ΚΑΤΩ ΑΠ'ΤΑ ΠΛΗΚΤΡΑ

SUBMARINE WAR

ΕΠΙΜΕΛΕΙΑ: Δ. ΑΣΗΜΑΚΟΠΟΥΛΟΣ

ΠΡΟΓΡΑΜΜΑΤΙΣΤΗΣ: ΓΙΩΡΓΟΣ ΡΟΥΣΟΣ

Amstrad CPC

Αυτό το μήνα, αγαπητοί αναγνώστες, θα ασχοληθούμε με κάτι διαφορετικό. Είμαστε σίγουροι ότι έχετε βαρεθεί να ακούτε όλο για εξωγήινους εισβολείς και άλλα τινά. Καιρός λοιπόν να υπερασπιστούμε τις θάλασσές μας από μερικά υποβρύχια που τριγυρνούν εδώ κι εκεί. Τα υποβρύχια αυτά, εκτός από το ότι κάνουν βόλτες για να ξεσκάσουν, μας ρίχνουν πού και πού και καμιά τορπίλη, όχι για κανένα σπουδαίο λόγο, αλλά για να σπάσει η μονοτονία. Εσείς βέβαια καλείστε να αντιμετωπίσετε τα υποβρύχια με το γνωστό τρόπο, ρίχνοντας βόμβες βυθού. Αν καταφέρετε να πετύχετε κάποιο υποβρύχιο, εκτός από το παράσημο που θα πάρετε, θα αμειφθείτε και με ανάλογα bonus points, που εξαρτώνται από το βάθος στο οποίο βρισκόταν το υποβρύχιο. Α, ξέχασα να σας πω, ότι υπάρχει και περιορισμένος χρόνος για την αποστολή σας. Επομένως, αν θέλετε να πάρετε παράσημο, εκτός από δεξιοτεχνία θα χρειαστείτε σίγουρα και γρηγοράδα. Το αντιτορπιλικό που σας διαθέτει το Πολεμικό Ναυτικό είναι εξοπλισμένο με τα τελειότερα όπλα. Ο χειρισμός του είναι εξίσου απλός. Αρκεί να έχετε κάπου εκεί κοντά κάποιο joystick. Αν δεν έχετε την τύχη να βρείτε ένα, τότε τα πλήκτρα κίνησης του δρομέα θα σας βγάλουν από πολλούς μπελάδες. Το πρόγραμμα πρέπει να τρέχει χωρίς μετατροπές σε όλους τους Amstrad CPC. Αλλά φτάνουν τα λόγια. Καιρός είναι να το δούμε από κοντά.

ΓΡΑΜΜΕΣ 10-180

Εδώ γίνεται ο αρχικός καθορισμός των μεταβλητών του προγράμματος και σχεδιάζονται τα U.D.G.s. Οι εντολές 130 και 140 συνθέτουν αρισμένα sprites. Αυτός ο τρόπος «ένωσης» των U.D.G.s. είναι πολύ χρήσιμος, γιατί αργότερα, όταν θα θελήσουμε να τους τυπώσουμε, θα καταφύγουμε σε μια απλή εντολή PRINT, ενώ, σε αντίθετη περίπτωση, θα αναγκαζόμασταν να τα τυπώσουμε ένα-ένα.

```
10 REM ************************
20 REM **                    **
30 REM **   GEORGE ROUSSOS   **
40 REM ** H OLIKH KATASTRVFH **
50 REM **AMSTRAD 464-664-6128**
60 REM **--------------------**
70 REM ************************
80 SYMBOL AFTER 128
90 FOR a=128 TO 148
100 READ b,c,d,e,f,g,h,i
110 SYMBOL a,b,c,d,e,f,g,h,i
120 NEXT
130 cloud$=CHR$(140)+CHR$(141)+CHR$(142)+CHR$(143)
140 clouds$=CHR$(144)+CHR$(145)+CHR$(146)+CHR$(147)
150 DATA 0,192,99,63,42,63,96,192,0,0,192,160,240,224,0,0,0,0,3,5,15,7,0,0,0,3,1
98,252,84,252,6,3,0,0,3,6,15,255,31,7
160 DATA 0,0,48,98,244,252,252,248,0,0,2,3,7,2,0,0,0,0,32,192,192,64,0,0,13,31,2
9,46,60,31,15,3,128,224,224,48,112,224,192,128
170 DATA 144,96,96,240,240,96,96,0,0,96,96,240,240,96,96,144,0,1,3,7,55,127,127,
255,199,239,233,191,255,255,255,255,0,248,248,247,23
9,255,255,255,0,0,0,128,192,0,128,192
180 DATA 255,255,127,31,30,0,0,0,255,255,255,255,255,206,0,0,255,255,255,255,247
,227,0,0,224,224,224,192,192,128,0,0,0,0,0,0,204,255
,255,255
```

ΚΑΤΩ ΑΠ'ΤΑ ΠΛΗΚΤΡΑ

ΓΡΑΜΜΕΣ 190-270

Λίγες οδηγίες δε βλάπτουν, δε συμφωνείτε; Αυτό σκέφτηκε και ο φίλος μας ο Γιώργος και συμπεριέλαβε μερικές μέσα στο πρόγραμμα. Δεν έχουμε τίποτα το ιδιαίτερο να πούμε γι΄ αυτές τις γραμμές, εκτός του ότι η γραμμή 270 θα μπορούσε να αντικαταστα θεί με ένα απλό CALL &BB18, ή ένα CALL &BB06.

```
190 MODE 1:BORDER 0:INK 0,0:INK 1,18:INK 2,6:INK 3,7,0:SPEED INK 5,5:BORDER 0:PA
PER 0
200 LOCATE 12,1:PEN 1:PRINT "OLIKH KATASTROFH"
210 MOVE 220,380:DRAW 380,380
220 PEN 2:LOCATE 3,3:PRINT "ESEIS EISTAI O ARXHGOS TOY SKAFOYS.H DOYLEIA SAS EIN
AI NA KATASTRECETE TOYS EXUROYS.":PRINT "":PEN 1
230 PRINT "MPVREIS NA KONTROLARHS TO SKAFOS ME TO <<JOYSTICK>> H ME TA KLHDIA PO
Y YPARXOYN STA DEJIA TOY PLHKTROLOGIOY."
240 PEN 2:PRINT "AKVMH YPARXEI KAI KAPOIA PROUESMIA XRONOY 3 LEPTVN"
250 LOCATE 13,21:PRINT "KALH TYXH....":PRINT:PEN 3
260 LOCATE 13,23:PRINT "PATA ENA KOYMPI"
270 A$=INKEY$:IF A$="" THEN 270
```

ΓΡΑΜΜΕΣ 280-480

Εδώ έχουμε τις αρχικές τοποθετήσεις των υποβρυχίων και του δικού μας σκάφους. Αρκετά έξυπνη είναι η χρήση των εντολών του τύπου PRINT SPACE$, γιατί μας γλυτώνουν από πολλά LOCATE. Επίσης τυπώνεται στην οθόνη το Score, καθώς και η ώρα που μας απομένει για να εκπληρώσουμε την αποστολή μας. Η εντολή SPEED INK της γραμμής 290 μεταβάλλει το χρόνο του flashing των χρωμάτων της οθόνης. Κάτι ακόμα που θα μπορούσαμε να αναφέρουμε, είναι ότι είναι καλύτερα να αντικατασταθούν οι πολλές εντολές INK της γραμμής 290 με ένα loop, όπως αυτό της γραμμής 370.

```
280 FOR a=1 TO 26:NEXT
290 INK 0,1:INK 1,0:INK 2,24:INK 3,11:INK 4,12:INK 5,26:INK 6,6:INK 7,6,24:SPEED
 INK 10,10:MODE 0
300 PAPER 1:LOCATE 1,1:PRINT SPACE$(20)
310 PEN 2:LOCATE 1,1:PRINT "SCORE:";score
320 PAPER 3:FOR a=2 TO 9:LOCATE 1,a:PRINT SPACE$(20):NEXT
330 PAPER 1:FOR a=23 TO 25:LOCATE 1,a:PRINT SPACE$(20);:NEXT
340 PLOT 23,40,5:DRAW 612,40:DRAW 612,10:DRAW 23,10:DRAW 23,40
350 PAPER 0
360 LOCATE 1,22
370 FOR a=1 TO 20:PRINT CHR$(148);:NEXT
380 PEN 5:PAPER 3
390 RESTORE 460
400 FOR a=1 TO 6
410 READ b,c
420 LOCATE b,c:PRINT CHR$(22)+CHR$(1);cloud$
430 LOCATE b,c+1:PRINT cloud$+CHR$(22)+CHR$(0)
440 NEXT
450 PAPER 1
460 DATA 3,3,7,5,10,2,6,4,15,2,13,6
470 PEN 2:LOCATE 2,24:PRINT "TIME:";:PAPER 6:PRINT "    ";:PAPER 2:PRINT SPACE$(1
0):PAPER 1
480 xship=9:sub1=10:sub2=18:sub3=5:sub4=17:clock=300:b=0:ybomb=9:acc=607:subomb=
0:xbom=INT(RND*19)+2:ybom=21:astic=0
```

ΓΡΑΜΜΕΣ 490-740

Αυτός είναι ο κύριος βρόχος του προγράμματος. Στις γραμμές 530-570 τυπώνονται το πλοίο μας και τα υποβρύχια, ενώ, αν είμαστε άτυχοι, η γραμμή 580 φροντίζει να εκτοξεύσει μια τορπίλη εναντίον μας, καλώντας την υπορουτίνα 1090. Οι γραμμές 600-650 φροντίζουν για την κίνηση των υποβρυχίων και ενεργούν ανάλογα, σε περίπτωση που κάποιο από αυτά ξέφυγε από τα όρια της οθόνης. Οι γραμμές 670-700 είναι υπεύθυνες για την ανάγνωση του joystick και των πλήκτρων. Η μεταβλητή b, που χρησιμοποιείται στη γραμμή 700, είναι μια σημαία που δείχνει αν έχουμε πατήσει SPACE ή FIRE, δηλαδή αν θέλουμε να ρίξουμε κάποια βόμβα βυθού. Τέλος, οι γραμμές 720 και 730 φροντίζουν ώστε το σκάφος μας να μένει μέσα στην οθόνη του Amstrad και να μην περιφέρεται από εδώ κι από `κει.

```
490 sub1=sub1+0.5:sub2=sub2-1:sub3=sub3+1.2:sub4=sub4-1.6:astic=astic+1
500 IF astic=5 THEN SOUND 1,50,10,5
510 IF astic=10 THEN SOUND 1,50,10,3:astic=0
520 GOSUB 970
530 LOCATE xship,9:PAPER 3:PEN 8:PRINT " ";CHR$(132);CHR$(133);" "
540 PAPER 0:PEN 4
550 LOCATE sub1,12:PRINT " ";CHR$(128);CHR$(129)
560 LOCATE sub2,15:PRINT CHR$(130);CHR$(131);" "
570 LOCATE sub3,18:PRINT "  ";CHR$(128);CHR$(129)
580 IF subomb=1 THEN GOSUB 1090
590 LOCATE sub4,21:PRINT CHR$(130);CHR$(131);"  "
600 IF sub1>16 THEN LOCATE sub1,12:PRINT "      ":sub1=1
610 IF sub2<2 THEN LOCATE sub2,15:PRINT "      ":sub2=18
620 IF subomb=1 THEN GOSUB 1090
630 IF sub3>16 THEN LOCATE sub3,18:PRINT "      ":sub3=1
640 IF sub4<2 THEN LOCATE sub4,21:PRINT "      ":sub4=18
650 IF INT(RND*10)+1=5 AND subomb=0 THEN subomb=1
660 IF subomb=1 THEN GOSUB 1090
670 a$=INKEY$
680 IF JOY(0)=4 OR INKEY(8)=0 THEN xship=xship-1
690 IF JOY(0)=8 OR INKEY(1)=0 THEN xship=xship+1
700 IF JOY(0)=16 AND b=0 OR a$=" " AND b=0 THEN x=xship+3:b=1
710 IF b=1 THEN GOSUB 750
720 IF xship=0 THEN xship=1
730 IF xship=18 THEN xship=17
740 GOTO 490
```

ΓΡΑΜΜΕΣ 750-820

Αυτή η υπορουτίνα αναλαμβάνει να σχεδιάσει τη βόμβα βυθού που ενδεχομένως έχουμε ρίξει και, επί πλέον, ελέγχει αν κτυπήθηκε κάποιο υποβρύχιο. Αν πράγματι χτυπήθηκε κάποιο, τότε καλείται η υπορουτίνα στη γραμμή 830. Η γραμμή 810 φροντίζει ώστε η βόμβα μας, αν δεν έχει πετύχει τίποτα, να εξαφανιστεί από την οθόνη. Αυτός ο τρόπος που χρησιμοποιεί ο Γιώργος για να παραστήσει τη βόμβα έχει ένα μειονέκτημα. Δεν είναι δυνατόν να έχουμε δύο βόμβες ταυτόχρονα στην οθόνη. Αυτό γίνεται γιατί χρησιμοποιείται μόνο μία μεταβλητή. Θα μπορούσαμε να έχουμε περισσότερες βόμβες, χρησιμοποιώντας ένα array.

```
750 ybomb=ybomb+1
760 PEN 4:LOCATE x,ybomb-1:PRINT " ":LOCATE x,ybomb:PRINT CHR$(138)
770 IF ybomb=12 AND x>=sub1+1 AND x<=sub1+2.5 THEN GOSUB 830
780 IF ybomb=15 AND x>=sub2 AND x<=sub2+1.8 THEN GOSUB 830
790 IF ybomb=18 AND x>=sub3+1 AND x<=sub3+2.5 THEN GOSUB 830
800 IF ybomb=21 AND x>=sub4 AND x<=sub4+2 THEN GOSUB 830
810 IF ybomb=21 AND x<>sub4 AND x<>sub4+2 THEN LOCATE x,ybomb:PRINT " ":b=0:ybom
b=9
820 RETURN
```

ΓΡΑΜΜΕΣ 830-960

Αυτή η υπορουτίνα αυξάνει το score μας ανάλογα με το ποιό υποβρύχιο είχαμε χτυπήσει. Επίσης, δημιουργεί και τον ήχο της έκρηξης (γραμμή 870). Εδώ φαίνεται και πόσους βαθμούς παίρνουμε, ανάλογα με το υποβρύχιο που χτυπάμε.

```
830 FOR a=1 TO 5
840 PEN 7:LOCATE x,ybomb:PRINT CHR$(134);CHR$(135)
850 IF a=1 OR a=3 THEN SOUND 3,0,15,6,0,0,2
860 PEN 7:LOCATE x,ybomb:PRINT CHR$(136);CHR$(137)
870 IF a=4 THEN SOUND 2,0,10,7,0,0,4
880 NEXT
890 LOCATE x,ybomb:PRINT "  ":
900 IF ybomb=12 THEN sub1=1:LOCATE 1,12:PRINT SPACE$(20);:score=score+1
910 IF ybomb=15 THEN sub2=16:LOCATE 1,15:PRINT SPACE$(20);:score=score+10
920 IF ybomb=18 THEN sub3=1:LOCATE 1,18:PRINT SPACE$(20);:score=score+30
930 IF ybomb=21 THEN sub4=16:LOCATE 1,21:PRINT SPACE$(20);:score=score+50
940 b=0:ybomb=9
950 PEN 2:PAPER 1:LOCATE 7,1:PRINT score:PAPER 0:PEN 4
960 RETURN
```

ΓΡΑΜΜΕΣ 970-1120

Μια ρουτίνα σχεδίασης και το τέλος του παιχνιδιού περιλαμβάνονται σε αυτή την υπορουτίνα. Σ' αυτό το σημείο του προγράμματος υπάρχει ένα πολύ σοβαρό λάθος. Αν προσέξετε, η ρουτίνα στη γραμμή 1020 καλείται με ένα GOTO που υπάρχει στη γραμμή 1000. Όμως, η ρουτίνα 970-1010 έχει κληθεί με ένα GOSUB. Άρα στο σωρό των GOSUBs παραμένει μία τιμή που δε χρησιμοποιείται, αφού αν θέλουμε να ξαναπαίξουμε το παιχνίδι ξαναγυρίζουμε στο πρόγραμμα με ένα GOTO. Αυτό το λάθος παρατηρείται συχνά και καλό θα ήταν να το αποφεύγετε, γιατί μετά από έναν ορισμένο αριθμό παιχνιδιών ο Amstrad θα βγάλει το μήνυμα Memory full αφού δε θα του έχει μείνει άλλος χώρος για να αποθηκεύσει άλλες τιμές για τα GOSUBs.

Τέλος στη γραμμή 1070 θα πρέπει να παραλειφθεί το RUN που υπάρχει στο τέλος, γιατί το πρόγραμμα δε θα τελειώνει ποτέ!

```
970 PLOT acc,32,3
980 DRAW acc,14
990 acc=acc-0.9
1000 IF acc<192 THEN 1020
1010 RETURN
1020 LOCATE 5,10:PEN 7:PAPER 0:PRINT "GAME IS OVER":FOR A=1 TO 1000:NEXT
1030 INK 8,24,6:LOCATE 3,20:PAPER 0:PEN 8:PRINT "AKOMA ENA PAIXNIDH  ( N / O )
"
1040 A$=LOWER$(INKEY$)
1050 IF A$="" THEN GOTO 1040
1060 IF A$="n" THEN 280
1070 IF A$="o" THEN INK 0,0:BORDER 0:PAPER 0:INK 1,18:PEN 1:MODE 1:RUN
1080 GOTO 1040
1090 LOCATE xbom,ybom:PRINT " ":ybom=ybom-1:LOCATE xbom,ybom:PRINT CHR$(139),
1100 IF ybom=9 AND xbom>=xship+1 AND xbom<=xship+2 THEN acc=191:GOSUB 1130:RETUR
N
1110 IF ybom=9 THEN LOCATE xbom,ybom:PAPER 3:PRINT " ":PAPER 0:xbom=INT(RND*19)+
2:ybom=21
1120 RETURN
```

ΓΡΑΜΜΕΣ 1130-1240

Εδώ έχουμε μερικές ακόμα ρουτίνες σχεδιασμού και τυπώματος χαρακτήρων στην οθόνη.

Συνοψίζοντας τώρα μπορούμε να πούμε ότι είναι ένα αρκετά καλό πρόγραμμα. Αν μάλιστα δεν είχε εκείνο το σοβαρό λάθος στη γραμμή 1000, το πρόγραμμα θα μπορούσε να ήταν και τέλειο. Φίλε Γιώργο, μπορείς να περάσεις από τα γραφεία μας για τα παραιτέρω!

```
1130 '
1140 FOR a=1 TO 10
1150 PEN 7:PAPER 3:LOCATE xship+1,9
1160 PRINT CHR$(134);CHR$(135)
1170 IF a=1 OR a=3 THEN SOUND 3,0,15,6,0,0,5
1180 LOCATE xship+1,9
1190 PRINT CHR$(136);CHR$(137)
1200 IF a=4 THEN SOUND 2,0,10,7,0,0,11
1210 NEXT
1220 LOCATE xship+1,9:PAPER 3:PRINT "  ":LOCATE xship+1,8:PRINT "  ":PAPER 0
1230 score=0
1240 PEN 4:ybom=22:RETURN
```