 8
720 IF POINT(L2,L1)=3 THEN GCOL 0
.3:MOVE 340-(L2*2).955-L1*2:PRINT CHR#12
8:GCOL 0.1:MOVE 335-(L2*2).950-L1*2:PRIN
T CHR#128
730 NEXT L
740 GCOL 0.0:MOVE 0.32:PRINT "EGGS"
:VDU 19.3.7:0:VDU 4
750 ENDPROC
760 DEFPROCscr
770 COLOUR 3:PRINT TAB(3.7)"HEN HOU
SE 000":TAB(16-LEN (STR$ HO).7)HO:COLOUR
1:PRINT TAB(4.9)"TOTAL EGGS":COLOUR 3:
PRINT TAB(2.11)"IN BASKET 0000":TAB(17-L
EN (STR$ TEC).11)TEC:TAB(3.13)"DROPPED 0
000":TAB(16-LEN (STR$ TED).13)TED
780 COLOUR 2:PRINT TAB(4.15)"SCORE
00000":TAB(15-LEN (STR$ SCT).15)SCT
790 IF SO=0 THEN FOR WA=50 TO 200 S
TEP 2.5:SOUND 1.4.WA.1:NEXT ELSE FOR WA=
0 TO 4000:NEXT
800 VDU 28.0.30.19.5:CLS:VDU 26
810 VDU 23.128.136.136.136.119.34.3
4.34.221
820 VDU 23.129.254.252.248.247.226.
226.162.221
830 VDU 23.130.255.191.159.127.39.3
5.35.221
840 FOR L3=5 TO 1 STEP -1
850 COLOUR 129:COLOUR 0:PRINT TAB
(L3-1.9-(5-L3):CHR#129:STRING$(19-(L3*2)
. CHR#128):CHR#130
```

```
860 NEXT
870 COLOUR 129:COLOUR 0
880 PRINT TAB(19.25):CHR#129
890 VDU 23.128.170.171.187.255.255.
255.255.255
900 COLOUR 128:COLOUR 2:PRINT TAB(0
.29)STRING$(20,CHR#128)
910 VDU 23.128.253.247.52.52.52.52.
247.253
920 COLOUR 128
930 COLOUR 3:PRINT TAB(6.8):CHR#128:
" ":CHR#128
940 COLOUR 1:PRINT TAB(5.6) " "
950 COLOUR 129:COLOUR 0:PRINT TAB(5
.7)"HEN HOUSE":COLOUR 128
960 COLOUR 3:PRINT TAB(0.5)"CAUGHT
OO DROPPED OO"
970 FOR LA=460 TO 760 STEP 4
980 SOUND 1.3.(LA-450)/4.1
990 IF LA<510 THEN GCOL 0.1
1000 IF LA>510 AND LA<570 THEN GCOL
0.2
1010 IF LA>570 THEN GCOL 0.3
1020 MOVE LA.890:DRAW LA.920
1030 NEXT
1040 ENDPROC
1050 DEFPROChegr
1060 VDU 23.128.170.170.127.127.63.3
1.15.3
1070 VDU 23.129.165.165.255.255.255.
255.255.255
1080 VDU 23.130.85.85.254.254.252.24
8.240.192
1090 VDU 23.131.1.2.4.8.28.42.0.0
1100 VDU 23.132.128.64.32.16.56.84.0
.0
1110 VDU 23.133.0.0.1.1.57.61.127.12
7
1120 VDU 23.134.126.255.153.189.255.
129.66.195
1130 VDU 23.135.0.0.128.128.156.188.
254.254
1140 VDU 23.136.127.63.63.31.15.4.7.
7
1150 VDU 23.137.231.255.255.255.126.
255.255.255
1160 VDU 23.138.254.252.252.248.240.
32.224.224
1170 VDU 23.139.126.60.60.24.0.0.0.0
1180 VDU 23.140.6.102.60.24.0.0.0.0
1190 VDU 23.141.0.0.1.1.9.13.31.31
1200 VDU 23.142.0.0.128.128.144.176.
248.248
1210 VDU 23.143.31.15.15.7.7.4.7.7
1220 VDU 23.144.248.240.240.224.224.
32.224.224
1230 VDU 23.145.56.126.255.255.255.1
26.56.0
1240 ENDPROC
1250 DEFPROCchens
1260 COLOUR 7
1270 IF HO=1 THEN CH=2
1280 IF HO=2 THEN CH=3
1290 IF HO=3 THEN CH=4
1300 IF HO=3 THEN CH=5
1310 IF CH=2 THEN PROClpr
1320 IF CH=3 THEN PROClpr:PROCr
1330 IF CH=4 THEN PROClpr:PROClpr
1340 IF CH=5 THEN PROClpr:PROClpr:PRO
Cm
1350 ENDPROC
1360 DEFPROCpr
1370 COLOUR 2
1380 PRINT TAB(0.17):CHR#128:CHR#129:
CHR#130
1390 PRINT TAB(16.17):CHR#128:CHR#129
:CHR#130
1400 PRINT TAB(0.18):CHR#131:" ":CHR#
132
1410 PRINT TAB(16.18):CHR#131:" ":CHR
```



```

1132
1420 COLOUR 3
1430 PRINT TAB(0.15)CHR#133:CHR#134:
CHR#135
1440 PRINT TAB(0.16)CHR#136:CHR#137:
CHR#138
1450 PRINT TAB(16.15)CHR#133:CHR#134
:CHR#135
1460 PRINT TAB(16.16)CHR#136:CHR#137
:CHR#138
1470 VDU 5:GCOL 0.2:MOVE 68.520:PRIN
T CHR#139:MOVE 1088.520:PRINT CHR#139:VD
U 4
1480 VDU 5:GCOL 0.1:MOVE 68.554:PRIN
T CHR#140:MOVE 1088.554:PRINT CHR#140:VD
U 4
1490 ENDPROC
1500 DEFPROCmlr
1510 COLOUR 2
1520 PRINT TAB(4.17)CHR#128:CHR#129:
CHR#130
1530 PRINT TAB(12.17)CHR#128:CHR#129
:CHR#130
1540 PRINT TAB(4.18)CHR#131: " :CHR#
132
1550 PRINT TAB(12.18)CHR#131: " :CHR
#132
1560 COLOUR 3
1570 PRINT TAB(4.15)CHR#133:CHR#134:
CHR#135
1580 PRINT TAB(4.16)CHR#136:CHR#137:
CHR#138
1590 PRINT TAB(12.16)CHR#136:CHR#137
:CHR#138
1600 PRINT TAB(12.15)CHR#133:CHR#134
:CHR#135
1610 VDU 5:GCOL 0.2:MOVE 320.520:PRI
NT CHR#139:MOVE 838.520:PRINT CHR#139:VD
U 4
1620 VDU 5:GCOL 0.1:MOVE 320.554:PRI
NT CHR#140:MOVE 838.554:PRINT CHR#140:VD
U 4
1630 ENDPROC
1640 DEFPROCm
1650 COLOUR 2
1660 PRINT TAB(8.17)CHR#128:CHR#129:
CHR#130
1670 PRINT TAB(8.18)CHR#131: " :CHR#
132
1680 COLOUR 3
1690 PRINT TAB(8.15)CHR#133:CHR#134:
CHR#135
1700 PRINT TAB(8.16)CHR#136:CHR#137:
CHR#138
1710 VDU 5:GCOL 0.2:MOVE 580.520:PRI
NT CHR#139:VDU 4
1720 VDU 5:GCOL 0.1:MOVE 580.554:PRI
NT CHR#140:VDU 4
1730 ENDPROC
1740 DEFPROCflap
1750 IF LA=1 THEN C=0
1760 IF LA=2 THEN C=4
1770 IF LA=3 THEN C=8
1780 IF LA=4 THEN C=12
1790 IF LA=5 THEN C=16
1800 COLOUR 3
1810 PRINT TAB(C.15)CHR#141:TAB(C+2.
15)CHR#142
1820 PRINT TAB(C.16)CHR#143:TAB(C+2.
16)CHR#144
1830 IF EG=1:SOUND 1.1.90.2:SOUND 1.
2.30.1:SOUND 1.0.0.1:SOUND 1.1.90.2:SOUN
D 1.2.30.1
1840 IF EG=0:SOUND 1.1.80.2:SOUND 1.
2.30.1:SOUND 1.0.0.1:SOUND 1.1.80.2:SOUN
D 1.2.30.1
1850 PRINT TAB(C.15)CHR#133:TAB(C+2.
15)CHR#135
1860 PRINT TAB(C.16)CHR#136:TAB(C+2.
16)CHR#138
1870 PROCntch
1880 ENDPROC
1890 DEFPROCflg
1900 VDU 23.146.0.1.15.10.0.3.3.0
1910 VDU 23.147.240.248.255.149.144.
240.240.48
1920 VDU 23.148.0.0.0.0.0.3.63.59
1930 VDU 23.149.240.96.224.224.240.2
40.240.240
1940 VDU 23.150.0.0.0.0.0.0.3.7
1950 VDU 23.152.1.3.0.0.0.0.0.0
1960 VDU 23.153.8.8.127.127.62.62.28
.0
1970 ENDPROC
1980 DEFPROCfalf
1990 COLOUR 1
2000 VDU 5:MOVE (((A+1)*2)*32)-6.(((3
1-UP)*32)-32:GCOL 0.0:PRINT CHR#153:VDU
4
2010 PRINT TAB(A,UP)CHR#146:CHR#147
2020 PRINT TAB(A,UP+1)CHR#148:CHR#14
9
2030 PRINT TAB(A,UP+2)CHR#150:CHR#15
1
2040 VDU 5:MOVE (((A+2)*32)+24.(((3
1-UP)*32):GCOL 0.3:PRINT CHR#152:VDU 4
2050 VDU 5:MOVE (((A+2)*32)-6.(((31-UP
)*32)-32:GCOL 0.2:PRINT CHR#153:VDU 4
2060 ENDPROC
2070 DEFPROC1
2080 PRINT TAB(A,UP) " :TAB(A,UP+1)
" :TAB(A,UP+2) "
2090 IF B=1 THEN B=2:VDU 23.151.224.
243.255.247.67.195.225.224:GOTO 2110
2100 IF B=2 THEN B=1:VDU 23.151.224.
224.224.224.64.192.224.224
2110 IF A="r" THEN PROCflg:A="l"
2120 IF A>0 THEN A=A-1
2130 PROCfalf
2140 ENDPROC
2150 DEFPROCfrg
2160 VDU 23.146.15.31.255.169.9.15.1
5.12
2170 VDU 23.147.0.128.240.80.0.192.1
92.0
2180 VDU 23.148.15.6.7.7.15.15.15.15
2190 VDU 23.149.0.0.0.0.0.192.252.22
0
2200 VDU 23.151.0.0.0.0.0.0.192.224
2210 VDU 23.152.2.3.0.0.0.0.0.0
2220 ENDPROC
2230 DEFPROCfart
2240 COLOUR 1
2250 VDU 5:MOVE (((A+2)*32)+24.(((31
-UP)*32)-32:GCOL 0.0:PRINT CHR#152:VDU 4
2260 PRINT TAB(A,UP)CHR#146:CHR#147
2270 PRINT TAB(A,UP+1)CHR#148:CHR#14
9
2280 PRINT TAB(A,UP+2)CHR#150:CHR#15
1
2290 VDU 5:MOVE (((A+2)*32)-6.16.(((31
-UP)*32):GCOL 0.3:PRINT CHR#152:VDU 4
2300 VDU 5:MOVE (((A+1)*2)*32).(((31-
UP)*32)-32:GCOL 0.2:PRINT CHR#153:VDU 4
2310 ENDPROC
2320 DEFPROCpr
2330 PRINT TAB(A,UP) " :TAB(A,UP+1)
" :TAB(A,UP+2) "
2340 IF B=1 THEN B=2:VDU 23.150.7.19
9.255.239.194.195.135.7:GOTO 2360
2350 IF B=2 THEN B=1:VDU 23.150.7.7.
7.7.2.3.7.7
2360 IF A="l" THEN PROCfrg:A="r"
2370 IF A<17 THEN A=A+1
2380 PROCfart
2390 ENDPROC
2400 DEFPROCntch
2410 IF HO=1 PROC1
2420 IF HO=2 PROC2
2430 IF HO=3 PROC3
2440 IF HO>3 PROC4
2450 ENDPROC
2460 DEFPROCvar
A=10:B=2:HO=1:SO=0
A$="l":LA=1
EG=0:EGS=0
ST=.7:UP=26
EGD=0
TEC=0:TED=0
RO=0:ROG=-1
J=0:RE=0:SCT=0
S1=0:S2=0:S3=0:S4=0:S5=0
PG=0:PF=1:B$="l":PIA=19
S1=0:S2=0:S3=0:S4=0:S5=0
2470 ENDPROC
2480 DEFPROC1
2490 CHI=RND(2)
2500 IF CHI=1 THEN LA=1:ENDPROC
2510 IF CHI=2 THEN LA=5:ENDPROC
2520 ENDPROC
2530 DEFPROC2
2540 CHI=RND(3)
2550 IF CHI=1 THEN LA=1:ENDPROC
2560 IF CHI=2 THEN LA=3:ENDPROC
2570 IF CHI=3 THEN LA=5:ENDPROC
2580 ENDPROC
2590 DEFPROC3
2600 CHI=RND(4)
2610 IF CHI=1 THEN LA=1:ENDPROC
2620 IF CHI=2 THEN LA=2:ENDPROC
2630 IF CHI=3 THEN LA=4:ENDPROC
2640 IF CHI=4 THEN LA=5:ENDPROC
2650 ENDPROC
2660 DEFPROC4
2670 CHI=RND(5)
2680 IF CHI=1 THEN LA=1:ENDPROC
2690 IF CHI=2 THEN LA=2:ENDPROC
2700 IF CHI=3 THEN LA=3:ENDPROC
2710 IF CHI=4 THEN LA=4:ENDPROC
2720 IF CHI=5 THEN LA=5:ENDPROC
2730 ENDPROC
2740 DEFPROC5
2750 IF EG<EGD THEN RE=2:PROCend
2760 VDU 28.0.30.19.4:CLS:VDU 26
2770 TEC=TEC+EGS:TED=TED+EGD
2780 SCT=SCT+(EGS+EGD)+(HO*10)
2790 EGS=0:EGD=0
2800 HO=HO+1
2810 PROCscr
2820 PROCchrg:PROChens
2830 IF HO=2:LA=1:PROCflap:LA=3:PROC
flap:LA=5:PROCflap:EG=0
2840 IF HO=3:LA=1:PROCflap:LA=2:PROC
flap:LA=4:PROCflap:LA=5:PROCflap:EG=0
2850 IF HO>3:LA=1:PROCflap:LA=2:PROC
flap:LA=3:PROCflap:LA=4:PROCflap:LA=5:PR
OCflap:EG=0
2860 FE=150
2870 S1=0:S2=0:S3=0:S4=0:S5=0
2880 A=16:RO=0:ROG=-1:PG=0:B$="l"
2890 PIA=19
2900 IF J=1 THEN J=0:UP=UP+2
3020 VDU 23.151.224.224.224.224.64.1
92.224.224
3030 PROCflg:PROCfalf
3040 TIME=0
3050 IF HO>3 AND HO<8 THEN ST=HO/6
3060 ENDPROC
3070 DEFPROCsmash
3080 PRINT TAB(C+1.28) " :TAB(C+1.27
)" "
3090 MOVE ((C+1)*2)*32.98:GCOL 0.3:D
RAW (((C+1)*2)*32)+64.98:MOVE 10.(((C+1)
*2)*32).102:GCOL 0.2:DRAW (((C+1)*2)*32)
+44.102:DRAW (((C+1)*2)*32)+40.106:DRAW
(((C+1)*2)*32)+20.106
3100 SOUND 0.2.140.3
3110 ENDPROC
3120 DEFPROCend
3130 VDU 28.0.30.19.5:CLS:VDU 26
3140 TEC=TEC+EGS:TED=TED+EGD
3150 SCT=SCT+(EGS+EGD)
3160 COLOUR 3:PRINT TAB(3.7)"HEN NOU
SE 000":TAB(16-LEN (STR$ HO),7)HO:COLOUR
1:PRINT TAB(4.9)"TOTAL EGGS:"COLOUR 3:
PRINT TAB(2.11)"IN BASKET 0000":TAB(17-L
EN (STR$ TEC),11)TEC:TAB(3.13)"DROPPED 0
000":TAB(16-LEN (STR$ TED),13)TED
3170 COLOUR 2:PRINT TAB(4.15)"SCORE
00000":TAB(15-LEN (STR$ SCT),15)SCT
3180 COLOUR 2:PRINT TAB(4.5)"END OF
GAME"
3190 IF RE=1 THEN PROCreso
3200 IF RE=2 THEN PROCreat
3210 IF RE=3 THEN PROCreath
3220 IF RE=4 THEN PROCreaf
3230 IF RE=5 THEN PROCreaf1
3240 COLOUR 3
3250 IF SO=0 THEN FOR WA=200 TO 50 S
TEP -2.5:SOUND 1.5.WA.1:NEXT ELSE FOR WA
=0 TO 4000:NEXT
3260 PROCchhi
3270 VDU 28.0.30.19.3:CLS:VDU 26
3280 GOTO 160
3290 ENDPROC
3300 DEFPROCjump
3310 PRINT TAB(A,UP) " :TAB(A,UP+1)
" :TAB(A,UP+2) "
3320 UP=UP-2
3330 SOUND 1.4.150.6
3340 IF (EG=1) AND (A$="l"):IF (A=C
1) AND (ED>=UP+1) THEN EGS=EGS+1:SOUND 1
.3.120.2:EG=0:PRINT TAB(C+1,UP+1) " :PRO
Cfalf
3350 IF (EG=1) AND (A$="r"):IF (A=C
1) AND (ED>=UP+1) THEN EGS=EGS+1:SOUND 1
.3.120.2:EG=0:PRINT TAB(C+1,UP+1) " :PROCF
art
3360 COLOUR 3
3370 PRINT TAB(9-LEN (STR$(EGS)),5):
EGS
3380 IF A$="l" THEN PROCfalf ELSE PR
OCfart
3390 TJ=TIME
3400 ENDPROC
3410 DEFPROCdown
3420 TJ=0
3430 PRINT TAB(A,UP) " :TAB(A,UP+1)
" :TAB(A,UP+2) "
3440 UP=UP-2
3450 SOUND 1.5.150.5
3460 IF A$="r" THEN PROCfalf ELSE PR
OCfart
3470 ENDPROC
3480 DEFPROCreso
3490 PRINT TAB(2.21)"YOU WERE HIT BY
"
3500 PRINT TAB(1.23)"THE ROTTEN EGG!
!!"
3510 ENDPROC
3520 DEFPROCreat
3530 PRINT TAB(0.21)"YOU FAILED TO C
ATCH"
3540 PRINT TAB(2.23)"AS MANY EGGS AS
"
3550 PRINT TAB(3.25)"YOU DROPPED!!"
3560 ENDPROC
3570 DEFPROCreath
3580 PRINT TAB(1.21)"YOU HAVE BEEN H
IT"
3590 PRINT TAB(1.23)"ON THE HEAD BY
AN"
3600 PRINT TAB(7.25)"EGG!!"
3610 ENDPROC
3620 DEFPROCslpy
3630 IF S1=0 AND C=0 THEN S1=1
3640 IF S2=0 AND C=4 THEN S2=5
3650 IF S3=0 AND C=8 THEN S3=9
3660 IF S4=0 AND C=12 THEN S4=13
3670 IF S5=0 AND C=16 THEN S5=17
3680 ENDPROC
3690 DEFPROCreaf
3700 PRINT TAB(1.21)"YOU SLIPPED ON
AN"
3710 PRINT TAB(4.23)"EGG YOLK!!!"
3720 ENDPROC
3730 DEFPROCslpd
3740 IF A$="l" THEN PROCcl1
3750 IF A$="r" THEN PROCclr
3760 ENDPROC
3770 DEFPROCcl1
3780 IF A=0 THEN PROCclr:ENDPROC
3790 IF A=S1 THEN RE=4:PROCend

```

```

3800 IF A=S2 THEN RE=4:PROCend
3810 IF A=S3 THEN RE=4:PROCend
3820 IF A=S4 THEN RE=4:PROCend
3830 IF A=S5 THEN RE=4:PROCend
3840 ENDPROC
3850 DEFPROCc1r
3860 IF A=1:S1 THEN RE=4:PROCend
3870 IF A=1:S2 THEN RE=4:PROCend
3880 IF A=1:S3 THEN RE=4:PROCend
3890 IF A=1:S4 THEN RE=4:PROCend
3900 IF A=1:S5 THEN RE=4:PROCend
3910 ENDPROC
3920 DEFPROCc1g
3930 VDU 23.159.0.0.0.0.2.3.1.15
3940 VDU 23.154.0.0.0.3.15.159.255.2
55
3950 VDU 23.155.0.8.16.8.208.224.240
.240
3960 VDU 23.156.57.249.255.255.30.0.
1.1
3970 VDU 23.157.255.255.255.255.207.
192.128.128
3980 VDU 23.158.240.240.240.224.224.
192.192.192
3990 VDU 23.160.3.1.0.0.0.0.0
4000 ENDPROC
4010 DEFPROCc1f
4020 IF PF=1 THEN PF=2:VDU 23.156.57
.249.255.255.30.0.1.1:VDU 23.157.255.255
.255.255.207.192.128.128:VDU 23.158.240.
240.240.224.224.192.192:GOTO 4040
4030 IF PF=2 THEN PF=1:VDU 23.156.57
.249.255.231.30.0.0.0:VDU 23.157.255.255
.255.255.207.192.97.97:VDU 23.158.240.24
0.240.224.192.192.128.128
4040 IF PIA>17 THEN PRINT TAB(18,27)
" :TAB(17,28) "
4050 IF PIA<19 THEN PRINT TAB(PIA,27)
" :TAB(PIA,28) "
4060 COLOUR 1
4070 IF PIA<PD THEN B$="r":PG=2:PRO
Cpr:ENDPROC
4080 IF A$="1" THEN PROCfalf
4090 IF A$="r" THEN PROCfart
4100 IF PIA=20 THEN PIA=PIA-1:PRINT
TAB(PIA,27)CHR$159:TAB(PIA,28)CHR$156
4110 IF PIA=19 THEN PIA=PIA-1:PRINT
TAB(PIA,27)CHR$159:CHR$154:TAB(PIA,28)CH
R$156:CHR$157
4120 IF PIA=18 THEN PIA=PIA-1:PRINT
TAB(PIA,27)CHR$159:CHR$154:CHR$155:TAB(P
IA,28)CHR$156:CHR$157:CHR$158:GOTO 4180
4130 IF PIA<18 THEN PIA=PIA-1:PRINT
TAB(PIA,27)CHR$159:CHR$154:CHR$155:TAB(P
IA,28)CHR$156:CHR$157:CHR$158
4140 VDU 5:MOVE ((PIA*2)*32)-4,(3*32
)-28:GCOL 0,3:PRINT CHR$160:VDU 4
4150 IF PIA=S1 THEN S1=0
4160 IF PIA=S2 THEN S2=0
4170 IF PIA=S3 THEN S3=0
4180 IF PIA=S4 THEN S4=0
4190 IF PIA=17 AND S5<0 THEN S5=0
4200 ENDPROC
4210 DEFPROCcprg
4220 VDU 23.159.0.16.8.16.11.7.15.15
4230 VDU 23.154.0.0.0.192.240.249.25
5.255
4240 VDU 23.155.0.0.0.0.64.192.128.2
40
4250 VDU 23.156.15.15.15.7.7.3.3.3
4260 VDU 23.157.255.255.255.255.243.
3.1.1
4270 VDU 23.158.156.159.255.255.120.
0.128.128
4280 VDU 23.160.3.2.0.0.0.0.0
4290 ENDPROC
4300 DEFPROCcprt
4310 IF PIA<18 THEN PRINT TAB(PIA,27)
" :TAB(PIA,28) "
4320 IF PIA=18 THEN PRINT TAB(PIA,27)
" :TAB(PIA,28) "
4330 IF PIA=19 THEN PRINT TAB(PIA,27)
" :TAB(PIA,28) "
4340 IF PIA<18 THEN VDU 5:MOVE (((PI
A-2)*2)*32)-38,(3*32)+32:GCOL 0,0:PRINT
CHR$160:VDU 4
4350 IF A$="1" THEN PROCfalf
4360 IF A$="r" THEN PROCfart
4370 IF PIA>18 THEN B$="1":PG=0:ENDP
ROC
4380 IF PF=1 THEN PF=2:VDU 23.156.15
.15.15.7.7.3.3:VDU 23.157.255.255.255.
255.243.3.1.1:VDU 23.158.156.159.255.255
.120.0.128.128:GOTO 4400
4390 IF PF=2 THEN PF=1:VDU 23.156.15
.15.15.7.3.3.1.1:VDU 23.157.255.255.255.
255.243.3.134.134:VDU 23.158.156.159.255
.231.120.0.0.0
4400 COLOUR 1
4410 IF PIA=18 THEN PIA=PIA-1:PRINT
TAB(PIA,27)CHR$159:TAB(PIA,28)CHR$156
4420 IF PIA=17 THEN PIA=PIA-1:PRINT
TAB(PIA,27)CHR$159:CHR$154:TAB(PIA,28)CH
R$156:CHR$157
4430 IF PIA=16 THEN PIA=PIA-1:PRINT
TAB(PIA,27)CHR$159:CHR$154:CHR$155:TAB(P
IA,28)CHR$156:CHR$157:CHR$158
4440 IF PIA<16 THEN PIA=PIA-1:PRINT
TAB(PIA,27)CHR$159:CHR$154:CHR$155:TAB(P
IA,28)CHR$156:CHR$157:CHR$158
4450 IF PIA<18 THEN VDU 5:MOVE (((PI
A-2)*2)*32)-38,(3*32)+28:GCOL 0,3:PRINT
CHR$160:VDU 4
4460 IF PIA=S1 THEN S1=0
4470 IF PIA=S2 THEN S2=0
4480 IF PIA=S3 THEN S3=0
4490 IF PIA=S4 THEN S4=0
4500 ENDPROC
4510 DEFPROCcchp1
4520 IF (S1=0) AND (S2=0) AND (S3=0)
AND (S4=0) AND (S5=0) THEN PG=0:ENDPROC
4530 IF S1>0 THEN PD=S1:GOTO 4580
4540 IF S2>0 THEN PD=S2:GOTO 4580
4550 IF S3>0 THEN PD=S3:GOTO 4580
4560 IF S4>0 THEN PD=S4:GOTO 4580
4570 IF S5>0 THEN PD=S5
4580 PG=1
4590 PROCp1g
4600 PIA=19:PF=1:B$="1"
4610 ENDPROC
4620 DEFPROCcrafs1
4630 PRINT TAB(3,21)"YOU HAVE BEEN"
4640 PRINT TAB(0,23)"BITTEN BY THE P
IG!"
4650 ENDPROC
4660 DEFPROCcprh1
4670 COLOUR 1
4680 PRINT TAB(3,4)"TOP FARMERS!!"
4690 COLOUR 3
4700 FOR H=1 TO 5:GOTO 1.RND(3)+2.1
40.3:PRINT TAB(0,3)+(H*4):HI(H):TAB(8,3+
(H*4)):HI(H):NEXT
4710 COLOUR 1:VDU 19.2.510:PRINT TA
B(1,27)"ESCAPE FOR INFO OR":COLOUR 2:PRI
NT TAB(0,29)"PRESS SPACE TO START"
4720 REPEAT:UNTIL INKEY(-99)
4730 VDU 28.0.30.19.4:CLS:VDU 26
4740 VDU 19.2.310:
4750 ENDPROC
4760 DEFPROCcchh1
4770 VDU 28.0.30.19.19:CLS:VDU 26
4780 H=1
4790 IF SCT>HI(H) THEN GOTO 4810 ELS
E H=H+1:IF H=6 THEN ENDPROC
4800 GOTO 4790
4810 IF H=1 THEN PROCH11
4820 IF H=2 THEN PROCH12
4830 IF H=3 THEN PROCH13
4840 IF H=4 THEN PROCH14
4850 IF H=5 THEN PROCH15
4860 GOTO 160
4870 DEFPROCch11
4880 HI(5)=HI(4)
4890 HI(4)=HI(3)
4900 HI(3)=HI(2)
4910 HI(2)=HI(1)
4920 HI(5)=HI(4)
4930 HI(4)=HI(3)
4940 HI(3)=HI(2)
4950 HI(2)=HI(1)
4960 HI(1)=SCT
4970 COLOUR 1
4980 PRINT TAB(3,21)"YOUR SCORE WAS"
4990 PRINT TAB(1,23)"RANKED NUMBER
ONE"
5000 HI(1)=""
5010 PROCTpin
5020 HI(1)=HI(4)
5030 ENDPROC
5040 DEFPROCch12
5050 HI(5)=HI(4)
5060 HI(4)=HI(3)
5070 HI(3)=HI(2)
5080 HI(5)=HI(4)
5090 HI(4)=HI(3)
5100 HI(3)=HI(2)
5110 HI(2)=SCT
5120 COLOUR 1
5130 PRINT TAB(3,21)"YOUR SCORE WAS"
5140 PRINT TAB(1,23)"RANKED NUMBER
TWO"
5150 HI(2)=""
5160 PROCTpin
5170 HI(2)=HI(4)
5180 ENDPROC
5190 DEFPROCch13
5200 HI(5)=HI(4)
5210 HI(4)=HI(3)
5220 HI(5)=HI(4)
5230 HI(4)=HI(3)
5240 HI(3)=SCT
5250 COLOUR 1
5260 PRINT TAB(3,21)"YOUR SCORE WAS"
5270 PRINT TAB(0,23)"RANKED NUMBER
THREE"
5280 HI(3)=""
5290 PROCTpin
5300 HI(3)=HI(4)
5310 ENDPROC
5320 DEFPROCch14
5330 HI(5)=HI(4)
5340 HI(5)=HI(4)
5350 HI(4)=SCT
5360 COLOUR 1
5370 PRINT TAB(3,21)"YOUR SCORE WAS"
5380 PRINT TAB(1,23)"RANKED NUMBER F
OUR"
5390 HI(4)=""
5400 PROCTpin
5410 HI(4)=HI(4)
5420 ENDPROC
5430 DEFPROCch15
5440 HI(5)=SCT
5450 COLOUR 1
5460 PRINT TAB(3,21)"YOUR SCORE WAS"
5470 PRINT TAB(1,23)"RANKED NUMBER F
IVE"
5480 HI(5)=""
5490 PROCTpin
5500 HI(5)=HI(4)
5510 ENDPROC
5520 ENDPROC
5530 DEFPROCc1p1n
5540 HI$=""
5550 PRINT TAB(1,25)"TYPE IN YOUR NA
ME!"
5560 COLOUR 3
5570 PRINT TAB(4,29)"....."
5580 CHR=1
5590 *FX 15
5600 G=GET
5610 *FX 15
5620 SOUND 1.3.RND(255).1
5630 COLOUR RND(3)
5640 IF CHR>1 AND G=127 THEN CHR=CHR
-1:PRINT TAB(3-CHR,29)".":HI$=MID$(HI$,
0,(LEN HI$)-1):GOTO 5600
5650 IF G=13 THEN VDU 28.0.30.19.3:C
LS:VDU 26:ENDPROC
5660 HI$=HI$+CHR$ G
5670 PRINT TAB(3-CHR,29)CHR$G
5680 IF CHR<12 THEN PRINT TAB(4-CHR,
29)". "
5690 CHR=CHR+1
5700 IF CHR=13 THEN REPEAT:G=GET:UNT
IL (G=13) OR (G=127)
5710 SOUND 1.3.RND(255).4
5720 IF G=13 THEN VDU 28.0.30.19.3:C
LS:VDU 26:ENDPROC
5730 IF G=127 THEN CHR=CHR-1:PRINT T
AB(3-CHR,29)".":HI$=MID$(HI$,0,(LEN HI$)
-1):GOTO 5600
5740 GOTO 5600
5750 DEFPROCc1nstr
5760 PRINT TAB(11,1)"EGGS by Paul Dy
er for the 32k ELECTRON
OR BBC"
5770 PRINT "" CATCH A FALLING EGG A
ND PUT IT IN YOUR BASKET. That's the whole
idea of this game. Above the ground will
be a number of nests, depending on the
screen you are on, and you must catch fal
ling eggs in your basket as they drop."
5780 PRINT " To warn you if an egg
is about to be layed the chicken will
cluck. There are two types of cluck and o
nly one signals laying. Now to the hazard
s. A rotten egg floats past and this mus
t be avoided at!"
5790 PRINT "all costs, falling eggs d
o no good to you except in a basket. Missed
eggs will smash on the floor leaving a s
lippery mess to be avoided. A large pig w
ill appear when!"
5800 PRINT "an egg is dropped to rem
ove the mess, but he is not all good so st
eer clear. Lastly if you do not collect a
s many (if not more) eggs than you dro
p the game will end, you only have one f
armer, so use him!"
5810 AX=INKEY(5000)
5820 PRINT "well. At the top of the s
creen is a multicoloured bar which repre
sents the amount of feed the hens have
left. Four notes will warn that it is low
. When it has all been used the screen wi
ll clear to show current score and level"
5830 AX=INKEY(200)
5840 PRINT "after which the next sc
reen will appear and be even harder
than the previous screen."
5850 AX=INKEY(200)
5860 PRINT " 1 point is scored for
a caught egg and 1 point lost for an egg
dropped. A bonus is scored for each compl
eted screen. Move your farmer to the left
with the Z key, right with the X key and
d jump with the RETURN key."
5870 AX=INKEY(200)
5880 PRINT " Last of all the sound
can be switched off with the Q key and o
n with the S key. And if you need to pau
se the game press the P key and restart wi
th the O key."
5890 AX=INKEY(200)
5900 PRINT "".....Press ANY key
to move on....."
5910 IF GET ENDPROC
5920 REM ERROR PROCEDURE
5930 IF ERR<>17 THEN GOTO 5970
5940 MODE 6:VDU 23.1.0.0.0.0:PROCin
str
5950 MODE 5:VDU 23.1.0.0.0.0:PROCTt
1
5960 RETURN
5970 MODE 6
5980 PRINT "REPORT:PRINT " at line
":ERL
5990 VDU 23.1.1.0.0.0:
6000 END

```


RECONNAISSANCE

• QL

Reconnaissance is a contrast to the usual shoot-em-up, high speed arcade games. It is a game which requires ingenuity, care and intelligence to successfully complete and can also be extremely addictive.

The basic aim is simple. You are a member of an elite corps whose assignment is to establish the precise location of four high explosive ordnance stores within an enemy camp prior to a bombing raid. You cannot gain admittance to the camp and so can only prowl about the perimeter.

You have divided the enemy camp into an 8×8 grid and it is this grid which forms the basis of your strategy. Each side is denoted by a letter, A – D and each grid square by numbers across the bottom and the top.

You cannot see into the camp as your survey will take place at night. The one item of equipment you have with you is a maser rifle which projects a coherent beam of microwave radiation. Around the perimeter of the camp, which is thoughtfully square, you have placed, at equal intervals, small reflective discs which are wired to your maser computer, these discs being identified as numbers on the sides (fig. 1).

The method of determining the precise location of each ordnance dump is as follows: positioning your rifle at the first co-ordinate you fire a shot. The entry position is shown in the window on the left of the screen e.g. A1. This is followed immediately by the exit co-ordinate which is itself determined from the route taken by the maser beam after being fired.

Fig. 2 shows this more clearly. Here it can be seen that an "invisible" ordnance store is centred on 3,4. Because of its size, each "store" overlaps into the surrounding eight squares and firing a beam into any of the eight squares will produce a different exit co-ordinate. The way it works is this:

A beam entering at A2 (top left) hits the store at 2,3, is reflected at right angles to the left and emerges at D3. The display in the left hand window would read 'A2 : D3'.

A shot entering at D5 is reflected to the

bottom and would emerge at C2.

Entry at C4 would mean exit at B5 and so on.

However, a direct hit, i.e. entry from A3, B4, C3 or D4 would lead to the message "absorbed" being displayed.

From this information it can be deduced that a store exists centred on 3,4. Hah! Obvious, you might think. But what if a second store exists with its centre on the next but one square as shown in fig. 3? The 'store' on 5,4 is contiguous with the one on 3,4 therefore reflecting the beam straight back out at A4.

Even more sneaky, if another "store" were positioned at 7,7 then a beam entering at B5 would bounce at right angles downwards where, at 6,6 it would encounter the top left of the third 'store' and be reflected yet again through ninety degrees bringing the beam back to its original path but displaced by one row downward.

For that manoeuvre the display would read: B5 (entry) : D6 (exit).

If a beam were aimed from C6 then, as it encountered an obstruction so close to the point of origin, it would re-emerge at the same point...

Since the game uses a total of four "stores", the possible combinations are truly mind boggling. Surprisingly, it's not that difficult when you've become used to it!

For each "shot" your score increases by one and the aim is to complete the game with the fewest possible tries. When you think you've located every single store you press <ESC> and a small prompt appears in the window below the grid asking for, first, the x co-ordinate and then the y co-ordinate of each of the four "stores". If you have it right then congratulations are offered but a wrong guess simply tells you you're wrong but does not specify how! Thus you are left to try and eliminate the possibilities.

The game does not allow you to try the same (or a reverse) co-ordinate twice; therefore a pencil is a very useful adjunct when playing this.

The screen display is organised into four windows, the largest of which holds the Grid plan and looks like figure 4. You are

repr
whic
draw
serie
arro
the f
Lin
dire
(108
func

Fig. 1

1
2
3
4
5
6
7
8

Fig. 3

1
2
3
4
5
6
7
8

represented by a turtle graphics arrow which has a handy advantage over a line drawn figure. The turtle draws the arrow in a series of four movements (fig. 5) and the arrow will appear pointing the way in which the turtle is aimed.

Lines 1170 to 1290 draw the arrow and the direction is set by the procedure "Gun" (1080 to 1120) which in turn calls three functions, Gun_x, Gun_y and Gun_r which

return the x and y position and the initial rotation.

The program uses windows, circles, blocks, filled shapes and turtles. In addition, the not inconsiderable calculating and data manipulation power of the QL is used extensively.

Happy spying!

• Ian Kennedy, Ian Williams and Steven Hollywood

Fig. 1

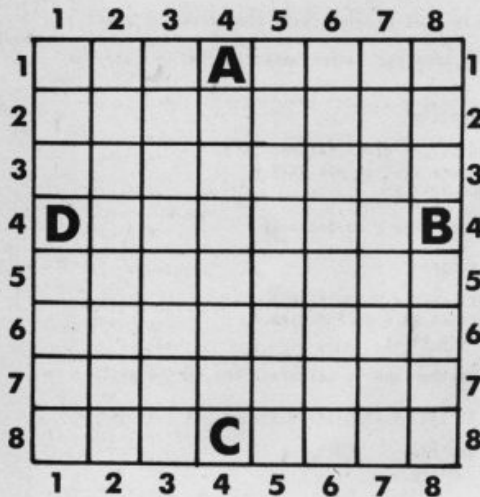


Fig. 2

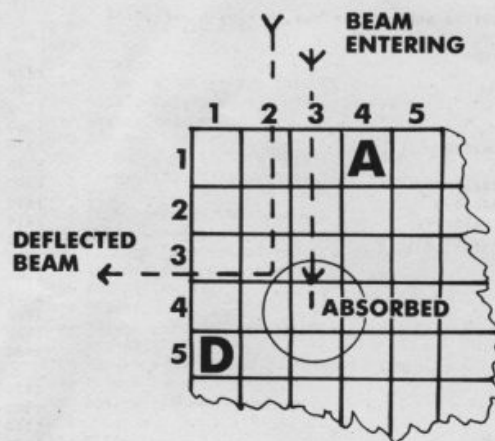


Fig. 3

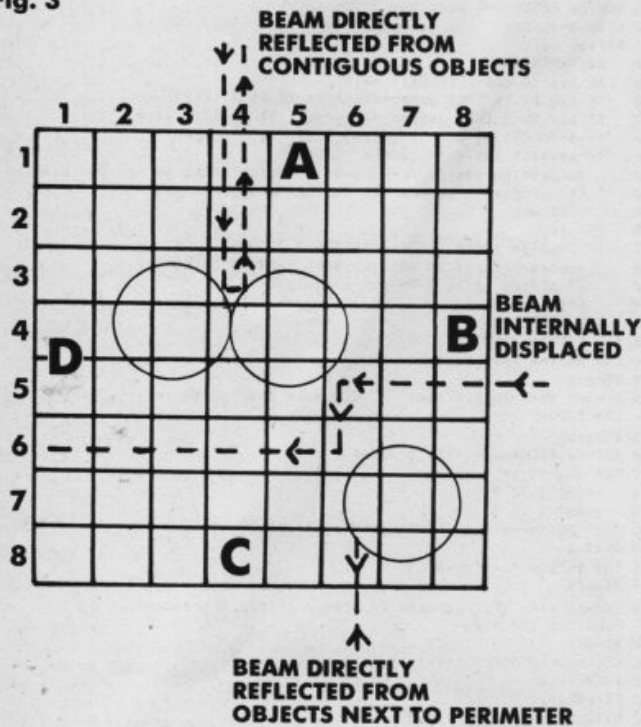
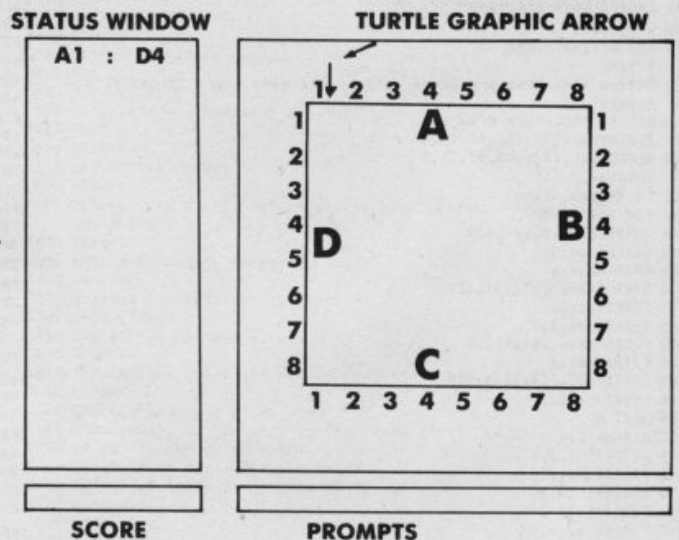


Fig. 4



RECONNAISSANCE

```

100 REMark
110 REMark
120 REMark
130 REMark
140 REMark
150 REMark
160 REMark
170 REMark
180 REMark
190 REMark
200 REMark
210 REMark
220 CLEAR
230 diabo
240 gun gunpos,7
250 REPEAT mainloop
260   move_gun
270   IF key && 64 THEN fire
280   IF key && 8 THEN ask
290   IF key && 32 THEN reveal:close_down
300   gun oldpos,0
310   gun gunpos,7
320 END REPEAT mainloop
330 STOP
340 REMark
350 REMark **** This routine sets up windows, arrays & variables
360 REMark
370 DEFINE PROCEDURE diabo
380 box=3
390 tab=4
400 mark=5
410 ques=6
420 OPEN#box,scr_512x256a0x0
430 CLS#box
440 OPEN#box,scr_280x240a200x0
450 OPEN#tab,scr_158x240a40x0
460 OPEN#mark,scr_158x12a40x241
470 OPEN#ques,con_280x12a200x241_10
480 BORDER#box,1,7,0
490 BORDER#tab,1,4,2
500 BORDER#mark,1,7,4
510 BORDER#ques,1,7
520 CLS#box
530 CLS#tab
540 CLS#mark
550 CLS#ques
560 CSIZE#tab,1,0
570 CSIZE#mark,2,0
580 INK#mark,7
590 INK#ques,4
600 PRINT#mark,"Score";
610 marks=0
620 print_score
630 size=7
640 num_things=4
650 draw_box
660 DIM grid(size+2,size+2),gone((size+1)*4-1),guess(num_things-1,1)
670 fill_grid(num_things)
680 xpos=0:ypos=0:gunpos=0
690 mov_num=1
700 END DEFINE diabo
710 REMark
720 REMark **** This procedure sets up the game board (#box) window
730 REMark
740 DEFINE PROCEDURE draw_box
750 BLOCK#box,200,160,40,30,4
760 BLOCK#box,196,158,42,31,2
770 INK#box,7
780 CSIZE#box,0,1
790 FOR a=1 TO 8
800 CURSOR#box,5,10+a*20
810 CSIZE#box,0,1
820 PRINT#box,a
830 CURSOR#box,265,10+a*20
840 PRINT#box,a
850 CSIZE#box,3,0
860 CURSOR#box,20+a*25,0
870 PRINT#box,a
880 CURSOR#box,20+a*25,218
890 PRINT#box,a
900 NEXT a
910 AT#box,0,0
920 CSIZE#box,2,1
930 PAPER#box,2
940 RESTORE 1010
950 FOR a=0 TO 3
960   READ a$,x,y
970   CURSOR#box,x,y
980   PRINT#box,a$
990 NEXT a
1000 END DEFINE draw_box
1010 DATA "A",132,35
1020 DATA "B",215,100
1030 DATA "C",132,165
1040 DATA "D",50,100
1050 REMark
1060 REMark **** This proc is used to set the turtle direction
    'gun?'

1070 REMark
1080 DEFINE PROCEDURE gun(pos,c)
1090 POINT#box,gun_x(pos),gun_y(pos)
1100 TURNTO#box,gun_r(pos)
1110 arrow(c)
1120 END DEFINE gun
1130 REMark
1140 REMark **** This one is uses the turtle graphics to draw an arrow
1150 REMark **** of colour (col)
1160 REMark
1170 DEFINE PROCEDURE arrow(col)
1180 INK#box,col
1190 PENDOWN#box
1200 MOVE#box,5
1210 TURN#box,135
1220 MOVE#box,3
1230 TURN#box,135
1240 MOVE#box,SQRT(18)
1250 TURN#box,135
1260 MOVE#box,3
1270 TURN#box,-45
1280 MOVE#box,-5
1290 END DEFINE arrow
1300 REMark
1310 REMark **** This is used to unscramble the arrows (x_coord.)
    position
1320 REMark **** mainly from the gunpos variable (returns graphic
    coords)
1330 REMark
1340 DEFINE FUNCTION gun_x(u)
1350 side=u DIV 8
1360 IF side && 1 THEN RETURN 81-37.5*(side && 2)
1370 IF side=2 THEN RETURN 71.1-(u MOD 8)*7.8
1380 RETURN 16.5+(u MOD 8)*7.8
1390 REMark
1400 REMark **** Ditto for the y co-ordinate
1410 REMark
1420 DEFINE FUNCTION gun_y(u)
1430 side=u DIV 8
1440 IF side=3 THEN RETURN 25+(u MOD 8)*8.4
1450 IF side=1 THEN RETURN 83.8-(u MOD 8)*8.4
1460 RETURN 93.5-40*(side && 2)
1470 REMark
1480 REMark **** Yet another one to calculate the turtle setting to
    'enable'
1490 REMark **** the turtle to be rotated to the right direction
1500 REMark
1510 DEFINE FUNCTION gun_r(u)
1520 RETURN 270-(u DIV 8)*90
1530 REMark
1540 REMark **** This routine does most of the moving of the arrow using
    the
1550 REMark **** KEYROW and INKEY functions
1560 REMark
1570 DEFINE PROCEDURE move_gun
1580 oldpos=gunpos
1590 REPEAT wait
1600   key=KEYROW(1)
1610   IF key && 104 THEN EXIT wait
1620   IF key && 16 THEN gunpos=(gunpos+1) && 31:EXIT wait
1630   IF key && 2 THEN gunpos=(gunpos-1) && 31:EXIT wait
1640   key=INKEY$(0) & " "
1650   IF key$(1) INSTR "AaBbCcDd" THEN
1660     gunpos=(gunpos && 7)+((key$(1) INSTR "AaBbCcDd"~1) DIV 2)*8
1670     IF (gunpos ^^ oldpos) && 16 THEN gunpos=gunpos ^^ 7
1680     EXIT wait
1690   END IF
1700   IF key$(1) INSTR "12345678" THEN
1710     gunpos=(gunpos && 24)+(key$(1) INSTR "12345678"~1)-1
1720     IF gunpos && 16 THEN gunpos=gunpos ^^ 7
1730     EXIT wait
1740   END IF
1750 END REPEAT wait
1760 END DEFINE move_gun
1770 REMark
1780 REMark **** Initialises 'n' BLOBS in the 'grid' array. The more
    the harder
1790 REMark
1800 DEFINE PROCEDURE fill_grid(n)
1810 FOR a=1 TO n
1820   x=RND(1 TO 8)
1830   y=RND(1 TO 8)
1840   IF NOT(grid(x,y)) THEN grid(x,y)=1:ELSE a=a-1
1850 NEXT a
1860 END DEFINE fill_grid
1870 REMark
1880 REMark **** Plot is used to draw a circle (representing a
    BLOB) in the box
1890 REMark
1900 DEFINE PROCEDURE plot(x,y,c)
1910 INK#box,c
1920 FILL#box,1
1930 CIRCLE#box,gun_x(x),gun_y(y+8)-1,6
1940 FILL#box,0
1950 END DEFINE plot
1960 REMark
1970 REMark **** The hub of the whole game, it 'moves' the ball within the
1980 REMark **** box, calculates and displays enter and exit points

```

```

1990 RE
2000 DI
2010 IF
2020
2030
2040
2050
2060
2070 EN
2080 go
2090 s
2100 PF
2110 ac
2120 IF
2130
2140
2150 EN
2160 IF
2170
2180
2190 EN
2200 IF
2210 IF
2220 RE
2230
2240
"0
2250
EX
2260
EX
2270
2280
2290
2300
2310
2320
2330
2340
2350
2360
2370
2380
2390
2400
2410
2420
2430
2440
2450
2460
2470 EN
2480 sc
2490 IF
2500
2510
2520
2530 EN
2540 IF
2550
2560
2570
2580 EN
2590 EN
2600 RE
2610 RE
2620 RE
2630 RE
2640 DE
2650 FC
2660
2670
2680
2690
2700 NE
2710 EN
2720 RE
2730 RE
2740 RE
2750 DE
2760 AT
2770 a$
2780 PF
2790 EN
2800 RE
2810 RE
2820 RE
2830 RE
2840 RE
2850 RE
2860 DE
2870 wr

```



```

1990 REMark
2000 DEFINE PROCEDURE fire
2010 IF gone(gunpos) THEN
2020   IF gone(gunpos)=1 THEN
2030     message "ALREADY TRIED !"
2040   ELSE message "REVERSE ALREADY TRIED !"
2050   END IF
2060   RETURN
2070 END IF
2080 gone(gunpos)=1
2090 side=gunpos DIV 8
2100 PRINT#tab, " (1 TO (mov_num<10));mov_num;" ; TO 4;"ABCD"
2110 side=1;"12345678"((gunpos && 16)/(16*7))+1); " : ";
2120 mov_num=mov_num+1
2130 IF side && 1 THEN
2140   xpos=(size+2)*(side ^ 3)/2
2150 ELSE xpos=((gunpos && 16)/16*7) ^ (gunpos && 7))+1
2160 END IF
2170 IF NOT(side && 1) THEN
2180   ypos=(size+2)*(side DIV 2)
2190 ELSE ypos=((gunpos && 16)/16*7) ^ (gunpos && 7))+1
2200 END IF
2210 IF NOT(side && 1) THEN xdir=0;ydir=1-side
2220 IF side && 1 THEN ydir=0;xdir=side-2
2230 REPEAT moveloop
2240   GO TO 2270
2250   IF ((ypos+ydir) INSTR "09") AND ((xpos+xdir) INSTR
2260     "09") THEN EXIT moveloop
2270   IF ((ypos+ydir) INSTR "09") THEN ypos=ypos+ydir;
2280   EXIT moveloop
2290   IF ((xpos+xdir) INSTR "09") THEN xpos=xpos+xdir;
2300   EXIT moveloop
2310   IF grid(xpos+xdir,ypos+ydir) THEN
2320     PRINT#tab,"absorbed"
2330     xpos=xpos+xdir
2340     ypos=ypos+ydir
2350     score 1
2360     EXIT moveloop
2370   END IF
2380   xpd=ydir*NOT(xdir)
2390   ypd=xdir*NOT(ydir)
2400   IF grid(xpos+xdir*xpd,ypos+ydir*ypd) THEN
2410     xdir=xpd;ydir=ypd
2420     GO TO 2240
2430   END IF
2440   xpos=xpos+xdir
2450   ypos=ypos+ydir
2460   IF xpos=0 OR xpos=9 OR ypos=0 OR ypos=9 THEN EXIT moveloop
2470 END REPEAT moveloop
2480 score 1
2490 IF xpos=0 OR xpos=9 THEN
2500   PRINT#tab,"BD" (1+NOT(xpos));"12345678"(ypos)
2510   e=((ypos-1) ^ (7*NOT(xpos)))+8+16*NOT(xpos)
2520   IF NOT(gone(e)) THEN gone(e)=2
2530 END IF
2540 IF ypos=0 OR ypos=9 THEN
2550   PRINT#tab,"CA" (1+NOT(ypos));"12345678"(xpos)
2560   e=((xpos-1) ^ (7*NOT(ypos)))+8+16*NOT(ypos)
2570   IF NOT(gone(e)) THEN gone(e)=2
2580 END IF
2590 END DEFINE fire
2600 REMark
2610 REMark **** This one is used to increase the score. It is
2620   increased by
2630   one, displayed and followed by a bleep (i) times.
2640   REMark
2650   DEFINE PROCEDURE score(i)
2660   FOR a=1 TO i
2670     marks=marks+1
2680     print_score
2690     BEEP 1000,50
2700   IF BEEPING THEN GO TO 2690
2710 NEXT a
2720 END DEFINE score
2730 REMark
2740 REMark **** Prints present score in the window(mark) to
2750   & figures.
2760   REMark
2770   DEFINE PROCEDURE print_score
2780   AT#mark,0,6
2790   a$="00000" & marks
2800   PRINT#mark,a$(LEN(a$)-5 TO);
2810 END DEFINE print_score
2820 REMark
2830 REMark **** The ask procedure is called when you think
2840   you've got it right.
2850   REMark
2860   REMark **** It asks for your input in console#ques,
2870   checks that the entered
2880   co-ordinates are valid and not repeated and
2890   increases the score
2900   REMark
2910   REMark **** or ends the game depending upon your guess.
2920   REMark
2930   DEFINE PROCEDURE ask
2940   wrong=0
2950   x=0;y=1
2960   guess(0,x)=0;guess(0,y)=0
2970   FOR a=1 TO num_things
2980     INPUT#ques,("Input x co-ordinate of no." & a) ! x$
2990     IF NOT(x$ INSTR "12345678") OR x$="" OR LEN(x$)<1 THEN
3000       message "Invalid Co-ordinate"
3010       GO TO 2910
3020     END IF
3030     IF x$="" THEN EXIT a
3040     INPUT#ques,("Input y co-ordinate of no." & a) ! y$
3050     IF NOT(y$ INSTR "12345678") OR y$="" OR LEN(y$)<1 THEN
3060       message "Invalid Co-ordinate"
3070       GO TO 2910
3080     END IF
3090     IF y$="" THEN EXIT a
3100     FOR b=0 TO a-2
3110       IF guess(b,x)=x$ AND guess(b,y)=y$ THEN
3120         message "You've already entered that one."
3130         a=a-1
3140         EXIT b
3150       END IF
3160     END FOR b
3170     IF guess(b,x)=x$ AND guess(b,y)=y$ THEN GO TO 3140
3180     IF NOT(grid(x$,y$)) THEN wrong=wrong+1
3190     guess(a-1,x)=x$
3200     guess(a-1,y)=y$
3210   END FOR a
3220   CLS#ques
3230   IF x$="" OR y$="" THEN RETURN
3240   PRINT#tab,mov_num;" : TO 3;"Guess";
3250   mov_num=mov_num+1
3260   FOR a=0 TO DIMN(guess)
3270     IF NOT(a && 1) THEN PRINT#tab,TO 8;
3280     PRINT#tab,"(;"guess(a,x);";";guess(a,y);");";
3290   NEXT a
3300   IF NOT(wrong) THEN correct
3310   score wrong*5
3320   message "You've guessed " & (num_things-wrong) & "
3330   correct co-ordinate(s)"
3340 END DEFINE ask
3350 REMark
3360 REMark **** Message sends a string parameter to the 'ques'
3370   channel. It also
3380   Bleeps and pauses returning with the window cleared.
3390   REMark
3400   DEFINE PROCEDURE message(e$)
3410   LOCAL a
3420   CLS#ques
3430   AT#ques,0,22-LEN(e$)/2
3440   INK#ques,7
3450   PRINT#ques,e$
3460   INK#ques,4
3470   BEEP 5000,0
3480   FOR a=0 TO 270:e$=INKEY$(0);NEXT a
3490   CLS#ques
3500 END DEFINE message
3510 REMark
3520 REMark **** Uses plot to put all the BLOBS on the screen as
3530   filled in circles
3540   REMark
3550   DEFINE PROCEDURE reveal
3560   FOR x=1 TO 8
3570     FOR y=1 TO 8
3580       IF grid(x,y) THEN plot x-1,y-1,7,1
3590     NEXT y
3600   NEXT x
3610 END DEFINE reveal
3620 REMark
3630 REMark This one is called if you have managed to find all
3640   the BLOBS
3650   REMark
3660   DEFINE PROCEDURE correct
3670   reveal
3680   message "At last you've got it"
3690   close_down
3700 END DEFINE correct
3710 REMark
3720 REMark This gives the option whether to rerun or end the program
3730   REMark
3740   DEFINE PROCEDURE close_down
3750   INPUT#ques,"Do you want another game (Y/N)"!a$
3760   IF a$="" THEN GO TO 3640
3770   IF a$ INSTR "Yy" THEN CLEAR:RUN
3780   IF NOT(a$ INSTR "Nn") THEN GO TO 3640
3790   CLOSE#box
3800   CLOSE#tab
3810   CLOSE#mark
3820   CLOSE#ques
3830   CLEAR
3840   MODE 0
3850   STOP
3860 END DEFINE close_down

```

OPERATION KRISTOS

• BBCB

A strange alien ship has been picked up on radar heading towards the Earth.

After three days it goes into orbit and fails to respond to radio messages. A preliminary scan reveals a large carbon dioxide laser aimed at London.

The Royal Navy attacks but all ships are lost when a strange wave of pink light radiates out from the ship.

Now the plan has been decided. A Space Marine must dock with the ship, plant a large mine and then teleport back before it explodes. Only then can the Earth be saved from the alien menace that hangs overhead.

You have been chosen for the mission and dock with the ship armed with only a teleport bracelet and a limpit mine that must be attached to the laser to ensure maximum destruction.

The following commands may be used but there are more: GET, DROP, INVENTORY, SCORE, EXAMINE, TELEPORT, PRIME (for the mine).

To move, use the following abbreviations: N—to go North, S—to go South, E—to go East, W—to go West, U—to go Up, D—to go Down.

Any other word may be abbreviated to four letters.

The symbol, which appears in a number of lines, is the up-arrow. This is to the right of !=' on the keyboard.

• JW Roynning

10 DATA "YOU'RE IN YOUR SPACESHIP BUT IT HAS BEEN WRECKED PROBABLY WHEN LANDIN G ON AUTO PILOT. AN EXIT LEADS NORTH"

20 DATA "YOU ARE STANDING IN A SHINY. METALLIC VACUUM TUBE. YOU CAN GO NORTH AND SOUTH"

30 DATA "YOU'RE STANDING IN A AIRLOCK . TO THE WEST IS A SHINY DOOR. TO THE SOUTH IS A CORRIDOR. THE DOOR HAS NO VIS IBLE MEANS OF BEING OPENED."

40 DATA "YOU ARE IN A STEEL CORRIDOR. THERE IS A SIDE EXIT TO THE EAST AND TH E CORRIDOR LEADS NORTH AND SOUTH."

50 DATA "YOU ARE IN A DARK STOREROOM. IT IS EXTREMELY COLD AND SMELLY IN HERE. THE ONLY EXIT IS TO THE NORTH."

60 DATA "YOU ARE STANDING ON A METAL GRILL ABOVE A CHANNEL IN THE FLOOR ABOUT 6 INCHES BELOW. THERE IS AN EXIT NORTH WARDS OVER THE GRILL AND A CORRIDOR LEA DS SOUTH."

70 DATA "YOU ARE ON THE FLIGHT DECK O F THE SHIP A LARGE VISUAL DISPLAY SHOWS A PICTURE OF THE EARTH AND TO YOUR LEF T IS A CONTROL PANEL. EXITS LEAD NOR

TH, SOUTH & EAST."

80 DATA "YOU'RE IN THE CONTROL ROOM. A LARGE BANK OF COMPUTERS ADORN THE NORTH WALL. EXITS LEAD EAST AND WEST."

90 DATA "YOU ARE IN THE REST ROOM. THE RE IS A BED HERE AND A SMALL, BEDSIDE TAB LE. AN SINGLE EXIT LEADS WEST."

100 DATA "YOU'RE IN THE CAPTAIN'S CABI N. RICH, SOFT CARPET COVERS THE FLOOR AND THE WALLS ARE TASTEFULLY PAINTED IN LI GHT SHADES. THERE IS A LARGE BED TO THE LEFTHAND SIDE. THE ONLY EXIT IS TO THE SOUTH"

110 DATA "YOU ARE IN A DARKISH ROOM AN D ALL AROUND YOU ARE LARGE BANKS OF LEAD ACID BATTERIES. THERE IS A DISTINC T SMELL OF GREASE HERE. UP IS A ROPE AND OTHER EXITS LEAD EAST AND SOUTH."

120 DATA "YOU'RE IN A SPARSE ROOM. AN E XIT LEADS EAST AND A RUBBISH CHUTE GOE S DOWN THROUGH THE FLOOR."

130 DATA "YOU ARE STANDING ON ONE SIDE OF A CHASM TO THE SOUTH THAT CAN BE CRO SSED ON AN EXTENDING BRIDGE. A LARGE RED BUTTON STICKS OUT FROM THE WALL."

140 DATA "YOU ARE STANDING ON ONE SIDE OF A CHASM TO THE NORTH. AN EXTENDED BRI DGE LEADS NORTH. ANOTHER EXIT IS TO THE SOUTH."

150 DATA "YOU ARE IN A RUBBISH COMPRES SOR. RUBBISH LIES EVERYWHERE AND THERE IS A SMELL OF DECAYING ORGANIC MATTER. THE COMPRESSOR CONTINUES SOUTHWARDS. ABOVE I S A CHUTE WHICH IS OUT OF REACH."

160 DATA "YOU ARE IN A RUBBISH COMPRES SOR. A STEP LADDER LEADS UPWARDS. THE COM PRESSOR CONTINUES NORTH."

170 DATA "YOU ARE AT A T-JUNCTION OF PASS AGES GOING NORTH, EAST AND WEST."

180 DATA "YOU'RE AT A DEAD END. THERE I S A HATCH IN THE WEST WALL. THE ONLY EX IT IS EAST."

190 DATA "YOU ARE IN A HEXAGONAL CORRI DOR. THE WALLS ARE HEAVILY CORRODED A ND DIGESTIVE JUICES LIE IN POOLS ON THE F LOOR. EXITS LEAD EAST AND WEST."

200 DATA "YOU ARE IN A CONTROL CENTRE. THERE IS A LARGE DOME OVERHEAD. A ENORMO US LASER GUN IS AIMED AT A HOLE IN THE DO ME TOWARDS THE EARTH."

210 DATA "YOU ARE IN A SMALL ROOM. A CO MPUTER IS PLUGGED IN ON THE WEST WALL. AN EXIT LEADS EAST."

220 DATA "YOU ARE IN A SMALL ROOM. ALL AROUND IS DEBRIS. IT LOOKS AS IF THERE HAS BEEN AN EXPLOSION. AN EXIT LEADS EAST ."

230 DATA "YOU ARE IN A DARKISH ROOM AN D ALL AROUND YOU IS THE REMAINS OF LEAD A CID BATTERIES. THERE IS A DISTINC T SMELL OF GREASE HERE. UP IS A ROPE AND OTHER EXITS LEAD EAST AND SOUTH."

240 RESTORE 250: FORN=&C00 TO &C12: READ


```

A: ?N=A: NEXT
250 DATA &A0,&00,&B9,&00,&0A,&49,&1B,&
C9,&16,&F0,&07,&20,&E3,&FF,&C8,&4C,&02,&
0C,&60
260 DIM O(23),O$(23): PROCobj: MODE7: I%=
2: R%=1: DIM D(6,24): B%=0: T%=0: PROCroomdat
a: GOSUB1050
270 REPEAT: INPUT "IN$: UNTIL LEN(IN$)>
0
280 N$="": V$=""
290 V$=LEFT$(IN$,4)
300 L=3: REPEAT: L=L+1: UNTIL L>LEN(IN$)
OR ASC(MID$(IN$,L,1))=32
310 REPEAT: L=L+1: UNTIL L>LEN(IN$) OR A
SC(MID$(IN$,L,1))<>32
320 N$=MID$(IN$,L,4)
330 P=INSTR(V$," ")
340 IF P<>0 V$=LEFT$(V$,P-1)
350 P=INSTR(N$," ")
360 IF P<>0 N$=LEFT$(N$,P-1)
370 GOTO1880
380
390 D%=V
400 IF D(D%,R%)=0 PRINT "YOU CAN'T GO T
HAT WAY": RETURN
410 IF D(D%,R%)>34 GOTO1690
420 IF B%=1 B%=0
430 R%=D(D%,R%): GOSUB 1050: RETURN
440 IF N>14 PRINT "YOU CAN'T PICK THAT
UP": RETURN
450 IF O(N)<>R% PRINT "I DON'T SEE THAT
HERE": RETURN
460 IF I%>=5 PRINT "YOU'RE CARRYING TOO
MUCH": RETURN
470 IF N=8 AND R%=16 PRINT "IT'S WEDGED
TIGHT": RETURN
480 IF N=4 AND LEN(O$(4))=34 PRINT "IT'
S TIED": RETURN
490 IFR%=5 AND O(8)<>-1 OR R%=5 AND O$(
8)="ZU;NUWRO;OTIXS" PRINT "IT'S TOO DARK
TO SEE": RETURN
500 IF N=5 AND LEN(O$(5))<>11 AND O(6)
<>-1 PROCd("BTN:XZU<O:I^ZXS;RO"): RETURN
510 IF N=5 AND LEN(O$(5))<>11 AND O(6)
=-1 PROCd("BTN:STTP;RO;TNO;LROS;BTNI;LRI
^"): O(5)=-1: I%=I%+1: O$(5)="Z:YWZXP;YTC":
PRINT "OK": RETURN
520 O(N)=-1: I%=I%+1: PRINT "OK": RETURN
530 IF N>15 PRINT "I NEVER LET YOU PICK
THAT UP": RETURN
540 IF O(N)<>-1 PRINT "YOU'RE NOT CARRY
ING IT": RETURN
550 IF N=11 AND R%=13 AND O(22)=99 PRO
Cd("OS^:_ITR^:Y^KH^:=:OS^:KI^HH^H:OS^:YN
OOTUZ;HXZUU^I:XTV^H:TU:=:OS^U:I^OIZXOH5O
S^:YIR^_\^:RH:^CO^U^_"): I%=I%-1: O(11)=1
3: O(22)=13: D(2,13)=14: RETURN
560 I%=I%-1: O(N)=R%: PRINT "OK": RETURN
570 PRINT "YOU ARE CARRYING:": IF I%=0 P
RINT "NOT A LOT": RETURN
580 FOR loop=1 TO 14
590 IF O(loop)=-1 PROCd(O$(loop))
600 NEXT
610 PRINT: RETURN
620 IF R%<>11 OR N<>32 PRINT "DON'T BE
A VANDAL": RETURN
630 IF O(10)<>-1 PRINT "WHAT WITH?": RET
URN
640 PROCd("BTN;HVZHS;NK;OS^:YZOO^IR^H;
LROS;OS^:;;;ZU_ITR^:W^"): R%=23: D(1,13)
=23: D(6,10)=23: D(3,12)=23
650 RETURN
660 IF N<>20 OR R%<>19 PRINT "DON'T BE
NASTY": RETURN
670 IF O(21)=19 PRINT "I CAN ONLY KILL
HIM ONCE": RETURN
680 PRINT "LOVE TO BUT HOW?": RETURN
690 IF N<>25 GOTO440
700 B%=1: PRINT "OK": RETURN
710 IF N=0 PRINT "SAY WHAT?": RETURN
720 PRINT "OK": PRINTMID$(IN$,L,255)
730 IF R%<>3 OR D(4,3)=4 RETURN
740 PROCd("OS^:_TTI;RH;MTRX^:TK^IZO^_
ZU^;RO;HWR^_HTK^U")
750 O(15)=3: D(4,3)=4
760 PRINT: RETURN
770 IF O(8)<>-1 PRINT "YOU'RE NOT CARRY
ING IT": RETURN
780 IF T%=1 PRINT "IT DOESN'T WORK": RET
URN
790 O$(8)="Z;WRO;OTIXS"
800 PRINT "OK": RETURN
810 IF O(8)<>-1 PRINT "YOU'RE NOT CARRY
ING IT": RETURN
820 IF LEN(O$(8))<>11 PRINT "IT'S NOT O
N": RETURN
830 O$(8)="ZU;NUWRO;OTIXS": PRINT "OK": R
ETURN
840 IF O(3)<>-1 PROCd("BTN;SZM^:UT;VZO
XS^H"): RETURN
850 IF N=8 AND O(8)=-1 PROCd("OS^:OTIX
S;KWZHORX;ZWW;V^WOH"): RETURN
860 IF N=3 AND R%<>5 PRINT "OK": RETURN
870 IF N<>3 PRINT "YOU CAN'T LIGHT THAT
": RETURN
880 PROCd("OS^:VZOXs;]WZI^H;NK;ZU^;HN^
_UWB;Z:]RI^:YZWW:^CKWT^_H5OS^I^:VNHO;SZ
M^:Y^U^:;;;^CKWTHRM^:_ZH^H;ZITNU^"): GO
TO1810
890 INPUT "NAME OF FILE",F$
900 O=OPENUP(F$): INPUT#O,R%,O$(4),O$(8
),O$(9),O$(5),I%
910 FOR loop=1 TO 23: INPUT#O,O(loop): N
EXT
920 INPUT#O,D(4,3),D(1,4),D(6,10),D(2,
15),D(2,13),D(4,14),D(3,19),D(3,12),D(4,
8)
930 CLOSE#O: PRINT "OK": GOTO1050
940 INPUT "NAME OF FILE",F$
950 O=OPENOUT(F$): PRINT#O,R%,O$(4),O$(
8),O$(9),O$(5),I%
960 FOR loop=1 TO 23: PRINT#O,O(loop): N
EXT
970 PRINT#O,D(4,3),D(1,4),D(6,10),D(2,
15),D(2,13),D(4,14),D(3,19),D(3,12),D(4,
8)
980 CLOSE#O: PRINT "OK": GOTO 1050
990 IF N=23 AND R%=3 PROCd("OS^:_TTI;S
ZH;UT;MRHRYW^:V^ZUH:T:]Y^RU^:;TK^U^_"): R
ETURN
1000 IF N<>19 OR R%<>18 PRINT "YOU CAN'T
DO THAT": RETURN
1010 IF O(13)<>-1 PROCd("BTN;U^_:HTV^O
SRU^;OT;TK^U;RO;LROS"): RETURN
1020 IF O$(9)<>"ZU:^VKOB;YTOOW^" PROCd(
"BTN;ZI^U<O;HOITU^:UTN^S"): RETURN
1030 IF O(23)=18 PRINT "IT'S ALREADY OPE
N": RETURN
1040 O(23)=18: PRINT "OK": RETURN
1050 RESTORE (R%*10): IF R%=5 AND O$(8)=

```

OPERATION KRISTOS

```

"ZU;NUWRO:OTIXS" OR R%>10 AND O$(8)="ZU;
NUWRO:OTIXS" PRINT"IT'S TOO DARK TO SEE"
:RETURN
1060 READ R$:PRINT R$
1070 FOR N=1 TO 23:IF O(N)<>R% GOTO1080
ELSE IF N<15 PRINT"HERE THERE IS ":PROC
d(O$(N)) ELSE PROCd(O$(N))
1080 NEXT
1090 RETURN
1100 IF R%<>20 PRINT"YOU HAVE TO SOLVE
THIS ADVENTURE":RETURN
1110 IF O(2)<>20 OR D(6,10)=11 OR D(4,1
4)=21 PRINT"IT'S BEING JAMMED":RETURN
1120 PROCd
1130 PROCd("OSRH;RH:UT;ORV^;OT;HW^K5BT
N<M^;T;OT::;HZM^;OS^;^ZIOS"):RETURN
1140 IF N<>4 PRINT"YOU CAN'T TIE THAT":
RETURN
1150 IF O(4)<>-1 PRINT"YOU'RE NOT CARRY
ING IT":RETURN
1160 IF R%<>10 OR O(19)<>10 PRINT"NO PO
INT IN DOING THAT HERE":RETURN
1170 PRINT"OK":D(6,10)=11:O$(4)="Z;W^U\
OS;T];ITK^;SZU^RU\;_TLULZI_H":O(4)=10:IX
=IX-1:RETURN
1180 PRINT"YOU CAN'T DO THAT":RETURN
1190 IF N=0 PRINT"FIRE! WHERE?":RETURN
1200 IF N<>14 PRINT"YOU CAN'T FIRE THAT
":RETURN
1210 IF O(N)<>-1 PRINT"YOU DON'T HAVE I
T":RETURN
1220 IF R%<>19 PRINT"DON'T BE VIOLENT":
RETURN
1230 IF O(21)=21 PRINT"AT WHAT?":RETURN
1240 PROCd("OS^;KSTOTU;IZB;SROH;OS^;KWZ
UO;ZU_;OS^I^;RH;ZU;^CKWTHRTU;T];KWZUO;]W
^HS")
1250 O(20)=99:O(21)=19:D(3,19)=20:RETUR
N
1260 IF N<>21 OR R%<>13 PRINT"YOU CAN'T
PRESS THAT":RETURN
1270 PROCd("Z;HXZUU^I;ZXORMZO^H5Z;_RZW;
]WZHS^H;OS^;LZIURU\;<SNVZU<5HN^UWB;Z;
^ITNK;T];::;ITYTOH;ZKK^ZI;ZO;OS^;TOS^I;
HR];OS^;XSZHV")
1280 TIME=0:REPEATUNTIL TIME>60
1290 PROCd("OS^B;TK^U;]RI^"):GOTO1810
1300 IF N=0 PRINT"I don't understand":R
ETURN
1310 IF R%<>4 AND R%<>15 GOTO530
1320 IF O(N)<>-1 GOTO530
1330 IF R%=15 AND N=7 PROCd("OS^;IT_;]W
R^H;HTNOSLZI_H;ZU_;Y^XTV^H::;L^_\^;Y^O
L^U;OS^;LZWWH"):D(2,15)=16:O(7)=98:IX=I
X-1:RETURN
1340 IF R%=15 AND D(1,6)=7 GOTO530
1350 IF N<>12 PROCd("Z;XZV^IZ;HWR_H;TNO
:ZU_;OIZXPH;OS^;TYQ^XOZ;WZH^I;\NU;WTXPH;
TUOT;RO;ZU_;]RI^H"):IX=IX-1:O(N)=99:RETU
RN
1360 PROCd("Z;MR^T;XZV^IZ;HWR_H;TNO;Z
U_;Z;WZH^I::;\NU;WTXPH;TUOT;OS^;V^OZW;ZU
_;]RI^H5OS^;IZB;RH;^]W^XO^;T];OS^;V^
OZW;ZU_;RO::;SROH;OS^;XZV^IZ")
1370 O(12)=4:O(16)=4:IX=IX-1:D(1,4)=6:R
ETURN
1380 IF R%=3 AND N=23 PROCd("OS^;_TTI;V
NHO;TK^U;TU;Z;X^IOZRU;ZXORTU"):RETURN
1390 IF R%=7 AND N=26 AND O(6)=99 PROCd
("BTN;H^;Z;HOR];KR^X^;T];LRI^;RUHR^")

```

```

:O(6)=7:RETURN
1400 IF R%=9 AND N=18 AND O(8)=99 PROCd
("BTN;]RU_;Z;OTIXS5HTV^TU^;VNHO;SZM^;Y^
U;I^Z_RU\;NU^I;OS^;HS^OH"):O(8)=R%:RET
URN
1410 IF N=34 AND R%=10 PRINT"IT LOOKS V
ERY LUXURIOUS":RETURN
1420 IF N=30 AND R%=15 AND O(11)=99 PRO
Cd("BTN;H^;Z;HVZWW;_ITR_"):O(11)=R%:RET
URN
1430 IF N=17 AND R%=18 AND O(23)=18 PRO
Cd("BTN;]RU_;Z;YWZHO^I"):O(14)=18:RETURN
1440 PRINT"YOU SEE NOTHING SPECIAL":RET
URN
1450 IF R%=3 AND D(4,3)=0 PROCd("BTN;_T
U<O;U^_;S^WK;OT;TK^U;Z;_TTI;_T::;BTN$")
:RETURN
1460 IF R%=19 AND O(21)=19 PRINT"KILL I
T":RETURN
1470 PRINT"YOU'RE ON YOU OWN HERE":RETU
RN
1480 IF R%<>10 AND R%<>21 PRINT"YOU CAN
'T DO THAT":RETURN
1490 IF R%=10 AND D(6,10)<>0 PRINT"YOU'
VE ALREADY DONE THAT":RETURN
1500 IF R%=10 PROCd("BTN;IRK;NK;OS^;XZI
K^O;OT;I^M^ZW;Z;ZKRU\STW^"):D(6,10)=41:
O(19)=R%:RETURN
1510 PROCd(" BTN;KNWW;OS^;XTVKNO^I;ZU_;
RO;NUKWN^H5::;OS^I^;RH;Z;VZHHRM^;^CKWTHR
TU"):R%=22:D(4,14)=22:GOTO1050
1520 S%=0
1530 IF D(4,3)=4 S%=100
1540 IF D(1,4)=6 S%=S%+100
1550 IF D(6,10)<>0 S%=S%+100
1560 IF D(2,15)=16 S%=S%+100
1570 IF D(4,14)=22 S%=S%+100
1580 IF D(3,19)=20 S%=S%+100
1590 IF D(2,13)=23 S%=S%+100
1600 PRINT"YOU SCORE ";S%;" OUT OF 800"
:RETURN
1610 IF N<>36 PRINT"YOU CAN'T DRINK THA
T":RETURN
1620 IF O(9)<>-1 PRINT"YOU'RE NOT CARRY
ING IT":RETURN
1630 IFO$(9)="ZU;^VKOB;YTOOW^" PRINT"YO
U ALREADY HAVE":RETURN
1640 PRINT"DELICIOUS":O$(9)="ZU;^VKOB;Y
TOOW^":RETURN
1650 IF N<>2 PRINT"EH??":RETURN
1660 IF O(2)<>-1 PRINT"YOU'RE NOT CARRY
ING IT":RETURN
1670 IF R%<>20 PROCd("BTN<M^;\TO;OT;ZOO
ZXS;RO;OT;OS^;WZH^I"):RETURN
1680 PROCd("TP;BTN;ZOOZXS;RO;OT;OS^;WZH
^I"):IX=IX-1:O(2)=20:D(4,20)=42:RETURN
1690 S%=D(D%,R%)-34
1700 ON S% GOTO1710,1720,1730,1750,1760
,1770,1780,1790,1800
1710 IF B%=1 AND V=1 AND R%=1 R%=2:GOTO
1050 ELSE IF B%=1 AND V=2 AND R%=3 R%=2
:GOTO1050 ELSE PROCd("OS^I^;RH;UT;ZRI;RU
:Z;MZXXNV;ONY^;ZU_;BTN_R^;]ITV;HN])TXZOR
TU"):GOTO1810
1720 PROCd("Z;MR^T;XZV^IZ;WTXPH;TUOT;B
TN;ZU_;Z::;WZH^I;XZUUTU;ZKK^ZIH;ZU_;]R
I^H;ZO;BTN"):GOTO1810
1730 IF O(5)<>-1 PROCd("OS^;ZU_ITR_H;ON
IU;ZITNU_5OS^B;KNWW;TNO::;OS^RI;YWZHO^IH;

```



```

PROCd
;Y^
:RET
KS V
PRO
:RET
PRO
TURN
:RET
N:_T
N$")
LL I
RETU
CAN
YOU'
;XZI
=41:
ZU_:
WTHR
800"
THA
ARRY
T"YO
OB;Y
ARRY
ZOO
WZH
RN
1760
GOTO
RX=2
I;RU
KZOR
OT;B
L;R
H;ON
IH;
ZU_:]RI^:ZO:BTN3UTO:M^IB:URX^:OSZO2"):GO
TO1810
1740 PROCd("Z:WR\SO:TU:BTNI:YWZXP:YTC;]
WZHS^H:ZU_:;OS^:ZU_ITR_H:HOZIO:\TRU\;XI
ZAB:ZU_:OS^U:H^W]6_~HOINXO"):O(10)=8:D(3
,8)=9:O(18)=8:O(17)=99:RX=9:GOTO1050
1750 PROCd("BTN:LZWP:T]]:OS^:_\^:T]]:OS
^:XSZHV:ZU_::_R^"):GOTO1810
1760 PROCd("OS^:LZWWH:T]]:OS^:XTVKI^HHTI
:WXTH^:RU_:;ZU_:BTN:ZI^:HWTLLWB:XINHS^_:
Y^OL^U:OS^:;LZWWH"):GOTO1810
1770 PROCd("BTN:OIRK:TM^I:Z:]^W^I:T]]:O
S^:KWZUO:ZU_:BTNI:YT_B:HOZIOH:OT:_RHHTWM
^:RU:OS^:;:_R^HORM^:QNRX^H"):GOTO1810
1780 PROCd("BTN:]ZWW:_TLU:OS^:STW^:ZU_:
YI^ZP:BTNI:;U^XP"):GOTO1810
1790 PROCd("OS^:VRU^:_CKWT^H:ZU_:BTN:S
ZM^:HZM^:_OS^ZIOS:YNO:BTN:SZM^:_R^:_RU:
OS^:ZOO^VKO"):GOTO1810
1800 IF O$(8)="Z:WRO:OTIXS" AND O(8)=-1
PROCd("BTNI:OTIXS:_T^H:TNO"):TX=1:RX=13
:GOSUB810:GOTO1050 ELSE RX=13:GOTO1050
1810 PRINT"OH DEAR YOU APPEAR TO HAVE D
IED"
1820 PRINT"YOU REALLY MUST BE MORE CARE
FUL IN"
1830 PRINT"FUTURE IF YOU WANT TO GET ON
IN THIS GAME"
1840 ptr=1:GOSUB 1520
1850 END
1860 DEFPROCend
1870 PRINT"WELL DONE.YOU HAVE SAVED THE
EARTH AND ARE AWARDED THE TRIPLE STAR.
YOUR SCORE IS 800 OUT OF 800":END
1880 RESTORE 2030
1890 V=0:REPEAT
1900 READ T$
1910 V=V+1
1920 UNTIL V$=T$ OR T$="*"
1930 IF T$="*" PRINT"Verb not known!!":
GOTO270
1940 IF N$=""" N=0:GOTO2010 ELSE IF V=16
OR V=17 GOTO2010
1950 RESTORE 2040
1960 N=0:REPEAT
1970 READ T$
1980 N=N+1
1990 UNTIL N$=T$ OR T$="*"
2000 IF T$="*" PRINT"Noun not known!!":
GOTO270
2010 ON V GOSUB380.380.380.380.380.380.
440.440.530.570.570.620.620.660.690.710.
710.770.810.840.890.940.990.1300.1100.11
30.1140.1180.1190.1260.1260.1050.1380.14
50.1480.1480.1480.1520.1610.1650.620.620
2020 GOTO270
2030 DATAN.S.E.W.U.D.GET.TAKE.DROP.INVE
.INV.SMAS.BREA.KILL.HOLD.SAY.SHOU.ON.OFF
.LIGH.LOAD.SAVE.OPEN.THRO.TELE.SLEE.TIE.
UNTI.FIRE.PUSH.PRES.LOOK.EXAM.HELP.LIFT.
RIP.PULL.SCOR.DRIN.PRIM.KICK.HIT.*
2040 DATABRAC.MINE.MATC.ROPE.BOX.WIRE.R
OD.TORC.BOTT.LEG.DROI.META.SCRE.BLAS.CAM
E.ANDR.HOLE.BED.HATC.PLAN.BUTT.BRID.DOOR
.OPEN.BREA.PANE.LASE.COMP.CHAS.RUBB.DOME
.BATT.ROBO.CARP.CHUT.JUIC.TABLE.AIRL
.*
2050 END
2060 DEFPROCd(D$)
2070 $&A00=D$

```

```

2080 CALL&C00
2090 PRINT:ENDPROC
2100 DEFPROCobj:RESTORE 2150
2110 FORN=1 TO 23
2120 READ O$(N),O(N)
2130 NEXT
2140 ENDPROC
2150 DATA"Z:O^W^KTIO:YIZX^W^O",-1,"Z:WR
VKRO:VRU^",-1,"Z:YTC:T]]:VZOXS^H",4,"Z:W^
U^OS:T]]:UBWTU:ITK^",5
2160 DATA"Z:HVZWW:YWZXP:YTC:NU^I:;OS^
:LZWPLZB",6,"Z:KR^X^:T]]:HOR]]:LRI^",99,"
Z:V^OZW:IT_",8,"ZU:NUWRO:OTIXS",99
2170 DATA"Z:YTOOW^:T]]:QNRX^",9,"ZU:ZU_I
TR_:W^",-1,"Z:HVZWW:_ITR_",99,"Z:HSRUB;
HS^O:T]]:V^OZW",1,"Z:HXI^L_IRM^I",14,"Z:
YWZHO^I",99
2180 DATA"OS^:_TTI:RH:UTL:TK^U",99
2190 DATA"Z:YWTLU6NK:MR^T:XZV^IZ:SZU^H
:]ITV:OS^:_LZW",99
2200 DATA"ITLH:T]]:]ZX^W^HH:ZU_ITR_H:HRO
:ZO:OS^:;XTVKNO^IH",8
2210 DATA"OS^:I^VZRUH:T]]:ZU_ITR_H:XTM^I
:OS^:]WTI",99
2220 DATA"Z:\ZKRU:STW^:W^Z_H:_TLULZI_H
",99
2230 DATA"Z:WZI^7V^UZXRU\VZU6^ZORU^K
WZUO:YZIH:;BTNI:LZB",19
2240 DATA"Z:WZI^VZU6^ZORU\KWZUO:WR^H
:_Z_:S^I^",99
2250 DATA"OS^:YIR^:_SZH:^CO^U^:_HTNOS
LZI_H",99
2260 DATA"OS^:SZOXS:SZH:Y^U:TK^U^:_OT:
I^M^ZW:Z:;:_ZIP:STW^",99
2270
2280 DEFPROCroomdata
2290 RESTORE 2340
2300 FORL=1 TO 23
2310 READ R$:FORM=1 TO 6:D(M,L)=(ASC(
MID$(R$,M,1))-65):NEXTM
2320 NEXTL
2330 ENDPROC
2340 DATAaAAAAA
2350 DATADBAAAA
2360 DATAAdAAAA
2370 DATAeFDAAA
2380 DATAEAAAAA
2390 DATAHEAAAA
2400 DATAKGIAAA
2410 DATAAfHAAA
2420 DATAAAAIAA
2430 DATAAHAAAA
2440 DATAAlMAKA
2450 DATAAALAAP
2460 DATAALgAAAA
2470 DATANRAVAA
2480 DATAahAAAA
2490 DATAPAAAMA
2500 DATAOATSAA
2510 DATAAARAAA
2520 DATAAAIIRAA
2530 DATAAAATAA
2540 DATAAAOAAA
2550 DATAAAOAAA
2560 DATAANMAKA

```

CAR RACE

• ATARI 16K

Keep the revs up for some super race track action for one or two players.

The race circuit is seen from above and two car—blue and red—are ready at their starting positions. You must aim to break the track record or beat the other car past the finishing line.

The race begins when either joystick is moved. If you crash, your car is put back to the beginning of its current lap. Hitting the track boundary slows down the cars and crashing with other obstacles causes the car to explode.

• Nick Pearce

```

10 GRAPHICS 0:POKE 82,0:?" CAR RACE"
20 ? " by Nick Pearce, (c) 1984":GOSUB 8000
200 GOSUB CARS:GOSUB INFO
360 FOR P=0 TO 1:POKE CRASH+P,0:NEXT P:POKE 19,0:POKE 20,0
399 REM *** LOOP
400 FOR P=0 TO 1
410 IF PEEK(CRASH+P) THEN 600
420 IF PEEK(LAPS+P)=RACELAPS THEN POP :GOTO 800
490 NEXT P
500 IF PEEK(CONSOL)=START THEN 200
550 GOSUB SHOW:IF TIME<6000 THEN 400
560 GOTO 200
599 REM CRASH
600 SOUND P,38,0,14:POKE PC+P,64
610 FOR I=0 TO 30:POKE PCOL+P,RND(0)*255:NEXT I:SOUND P,0,0,0
620 POKE PC+P,80:POKE PX+P,PX(P):POKE PY+P,PY(P)
640 POKE PCOL+P,PCOL(P):POKE PCOL+2+P,PCOL(P+2):POKE PC+P,0:POKE CRASH+P,0
690 GOTO 490
799 REM RACE OVER
800 FOR K=14 TO 2 STEP -1:SOUND 0,40,0,K:NEXT K:GOSUB SHOW
810 RACED=1:CHAMP$="BLUE":IF P THEN CHAMP$="RED"
820 FOR P=0 TO 1:POKE CRASH+P,1:SOUND P,0,0,0:NEXT P
825 FOR N=1 TO LEN(CHAMP$):CHAMP$(N,N)=CHR$(ASC(CHAMP$(N))+128):NEXT N
830 RTIME$=STR$(TIME):IF TIME<BEST THEN BEST=TIME
835 FOR N=1 TO LEN(RTIME$):RTIME$(N,N)=CHR$(ASC(RTIME$(N))+128):NEXT N
840 GOSUB DELAY:GOTO 200
999 REM *** SHOW
1000 TIME=INT((PEEK(19)*256+PEEK(20))/5)
1010 N$=STR$(TIME):M$="0000":M$(5-LEN(N$))=N$
1020 POSITION 28,22:FOR I=1 TO 4:? CHR$(ASC(M$(I))+224);:NEXT I
1040 N$=STR$(PEEK(LAPS)):M$="00":M$(3-LEN(N$))=N$
1050 POSITION 22,22:FOR I=1 TO 2:? CHR$(ASC(M$(I))+128);:NEXT I
1060 N$=STR$(PEEK(LAPS+1)):M$="00":M$(3-LEN(N$))=N$
1070 POSITION 36,22:FOR I=1 TO 2:? CHR$(ASC(M$(I))+128);:NEXT I
1090 RETURN
1999 REM *** CARS AT START
2000 FOR P=0 TO 1:SOUND P,0,0,0
2010 POKE PX+P,PX(P):POKE PY+P,PY(P):POKE PC+P,0
2020 POKE PCOL+P,PCOL(P):POKE PCOL+2+P,PCOL(P+2)
2030 POKE LAPS+P,0:POKE LINE+P,0:POKE COUNT+P,1:POKE CRASH+P,1
2050 NEXT P
2090 RETURN
2999 REM *** INFO DISPLAY
3000 J=0:SOUND 0,100,10,4:GOSUB DELAY:SOUND 0,0,0,0
3002 J=J+1:IF J=4 THEN J=1

```



```

3010 POSITION 0,22: ? BLANK$;: POSITION 0,22
3020 IF J=1 THEN ? " car race": GOTO 3050
3030 IF J<>2 THEN 3040
3032 ? " race is ";: N$=STR$(RACELAPS): FOR N=1 TO LEN(N$): ? CHR$(ASC(N$(N))+224)
;: NEXT N: ? " laps": GOTO 3050
3040 ? " track record"; CHR$(26); " ";: N$=STR$(BEST): FOR N=1 TO LEN(N$): ? CHR$(ASC
(N$(N))+224);: NEXT N
3050 FOR I=1 TO 5: GOSUB DELAY: IF PEEK(CONSOL)=START THEN 3500
3060 POSITION 20,22: ? BLANK$;: POSITION 20,22: ? " PRESS START"
3080 IF PEEK(CONSOL)=START THEN 3500
3090 GOSUB DELAY: POSITION 20,22: ? BLANK$;: POSITION 20,22
3110 IF NOT RACED THEN ? " PRESS START": GOTO 3130
3120 ? " "; CHAMP$; " WON IN "; RTIME$;
3130 IF PEEK(CONSOL)<>START THEN NEXT I: GOTO 3002
3500 POP : POKE 77,0
3510 POSITION 0,22: ? " blue "; CHR$(140); "time"; CHR$(140); " red"; CHR$(140)
3520 POSITION 20,22: ? BLANK$;: FOR P=0 TO 1: SOUND P,186,4,6: NEXT P
3600 POSITION 24,22: ? "go! go! go!": GOSUB DELAY
3630 IF STICK(0)<>15 OR STICK(1)<>15 THEN 3700
3640 POSITION 24,22: ? "go! go! go!": GOSUB DELAY
3680 IF STICK(0)=15 AND STICK(1)=15 THEN 3600
3700 POSITION 20,22: ? BLANK$;
3800 POKE 77,0: FOR K=14 TO 2 STEP -1: SOUND 0,40,0,K: NEXT K
3900 RETURN
3990 REM *** DELAY
4000 FOR K=0 TO 50: NEXT K: RETURN
7999 REM *** INITIALIZE
8000 SETCOLOR 1,1,0: SETCOLOR 2,13,10: SETCOLOR 4,11,2
8100 CONSOL=53279: START=6: SDMCTL=559: POKE 752,1
8120 RACELAPS=10: BEST=6000: RACED=0
8140 SHOW=1000: CARS=2000: INFO=3000: DELAY=4000
8220 ? "You are presented with a birds-eye view"
8230 ? "of a motor racing circuit. A blue car"
8240 ? "and a red car are ready at their start-"
8250 ? "ing positions. The race is over "; RACELAPS; " laps"
8260 ? "of the track (in clockwise direction)."
8270 ? "Try to break the track record or beat"
8280 ? "the other car to the finishing line!!"
8290 ? "Use a joystick to drive your car (BLUE"
8300 ? "in jack 1 and RED in jack 2); press the"
8310 ? "trigger to go at full speed. The race"
8320 ? "begins when either stick is moved. The"
8330 ? "time is measured in tenths of seconds."
8340 ? "If you crash, your car is put back to"
8350 ? "the beginning of it's current lap."
8360 ? "To restart the race at any time, press"
8370 ? "the START key."
8390 ? " PRESS START";
8400 IF PEEK(CONSOL)<>START THEN 8400
8500 FOR I=1 TO 2: POSITION 12,23: SOUND 0,100,10,8: ? "wait 15 seconds";
8520 GOSUB DELAY: POSITION 12,23: SOUND 0,0,0,0: ? " ";: GOSUB DELAY: N
EXT I
9000 RAMTOP=106: MYTOP=PEEK(RAMTOP)-12: POKE RAMTOP,MYTOP: MEM=MYTOP*256
9020 GRAPHICS 0: SETCOLOR 4,11,2: POKE SDMCTL,0: D=PEEK(560)+256*PEEK(561)
9040 POKE D+3,68: FOR I=D+6 TO D+26: POKE I,4: NEXT I: POKE D+27,6: POKE D+28,6
9100 CHIGH=206: COUNT=207: PCOL=704: PP=1536: PC=1544: PX=1548: PY=1552
9110 CRASH=1560: SKID=1562: LAPS=1564: LINE=1566
9120 PMBASE=54279: GRCTL=53277: GPRIOR=623: CHBAS=756
9150 DIM M$(6), N$(6), PX(1), PY(1), PCOL(3), CHAMP$(4), BLANK$(20), RTIME$(5)
9160 BLANK$=""
9200 DIM FILL$(17): FOR I=1 TO 17: READ J: FILL$(I)=CHR$(J): NEXT I

```

CAR RACE

```

9210 DATA 104,104,104,133,208,160,0,132,207,104,104,145,207,200,208,251,96
9220 DIM COPY$(23):FOR I=1 TO 23:READ J:COPY$(I)=CHR$(J):NEXT I
9230 DATA 104,104,104,133,206,104,104,133,208,160,0,132,205,132,207,177,205,145,
207,200,208,249,96
9249 REM CHARSET
9250 FOR I=0 TO 3:K=USR(ADR(COPY$),224+I,MYTOP+I):NEXT I
9260 FOR I=24 TO 127:READ J:POKE MEM+I,J:NEXT I:POKE CHBAS,MYTOP
9270 DATA 85,85,85,85,85,85,85,85
9272 DATA 1,13,13,5,21,213,213,85
9274 DATA 85,87,87,84,80,112,112,64
9276 DATA 64,112,112,80,84,87,87,85
9278 DATA 85,213,213,21,5,13,13,1
9280 DATA 245,85,85,85,85,85,85,85
9282 DATA 85,85,85,85,85,85,85,95
9284 DATA 213,213,213,213,85,85,85,85
9286 DATA 85,85,85,85,67,87,87,87
9288 DATA 0,12,51,0,0,48,204,0
9290 DATA 101,101,101,101,101,101,101,101
9292 DATA 48,12,48,12,48,12,48,12
9294 DATA 0,0,255,0,255,0,255,0
9299 REM PMG
9300 FOR I=MYTOP+8 TO MYTOP+11:K=USR(ADR(FILL$),I,0):NEXT I
9310 K=USR(ADR(FILL$),6,0):POKE PMBASE,MYTOP+4:POKE GPRIOR,33
9340 FOR I=0 TO 3:POKE PP+I,MYTOP+8+I:NEXT I
9350 FOR I=1605 TO 1611:READ J:POKE I,J:NEXT I:DATA 32,48,16,0,48,32,16
9380 PCOL(0)=122:PCOL(1)=58:PCOL(2)=254:PCOL(3)=254
9390 PX(0)=158:PX(1)=166:PY(0)=54:PY(1)=54
9400 FOR I=0 TO 79:READ J:POKE MEM+1792+I,J:NEXT I:POKE CHIGH,MYTOP+7
9420 DATA 0,24,24,24,24,24,24,0
9424 DATA 0,36,36,0,0,36,36,0
9428 DATA 0,0,60,60,60,0,0,0
9430 DATA 0,36,36,0,0,36,36,0
9440 DATA 0,16,56,56,28,28,8,0
9450 DATA 16,16,0,36,36,0,8,8
9470 DATA 0,8,28,28,56,56,16,0
9480 DATA 8,8,0,36,36,0,16,16
9482 DATA 73,235,126,63,124,254,91,16
9484 DATA 33,132,16,66,8,34,64,16
9499 REM VBI
9500 FOR I=1700 TO 1709:READ J:POKE I,J:NEXT I:POKE 1702,MYTOP+4
9520 DATA 104,162,0,160,0,169,7,76,92,228
9550 FOR I=MEM+1024 TO MEM+1346:READ J:POKE I,J:NEXT I:K=USR(1700)
9600 DATA 216,162,1,189,24,6,208,66,214,207,208,62,188,132,2,200
9610 DATA 152,24,125,26,6,149,207,188,120,2,152,41,8,208,3,254
9620 DATA 12,6,152,41,4,208,3,222,12,6,152,41,2,208,6,254
9630 DATA 16,6,254,16,6,152,41,1,208,6,222,16,6,222,16,6
9640 DATA 192,15,240,6,185,64,6,157,8,6,138,168,200,200,24,189
9650 DATA 8,6,105,8,153,8,6,189,12,6,153,12,6,189,16,6
9660 DATA 153,16,6,202,16,157,162,1,189,24,6,208,55,138,10,168
9670 DATA 169,0,153,1,210,189,120,2,201,15,240,40,169,38,153,1
9680 DATA 210,189,26,6,240,5,169,6,153,1,210,189,120,2,74,133
9690 DATA 209,189,132,2,208,4,169,0,240,2,169,16,24,101,209,105
9700 DATA 36,153,0,210,202,16,193,162,1,160,0,189,4,208,41,2
9710 DATA 240,1,200,189,30,6,240,9,152,208,12,157,30,6,254,28
9720 DATA 6,152,240,3,157,30,6,169,0,157,26,6,189,4,208,41
9730 DATA 8,208,7,189,12,208,41,3,240,7,169,1,157,24,6,208
9740 DATA 37,189,4,208,208,7,169,20,157,26,6,208,25,41,4,208
9750 DATA 18,138,168,200,200,185,4,208,41,4,208,7,185,12,208,41
9760 DATA 12,240,3,254,26,6,202,16,160,141,30,208,162,3,189,0
9770 DATA 6,133,204,189,4,6,133,203,169,0,160,255,200,145,203,192
9780 DATA 7,208,249,189,16,6,157,4,6,133,203,189,8,6,133,205

```



```

AND x%>=cloudx(t) AND x%<cloudx(t)+3 TH
EN dy%=-dy%:z%=1:SOUND 1,200,15,5
1035 NEXT
1037 IF z%=1 THEN GOTO 1060
1040 LOCATE x%,y%
1050 PRINT CHR$(INT(193+4*RND(1)))
1060 b=b+1:IF b=3 THEN GOSUB 1500:b=0
1499 RETURN
1500 '**** Move Bailiff ****
1510 bailiff=bailiff+1:IF bailiff=30 TH
N LOCATE bailiff,6:PRINT CHR$(32);CHR$(3
2);CHR$(11);CHR$(8);CHR$(32):bailiff=1
1520 LOCATE bailiff,6:PRINT bailiff$
1530 PEN 3:LOCATE bailiff+2,6:PRINT head
$
1540 PEN 0
1550 Poun%=Poun%-1:PEN 2:LOCATE 2,3:PRIN
T"#";Poun%:SOUND 2,1,5,15
1560 IF Poun%<1 THEN GOTO 4000
1570 PEN 0:b=0
1599 RETURN
2000 '**** Move Stretcher Team ****
2010 LOCATE man%,23:PRINT CHR$(32);CHR$(
9);CHR$(9);CHR$(32);CHR$(10);CHR$(8);CHR
$(32):man%=man%-1
2020 LOCATE man%,23:PEN 3:PRINT top$
2030 LOCATE man%,24:PEN 0:PRINT bottom$
2035 RETURN
2040 LOCATE man%,23:PRINT CHR$(32);CHR$(
8);CHR$(10);CHR$(32);CHR$(11);CHR$(9);CH
R$(9);CHR$(32):man%=man%+1
2060 LOCATE man%,23:PEN 3:PRINT top$
2070 LOCATE man%,24:PEN 0:PRINT bottom$
2999 RETURN
3000 '**** Collisions? ****
3010 IF x%>=bailiff AND x%<bailiff+3 TH
N scf%=1:SOUND 2,16,20,15,1,10
3020 dy%=-dy%
3999 RETURN
4000 '**** Lose A Life ****
4010 lives%=lives%-1:IF lives%<1 THEN GO
TO 9000
4015 PEN 2:LOCATE 18,2:PRINT"MEN";lives%
4020 PEN 2:LOCATE 15,12:PRINT"AAARGH !":
SOUND 1,1500,15,15,5,5,12
4030 FOR t=1 TO 400:NEXT
4100 ll%=0:Poun%=200:man%=15
4110 GOTO 30
4500 '**** Score ****
4505 sc%=sc%+Poun%
4510 PEN 3
4520 LOCATE 2,2:PRINT"1UP ";sc%
4530 doll%=100:Poun%=200
4535 PEN 2:LOCATE 2,3:PRINT"#";Poun%:LOC
ATE 30,3:PRINT"$";doll%
4540 screen%=screen%+1
4590 RETURN
5000 '**** Characters ****
5010 SYMBOL AFTER 139
5020 SYMBOL 193,2,5,10,140,80,36,216,192
5030 SYMBOL 194,192,220,32,80,142,144,24
5040 SYMBOL 195,130,68,36,28,42,75,83,0
5050 SYMBOL 196,6,230,24,23,48,72,140,0
5060 SYMBOL 140,238,0,187,0,238,0,187,0
5070 SYMBOL 197,63,127,127,255,255,127,6
3,14
5080 SYMBOL 198,12,190,191,223,255,255,2
51,112

```

```

5090 SYMBOL 199,56,124,126,254,252,248,1
12
5095 SYMBOL 207,0,0,0,0,60,126,126,255
5100 SYMBOL 200,62,88,170,95,182,120,60,
24
5120 SYMBOL 202,60,94,33,221,247,15,126,
60
5125 SYMBOL 203,126,118,247,251,60,118,1
02,119
5130 SYMBOL 204,0,0,0,255,127,128,0,0
5140 SYMBOL 205,126,110,239,223,60,110,1
02,238
5145 SYMBOL 206,0,0,0,255,254,1,0,0
5150 SYMBOL 208,62,88,170,95,178,110,56,
128
5160 SYMBOL 209,252,42,21,255,255,143,4,
7
5170 '**** Set character strings ****
5180 bottom$=CHR$(203)+CHR$(9)+CHR$(9)+C
HR$(205)
5190 top$=CHR$(200)+CHR$(10)+CHR$(204)+C
HR$(206)+CHR$(11)+CHR$(202)
5200 cloud$=CHR$(197)+CHR$(198)+CHR$(199
)
5210 bailiff$=CHR$(32)+CHR$(209)+CHR$(11
)+CHR$(8)+CHR$(32)+CHR$(207)
5220 head$=CHR$(208)
5999 RETURN
6000 '**** Set Screen Up ****
6020 BORDER 11:PAPER 1:CLS
6030 a$(1)=" "
6040 a$(2)=""
6045 PEN 2:PAPER 0
6050 FOR t=5 TO 25:LOCATE 32,t:PRINT a$(
1):NEXT
6060 FOR t=5 TO 25 STEP 4:LOCATE 32,t:PR
INT a$(2):NEXT
6070 PLOT 1,340,0:DRAW 636,340,0:DRAW 63
6,399,0:DRAW 1,399,0:DRAW 1,340,0
6080 PAPER 1:LOCATE 2,2:PEN 3:PRINT"1UP"
/sc%:LOCATE 30,2:PRINT"HI";hi%:PEN 2:LOC
ATE 2,3:PRINT"#";Poun%:LOCATE 30,3:PRINT
"$";doll%:LOCATE 18,2:PRINT"MEN";lives%
6090 PEN 3:FOR t=1 TO screen%:cloudx(t)=
INT(1+26*RND(1)):cloudy(t)=INT(8+5*RND(0
)):LOCATE cloudx(t),cloudy(t):PRINT clou
d$:NEXT
6999 RETURN
8000 '**** Title Screen ****
8005 INK 1,11:INK 2,6:INK 0,0:INK 3,26:M
ODE 1:PAPER 1:CLS
8010 t$="S Y A E S":b$=" K C P R ":yc=2
8020 IF yc=1 THEN yc=2:cy=1 ELSE yc=1:cy
=2
8030 PEN 0:LOCATE 4,7:PRINT"The Cast":PE
N 2:LOCATE 4,8:PRINT"---"
8040 PEN 3:LOCATE 4,10:PRINT top$:PEN 0:
LOCATE 4,11:PRINT bottom$:" ..... Fred
& Bill"
8050 LOCATE 5,13:PRINT bailiff$:PEN 3:LO
CATE 7,13:PRINT head$:PEN 0:PRINT" ..
... Super Bailiff"
8060 LOCATE 6,15:PRINT CHR$(196):" ..
... The Debttter"
8070 PEN 3:LOCATE 5,17:PRINT cloud$:PEN
0:PRINT" ..... The Cloud"
8080 PEN 2:LOCATE 10,20:PRINT"Press i fo
r instructions":LOCATE 10,22:PRINT"Press
any key to Play"
8085 PEN 3:LOCATE 16,yc:PRINT t$:PEN 0:L

```



```

LOCATE 16,CY:PRINT B$
8090 C$=INKEY$:IF C$="" THEN 8020
8100 IF C$="I" OR C$="I" THEN I=1
8999 RETURN
9000 '*** Game over ***
9010 A$="GAME OVER"
9020 PEN 2:LOCATE 18,2:PRINT"MEN":LIVES%
9035 PEN 0
9040 IF SC%>HI% THEN HI%=SC%
9050 FOR T=1 TO 9:LS=MID$(A$,T,1):LOCATE
T+14,12:PRINT L$:FOR S=1 TO 200:NEXT
9060 NEXT
9999 GOTO 8
10000 'Instructions
10010 MODE 1:PEN 0:PAPER 1:CLS:LOCATE 14
,1:PRINT"Instructions"
10020 LOCATE 14,2:PRINT"-----"
10030 PRINT:PRINT"Once upon a time in th
e city of London ascertain female Politic
ian who will remain unnamed, became

```

```

Prime Minister of jolly old England.
10040 PRINT"Suddenly jolly old England b
ecame not sojolly as the # went down aga
inst the $."
10050 PRINT"As this happened the People
of England became bankrupt and started
committing suicide."
10060 PRINT:PRINT"You guide Fred & Bill
the tax collectors left & right with your
joystick to try to bounce the People u
P to SUPER BAILIFF at the top of the scre
en."
10070 PRINT"If you do this you get the t
ax the Person owes you if you don't
before the # reaches 0 then you lose a
life."
10080 PRINT:PRINT:PRINT:PRINT"
Press a key to start"
10090 C$=INKEY$:IF C$="" THEN 10090
10100 RETURN

```

LOW LEVEL

• CBM 64

Test your flying skills as you fly close to the ground to avoid enemy radar, miss buildings and dodge the hostile planes the they swoop down from the sky.

You don't have any weapons system but you do have large fuel reserves. But are you up to facing this ordeal?

Words inside square brackets are

converted Commodore graphic symbols. For instance [CLEAR] means press SHIFT and CLR/HOME key.

The ^ symbol, which appears in a number of lines, is the up-arrow ↑ (just to the left of the restore key).

• Ian Brown

```

10 POKE18025,0:POKE18026,1:POKE18020,0
12 REM ** ENABLE ALL 8 SPRITES **
13 V=53248:POKEV+21,255
15 REM ** SET ALL SPRITES TO MULTICOLOUR MODE **
16 POKEV+28,255
17 REM ** COLOUR ALL SPRITES **
18 POKEV+37,0:REM MULTICOLOUR 1
19 POKEV+38,12:REM MULTICOLOUR 2
20 POKEV+39,15:POKEV+46,15:POKEV+45,15:REM COLOUR SPRITES 0,6,7 LIGHT GREY
21 POKE53280,6:POKE53281,6:REM *** BACKGROUND AND BORDER BLUE ***
22 PRINT"[CLEAR][WHITE]LOW LEVEL WAS WRITTEN BY IAN BROWN."
23 PRINT"GRAPHICS BEING DEFINED AND MACHINE CODE BEING ENTERED.[RIGHT]PLEASE WAI
T..."
24 REM ** DISABLE I/O AND KEYBOARD SCAN **
26 POKE56334,PEEK(56334)AND254:POKE1,PEEK(1)AND251
28 REM ** COPY FIRST 63 CHARACTERS FROM ROM **
30 FORI=0TO63:FORJ=0TO7:POKE12288+I*8+J,PEEK(53248+I*8+J):NEXTJ:NEXTI
32 REM ** ENABLE I/O AND KEYBOARD SCAN **
34 POKE1,PEEK(1)OR4:POKE56334,PEEK(56334)OR1
36 REM ** COPY IN PROGRAMMABLE CHARACTERS FROM DATA STATEMENTS **
38 FORI=1TO13:READCHAR:FORJ=0TO7:READNO:POKE12288+CHAR*8+J,NO:NEXTJ:NEXTI
39 REM ** CLEAR MEMORY FOR THE SPRITES **
40 FORX=0TO129:POKE832+X,0:NEXTX
41 REM ** COPY IN PLANE CONTROLLED BY PLAYER FROM DATA STATEMENTS **
42 FORX=0TO34:READQ:POKE832+X,Q:NEXTX
43 REM ** COPY IN ENEMY PLANE FROM DATA **
44 FORX=0TO34:READQ:POKE896+X,Q:NEXTX
59 REM *** SET CHAR MEMORY POINTER TO MEM. 12288 ***

```

LOW LEVEL

```

60 POKE53272,(PEEK(53272)AND240)+12
69 REM *** ENABLE MULTICOLOUR MODE PROG. CHARS. ***
70 POKE53270,PEEK(53270)OR16
71 POKE53282,11:REM MULTICOLOUR 1
72 POKE53283,0:REM MULTICOLOUR 2
79 REM ** POKE DATA FOR M/C SCROLLING ROUTINE INTO MEMORY **
80 FORX=0TO97:READQ:POKE16384+X,Q:NEXTX
84 REM ** POKE DATA FOR M/C ROUTINE MOVING ENEMY PLANE 1 INTO MEMORY **
85 FORX=0TO49:READQ:POKE16500+X,Q:NEXTX
89 REM ** POKE DATA FOR M/C ROUTINE MOVING ENEMY PLANE 2 INTO MEMORY **
90 FORX=0TO89:READQ:POKE16600+X,Q:NEXTX
94 REM ** POKE IN M/C ROUTINE MOVING ENEMY PLANES 3 AND 4 INTO MEMORY **
95 FORX=0TO89:Q=PEEK(16600+X):P=Q:IFQ=4THEN100
96 IFQ=5THEN102
97 IFQ=90THEN104
99 POKE16700+X,Q:POKE16800+X,P:NEXTX:GOTO107
100 IFPEEK(16601+X)=208THENQ=6:P=8
101 GOTO99
102 IFPEEK(16601+X)=208THENQ=7:P=9
103 GOTO99
104 IFPEEK(16601+X)=70THENQ=91:P=92
105 GOTO99
107 POKE16714,8:POKE16716,8:POKE16723,247:POKE16739,8:POKE16744,140
108 POKE16814,16:POKE16816,16:POKE16823,239:POKE16839,16:POKE16844,180
109 REM ** POKE IN M/C ROUTINE MOVING ENEMY PLANE 5 INTO MEMORY **
110 FORX=0TO49:Q=PEEK(16500+X):IFQ=2THEN113
111 IFQ=3THEN115
112 POKE16900+X,Q:NEXTX:GOTO118
113 IFPEEK(16501+X)=208THENQ=10
114 GOTO112

115 IFPEEK(16501+X)=208THENQ=11
116 GOTO112
118 POKE16914,32
119 POKE16916,32
120 POKE16923,223
121 POKE16939,32
122 POKE16944,155
124 REM ** POKE IN M/C CONTROL ROUTINE FROM DATA **
125 FORX=0TO36:READQ:POKE17000+X,Q:NEXTX
127 REM ** POKE IN M/C ROUTINE MOVEING PLANE CONTROLLED BY PLAYER FROM DATA **
128 FORX=0TO93:READQ:POKE17100+X,Q:NEXTX
129 REM ** POKE IN M/C ROUTINE CHECKING FOR COLLISIONS FROM DATA **
130 FORX=0TO27:READQ:POKE17200+X,Q:NEXTX
132 REM ** POKE IN M/C ROUTINE MAKING PLANE ENGINE SOUND EFFECT **
133 FORX=0TO37:READQ:POKE17300+X,Q:NEXTX
138 GOTO747
173 REM *** SET UP BACKGROUND ***
174 PRINT"[CLEAR][DOWN][DOWN][DOWN][DOWN][DOWN][DOWN][DOWN][DOWN][DOWN][DOWN][DOWN][c
4] @[[\
175 PRINT"[c 4] @[[\ @[[\ ]^^^ @[[\ @[[\
176 PRINT"[c 8] ]^^^ ]^^^ ]^^^ $&' $&' ]^^^ $&' ]^^^"
177 PRINT"[c 6] ]^^^ ]^^^ ]^^^ # # ]^^^ # ]^^^"
178 PRINT"[c 6] )_!!!)_!!!)_!!!))(( ))(( ))_!!!))(( ))_!!!) "
179 PRINT"[DOWN]"
180 PRINT"[CYAN] ) ) ) ) ) [GREEN] ) ) ) ) ) )"
181 PRINT"[CYAN] ) ) ) ) ) [GREEN] ) ) ) ) )"
182 PRINT"[CYAN] ) ) ) ) ) [GREEN] ) ) ) ) )"
183 PRINT"[CYAN] ) ) ) ) ) [GREEN] ) ) ) ) )"
184 PRINT"[CYAN] ) ) ) ) ) [GREEN] ) ) ) ) ) ) ) ) ) ) ) [UP][UP][UP][UP][U
P][UP][WHITE]"
185 REM ** COLOUR 4 CHARACTER SPACES AT THE RIGHT OF THE SCREEN **
186 POKE55855,15:POKE55815,15:POKE55775,11:POKE55735,11
187 REM ** CONTROL VERTICAL STARTING DIRECTION OF SPRITES 2,3,4 **
189 POKE18010,0:POKE18011,0:POKE18012,222
197 REM ** SET SPRITE POINTERS TO THE AREAS OF MEMORY WHICH CONTAIN THE DATA **
198 POKE2040,13:POKE2046,13:POKE2047,13:REM SPRITES 0,6,7,BLOCK 13
199 FORX=1TO5:POKE2040+X,14:NEXTX:REM SPRITES 1 TO 5 ,BLOCK 14
200 REM ** POSITION SPRITES FOR START **

```



```

201 POKE53248,60:POKE53249,70:REM SPRITE 0
202 POKE53264,126:POKE53250,100:POKE53251,70:REM SPRITE 1
203 POKE53252,255:POKE53253,110:REM SPRITE 2
204 POKE53254,100:POKE53255,55:REM SPRITE 3
205 POKE53256,170:POKE53257,85:REM SPRITE 4
206 POKE53258,125:POKE53259,100:REM SPRITE 5
207 POKE53260,0:POKE53261,176:REM SPRITE 6
208 POKE53262,220:POKE53263,176:REM SPRITE 7
249 REM ** CLEAR COLLISION DETECTION LOCATIONS **
250 POKEV+30,255:K=PEEK(V+30):POKEV+31,255:K=PEEK(V+31)
699 REM ** CALL M/C ROUTINES, INCREASE SCORE AND CHECK IF LIFE LOST **
700 SYS17000:IFPEEK(18020)>0THEN700
701 IFPEEK(18026)<>0THENS=S+1:PRINT"[UP]      SCORE ";S:GOTO700
702 REM ** LIFE LOST **
703 POKE18026,1
704 REM ** SOUND EFFECT **
706 POKE54296,15:POKE54295,0
707 W=54272:POKEW+6,0:POKEW+5,12:POKEW+1,20:POKEW+4,129:FORN=1TO2200:NEXTN
708 POKEW+4,0
734 IFPEEK(V+21)=255THENPOKEV+21,191:GOTO174
735 IFPEEK(V+21)=191THENPOKEV+21,63:GOTO174
736 IFS>HTHENH=S
738 PRINT"[HOME][WHITE][RIGHT][RIGHT][RIGHT][RIGHT][RIGHT][RIGHT][RIGHT][RIGHT][RIGHT][RIGHT][RIGHT][RIGHT][RIGHT][RIGHT]GAME OVER"
739 FORX=1TO4000:NEXTX
746 REM ** TITLE SCREEN **
747 POKEV+21,192:POKEV+16,64:PRINT"[CLEAR][DOWN][RIGHT][RIGHT][RIGHT][RIGHT][RIGHT][RIGHT][RIGHT][RIGHT][RIGHT][RIGHT][RIGHT][RIGHT][RIGHT][RIGHT]
HT)[RIGHT][RIGHT][RIGHT][RIGHT][RIGHT][RIGHT][RIGHT][RIGHT][RIGHT][RIGHT][RIGHT][RIGHT][RIGHT][RIGHT][RIGHT]
LOW LEVEL"
748 POKE2046,13:POKE2047,13:POKEV+14,74:POKEV+15,70
749 POKEV+12,10:POKEV+13,70
750 PRINT"[DOWN][RIGHT][RIGHT][RIGHT][RIGHT][RIGHT][RIGHT][RIGHT][RIGHT][RIGHT][RIGHT][RIGHT][RIGHT][RIGHT][RIGHT][RIGHT]
RIGHT)[RIGHT][RIGHT][RIGHT][RIGHT][YELLOW]HIGH SCORE[WHITE]";H
752 PRINT"[DOWN][RIGHT][RIGHT][RIGHT][RIGHT][RIGHT][RIGHT][RIGHT][RIGHT][RIGHT][RIGHT][RIGHT][RIGHT][RIGHT][RIGHT][RIGHT]
RIGHT)[RIGHT][RIGHT][RIGHT][RIGHT][YELLOW]LAST SCORE[WHITE]";S
754 PRINT"[DOWN][RIGHT][RIGHT][RIGHT][RIGHT][RIGHT][RIGHT][RIGHT][RIGHT][RIGHT][RIGHT][RIGHT][RIGHT][RIGHT][RIGHT][RIGHT]
RIGHT)[YELLOW]USE JOYSTICK IN PORT 2"
756 PRINT"[DOWN][RIGHT][RIGHT][RIGHT][RIGHT][RIGHT][RIGHT][RIGHT][RIGHT][RIGHT][RIGHT][RIGHT][RIGHT][RIGHT][RIGHT][RIGHT]
RIGHT)OR THE FOLLOWING KEYS:[DOWN]"
758 PRINTSPC(15)"[WHITE]A[YELLOW] = UP"
760 PRINTSPC(15)"[WHITE]Z[YELLOW] = DOWN"
762 PRINTSPC(15)"[WHITE]N[YELLOW] = LEFT"
764 PRINTSPC(15)"[WHITE]M[YELLOW] = RIGHT[WHITE]"
766 PRINT"[DOWN]AFTER BOMBING AN ENEMY INSTALLATION YOU ARE RETURNING HOME,OUT O
F ";
768 PRINT"AMMO BUT HIGH ON FUEL. YOU MUST FLY LOW OVER THE CITY SO AS TO ";
770 PRINT"KEEP UNDER THE ENEMY RADAR.      AVOID THE SUICIDAL ENEMY PLANES WHICH";
772 PRINT"      FLY DIRECTLY IN YOUR PATH AND WATCH OUT FOR THE TOPS OF THE";
774 PRINT" SKYSCRAPERS."
776 PRINT"[DOWN]";SPC(6);"[YELLOW]PRESS SPACE OR FIRE TO START[HOME]";S=0
777 IFPEEK(197)=60ORPEEK(56320)=111THEN780
779 GOTO777
780 FORX=0TO13:POKEV+2+X,0:NEXTX:POKEV+21,255:GOTO174
999 REM ** DATA FOR FLATS **
1000 DATA 0,255,191,175,171,169,169,169,169: REM @
1001 DATA 27,255,255,255,255,85,85,105,105: REM [
1002 DATA 28,192,240,252,255,85,85,105,105: REM \
1003 DATA 29,169,169,169,169,169,169,169,169: REM ]
1004 DATA 30,85,85,105,105,85,85,105,105: REM ^
1005 DATA 31,169,233,249,255,255,255,255,255: REM _
1006 DATA 33,85,85,85,255,255,255,255,255: REM !
1009 REM ** DATA FOR PALM TREES **
1010 DATA 35,8,8,8,8,8,8,8,8: REM #
1011 DATA 36,0,0,3,15,63,252,240,195: REM $
1012 DATA 38,0,48,63,255,252,63,255,200: REM &
1013 DATA 39,0,0,192,240,252,63,15,195: REM '
1014 DATA 40,251,251,251,234,255,255,255,255: REM (
1015 DATA 41,255,255,255,255,255,255,255,255: REM )
1019 REM ** DATA FOR PLANE CONTROLLED BY PLAYER **

```

LOW LEVEL

```

1020 DATA 65.80,0.144,84.0,165
1021 DATA 85.80,170.170,148,170,170
1022 DATA 169,170,85,170,170,85,170
1023 DATA 85,85,85,1,84,0,1
1024 DATA 80,0,1,64,0,0,0
1029 REM ** DATA FOR ENEMY PLANES **
1030 DATA 0.5,65,0.21,7,5
1031 DATA 85,95,23,255,255,127,255
1032 DATA 255,255,85,255,255,85,255
1033 DATA 85,85,85,0.21,64,0
1034 DATA 5,64,0,1,64,0,0
1039 REM ** DATA FOR M/C SCROLLING ROUTINE **
1040 DATA 173,144,5,141,80,70,173,184,5,141
1041 DATA 81,70,173,224,5,141,82,70,173,8
1042 DATA 6,141,83,70,173,48,6,141,84,70
1043 DATA 162,1,189,144,5,157,143,5,189,184
1044 DATA 5,157,183,5,189,224,5,157,223,5
1045 DATA 189,8,6,157,7,6,189,48,6,157
1046 DATA 47,6,232,224,40,208,221,173,80,70
1047 DATA 141,183,5,173,81,70,141,223,5,173
1048 DATA 82,70,141,7,6,173,83,70,141,47
1049 DATA 6,173,84,70,141,87,6,96
1050 REM ** DATA FOR M/C ROUTINE MOVING ENEMY PLANE 1 **
1051 DATA 206,2,208,173,2,208,201,0,208,38
1052 DATA 173,16,208,41,2,201,2,208,16,173
1053 DATA 16,208,41,253,141,16,208,169,255,141
1054 DATA 2,208,76,164,64,173,16,208,9,2
1055 DATA 141,16,208,169,100,141,2,208,96,96
1059 REM ** DATA FOR M/C ROUTINE MOVING ENEMY PLANE 2 **
1060 DATA 206,4,208,173,4,208,201,0,208,38
1061 DATA 173,16,208,41,4,201,4,208,16,173
1062 DATA 16,208,41,251,141,16,208,169,255,141
1063 DATA 4,208,76,8,65,173,16,208,9,4
1064 DATA 141,16,208,169,150,141,4,208,173,90
1065 DATA 70,201,0,208,17,206,5,208,173,5
1066 DATA 208,201,50,240,1,96,169,1,141,90
1067 DATA 70,96,238,5,208,173,5,208,201,120
1068 DATA 240,1,96,169,0,141,90,70,96,96
1069 REM ** DATA FOR M/C CONTROL ROUTINE WHICH CALLS ALL OTHER ROUTINES **
1070 DATA 32,116,64,32,216,64,32,60,65,32
1071 DATA 160,65,32,4,66,238,100,70,32,204
1072 DATA 66,173,100,70,201,16,208,8,32,0
1073 DATA 64,169,0,141,100,70,96
1079 REM ** DATA FOR M/C FOR MOVING PLANE CONTROLLED BY PLAYER **
1080 DATA 32,148,67,173,0,220,201,126,240,32
1081 DATA 201,125,240,44,201,119,240,44,201,123
1082 DATA 240,56,173,197,0,201,10,240,13,201
1083 DATA 12,240,25,201,36,240,25,201,39,240
1084 DATA 37,96,206,1,208,173,1,208,201,49
1085 DATA 208,5,169,50,141,1,208,96,238,1
1086 DATA 208,96,238,0,208,173,0,208,201,61
1087 DATA 208,5,169,60,141,0,208,96,206,0
1088 DATA 208,173,0,208,201,49,208,245,169,50
1089 DATA 141,0,208,96
1099 REM ** DATA FOR M/C ROUTINE CHECKING FOR COLLISIONS **
1100 DATA 173,31,208,41,1,201,1,240,10,173
1101 DATA 30,208,41,1,201,1,240,1,96,169
1102 DATA 0,141,100,70,141,106,70,96
1109 REM ** DATA FOR M/C ROUTINE MAKING PLANE ENGINE SOUND EFFECT **
1110 DATA 169,0,141,4,212,169,55,141,6,212
1111 DATA 169,4,141,24,212,169,150,141,5,212
1112 DATA 169,6,141,1,212,169,200,141,0,212
1113 DATA 169,33,141,4,212,76,48,67

```

READY.

SEND US YOUR GAMES!



Have you written a games program that you'd like to see printed within C&VG? You have? Then fill in this form and rush it to us at Computer & Video Games, Program Listings, Priory Court, 30-32 Farringdon Lane, London EC1R 3AU. We pay £25 for each listing we print — so get your game to us today!

We'd also like you to send us a photograph of yourself — it doesn't matter how silly you look — so we can feature your smiling face alongside your listing and make you REALLY famous. Money and fame — what more could you ask for?

Please fill in the form below carefully so we know exactly what's going on in your game. And please include full instructions and program details along with your tape. Don't forget to keep a copy of your game for yourself. We're afraid that we can't return your tapes to you.

Talking of tapes — that's how we'd like you to send your games in to us. On tape or disc — with a listing if you have a printer.

Lots of our readers have been asking just what happens to their games when they arrive at C&VG. Well, we send them out to our team of hard working reviewers who let us know just what they think of your game. If they like it, your tape will go into our "to be

published" files. And that's where the waiting starts. We receive several hundred tapes each month — and loads of them are printable. So it can take quite a while to see your game in print. But we like to think that it's worth the wait!

And starting here and now you could become C&VG's Programmer of the Year — which we'll judge at Christmas time. So you could get an unexpected Christmas bonus. And don't worry those of you who already have had programs printed — you'll be eligible too!

Program name:

Machine make: Model:

Other models it should run on: Number of K needed to run it:

Other equipment (joysticks, Extended Basic, add ons, etc.) needed to run it:

Author's Christian name: Surname:

Address:

Tel: Date:

Type of game: (If original please say so)

Loading instructions

Game instructions: (If not included in the listing)

Office use only		Evaluator's comments	
Date received:		Good enough to publish	<input type="checkbox"/>
Acknowledgement sent: <input type="checkbox"/>		Needs some tidying up	<input type="checkbox"/>
Name of evaluator:		Not worth publishing	<input type="checkbox"/>
Date sent out:		Same game already published on this micro	<input type="checkbox"/>
Date due back:		Wouldn't load	<input type="checkbox"/>
Needs to be returned to author for alterations:	Date sent:		
Due to be published in issue of magazine:			



Each and every one of these listings has been tested by our reviewers and checked by the Bug Hunters at IDEAS CORP's Program Research HQ. But if you have problems with the listings in this Book of Games don't forget that the Bug Hunters can be contacted on 01-251-6222 every TUESDAY. Big Red, Otiss, X and B-Con will be waiting to deal with your calls in their own particular style. And remember IDEAS CORP are always on the look out for wonderful new listings for all the major micros. So if you have a game you reckon can pass the Bug Hunter's tough tests then why not send it to C+VG? You'll find the special software form inside this book. Plenty of credits go to the best games...

**WARNING
HAVE YOU SEEN
THIS MAN**



**HIGHLY DANGEROUS
WANTED FOR
TREACHERY**