

ΤΟΜΟΣ 1 • ΤΕΥΧΟΣ 1 • 300 ΔΡΧ.  
ΟΚΤΩΒΡΙΟΣ • ΝΟΕΜΒΡΙΟΣ • ΔΕΚΕΜΒΡΙΟΣ 1983  
ΕΚΔΟΣΕΙΣ COMPUPRESS Ο.Ε.

# PIXEL

ΠΑΝΩ ΑΠΟ 50  
ΠΡΟΓΡΑΜΜΑΤΑ  
& Videοπαιχνίδια για micros

**PIXELMANIA ONE**  
Κερδίστε ένα  
Jupiter Ace

Jupiter ACE  
PERSONAL COMPUTER

ΜΙΑ ΕΙΔΙΚΗ ΕΚΔΟΣΗ ΤΟΥ 50<sup>ου</sup> ΠΥΡΡΕΓΓΙΟΛΟΥ

**144**  
ΣΕΛΙΔΕΣ

FLIGHT SIMULATION  
ΟΡΜΟΙ  
ΠΕΡΕΡΑΣΙΑ ΚΕΙΜΕΝΟΥ  
ΙΒΡΑΕΙΣ  
ΑΓΝΩΣΤΗ ΒΑΒΩΝ ΑΥΤΟΚΙΝΗΤΟΥ

BLITZ  
Η ΓΗ ΣΕ ΚΙΝΔΥΝΟ  
ΗΜΕΡΟΛΟΓΙΟ  
ΣΤΑΤΙΣΤΙΚΗ ΠΟΛΗΣΙΕΩΝ  
ΔΙΕΥΘΥΝΣΙΟΓΡΑΦΟΣ

ΑΡΧΙΣΤΟΛΗ ΣΤΟΝ ΠΛΑΝΗΤΗ ΔΕΛΤΑ  
LUNAR LANDING  
ΤΗΛΕΦΩΝΙΚΟΣ ΚΑΤΑΛΟΓΟΣ  
ΑΠΕΡΓΙΑ  
ΤΡΙΣΔΙΑΣΤΑΤΟΣ ΛΑΒΥΡΙΝΘΟΣ

ΡΩΤΗΖΗ ΑΥΤΟΚΤΟΝΙΑΣ  
ΝΑΥΜΑΧΙΑ  
ΤΟ ΡΑΛΛΥ ΤΗΣ ΚΑΤΑΣΤΡΟΦΗΣ  
ΤΟ ΜΕΓΑΛΟ ΦΑΓΟΠΟΙ  
ΜΕΤΕΩΡΙΤΕΣ

BREAKOUT  
STAR TREK  
ΠΟΛΕΜΟΣ ΤΩΝ ΑΣΤΡΩΝ  
ΜΟΝΟΜΑΧΙΑ ΤΩΝ ΙΠΠΟΤΩΝ  
TRON

# ΒΑΣΙΣΤΕΙΤΕ ΣΤΟΝ «ΠΡΟΣΩΠΙΚΟ» ΣΑΣ ΥΠΟΛΟΓΙΣΤΗ ΤΟΝ IBM PERSONAL COMPUTER



ΚΑΙ ΣΤΗΝ ABC τον εξουσιοδοτημένο  
αντιπρόσωπο της IBM για:

- Ολοκληρωμένες λύσεις
- Εγγυημένη τεχνική υποστήριξη
- Υπεύθυνη εκπαίδευση



## ABC

Advanced Business Computing & Professional Services ΕΠΕ

Λεωφ. Συγγρού 137, Ν. Σμύρνη - Αθήνα, Τηλ.: 932.0590 - 9598.555

## ΓΡΑΜΜΑ ΑΠΟ ΤΟΝ ΕΚΔΟΤΗ

---

Αγαπητοί αναγνώστες,

Όλοι μας γνωρίζουμε ότι το software αποτελεί το "φάντασμα στη μηχανή" κάθε μικροϋπολογιστή και ότι το hardware μόνο του είναι τόσο άχρηστο όσο ένα αυτοκίνητο χωρίς καύσιμα.

Το περιοδικό που κρατάτε στα χέρια σας λοιπόν είναι μια προσπάθεια του "COMPUTER ΓΙΑ ΟΛΟΥΣ" να καλύψει τον τομέα των home/personal computers από πλευράς προγραμμάτων προσφέροντάς σας έτοιμα listings που μπορείτε να πληκτρολογήσετε στο μικροϋπολογιστή σας για να δημιουργήσετε μια "βιβλιοθήκη προγραμμάτων" ή και να μελετήσετε για να ανακαλύψετε κάποιες καινούργιες προγραμματιστικές τεχνικές.

Η ανάγκη για τη δημιουργία ενός τέτοιου έντυπου εντοπίστηκε στο ερωτηματολόγιο του 5ου τεύχους του "COMPUTER ΓΙΑ ΟΛΟΥΣ" όπου διαπιστώθηκε ότι ένα μεγάλο ποσοστό αναγνωστών ζητούσαν περισσότερα έτοιμα προγράμματα. Αποτέλεσμα αυτού του feedback είναι η εμφάνιση του PIXEL, του πρώτου ελληνικού περιοδικού που ασχολείται αποκλειστικά με listings προγραμμάτων για τα δημοφιλέστερα micros της ελληνικής αγοράς. Το καινούργιο αυτό έντυπο θα είναι τριμηνιαίο, με προοπτική να πυκνώσουν στο μέλλον οι εκδόσεις του.

Το πρώτο λοιπόν τεύχος του PIXEL περιλαμβάνει πολλά και εξαιρετικά ενδιαφέροντα προγράμματα για τους παρακάτω μικροϋπολογιστές (η σειρά παρουσίας των μοντέλων έγινε με βάση την τιμή):

- \* SINCLAIR ZX-81
- \* JUPITER ACE
- \* SINCLAIR ZX-SPECTRUM
- \* ORIC 1
- \* TEXAS INSTRUMENTS TI 99/4A
- \* TRS-80 COLOR
- \* COMMODORE VIC-20
- \* NEWBRAIN
- \* BBC MODEL B
- \* EPSON HX-20
- \* APPLE IIe
- \* OSBORNE 1

Όλα τα listings έχουν ελεγχθεί από συνεργάτες του περιοδικού έτσι ώστε να είναι "bug-free" και συνεπώς μπορείτε άφοβα να τα πληκτρολογήσετε. Κλείνοντας αυτό το σημείωμα θα θέλαμε να υπογραμμίσουμε ότι καταβλήθηκε κάθε προσπάθεια ώστε η επιλογή των προγραμμάτων που δημοσιεύονται στο πρώτο τεύχος του "PIXEL" να είναι ότι καλύτερο μπορούσε να γίνει και πιστεύουμε ότι πραγματικά το τεύχος αυτό αποτελεί μια προσφορά στους αναγνώστες μας.

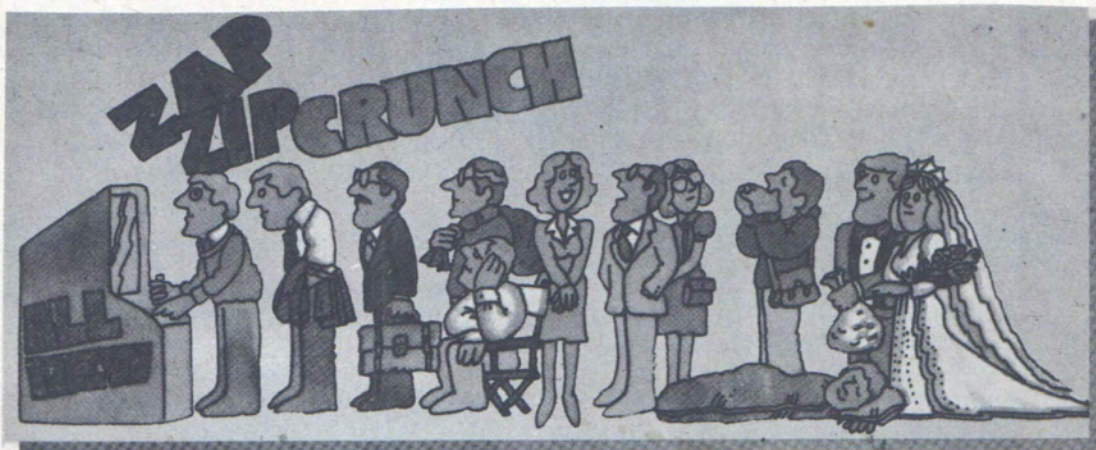
N. ΜΑΝΟΥΣΟΣ

Εκδότης - Διευθυντής

# PIXEL

ΟΚΤΩΒΡΙΟΣ · ΝΟΕΜΒΡΙΟΣ · ΔΕΚΕΜΒΡΙΟΣ 1983

ΤΕΥΧΟΣ Νο.1



ΙΔΙΟΚΤΗΣΙΑ : COMPUPRESS Ο.Ε.  
ΕΚΔΟΤΗΣ Δ/ΝΤΗΣ : Νίκος Μανούσος  
ΑΡΧΙΣΥΝΤΑΚΤΗΣ : Χάρης Σώκος  
ΔΗΜΟΣΙΕΣ ΣΧΕΣΕΙΣ / ΔΙΑΦΗΜΙΣΗ : Άγαπη Λαλιώτη  
ΚΑΛΛΙΤΕΧΝΙΚΗ ΕΠΙΜΕΛΕΙΑ : Εύη Κώστη  
ΓΡΑΜΜΑΤΕΙΑ : Δήμητρα Παπαγιαννακοπούλου  
                  Λίνα Ρόμπολα  
ΓΡΑΦΕΙΑ : Βερανζέρου 15, 106 77 - ΑΘΗΝΑ  
ΩΡΕΣ ΓΡΑΦΕΙΟΥ : 8<sup>30</sup> — 16<sup>00</sup> καθημερινώς  
ΤΗΛΕΦΩΝΟ : 3644686 — 3601761

“PIXEL” Τριμηνιαίο Περιοδικό Προγραμμάτων για Μικροϋπολογιστές • ΥΠΕΥΘΥΝΟΣ ΣΥΜΦΩΝΑ ΜΕ ΤΟ ΝΟΜΟ: Νίκος Μανούσος, 25<sup>12</sup> Μαρτίου 27 — Χολαργός.  
ΦΩΤΟΣΤΟΙΧΕΙΟΘΕΣΙΑ: ΚΑΤΑΓΡΑΜΜΑ • ΑΝΑΠΑΡΑΓΩΓΗ ΕΞΟΦΥΛΛΟΥ — ΕΣΩΤΕΡΙΚΩΝ / ΔΙΑΧΩΡΙΣΜΟΙ ΤΕΤΡΑΧΡΩΜΙΩΝ: Πέτρος Καραμανέας.  
ΕΚΤΥΠΩΣΗ: Λιθογραφία Αθηνών • ΒΙΒΛΙΟΔΕΣΙΑ: Αφοι Βαλαβάνη ΕΠΙΤΑΓΕΣ: Περιοδικό PIXEL, Βερανζέρου 15, 106 77 — ΑΘΗΝΑ.

ΕΚΔΟΣΕΙΣ COMPUPRESS Ο.Ε.

## ΠΕΡΙΕΧΟΜΕΝΑ

<b>ΣΗΜΕΙΩΜΑ</b> .....	3
• <b>ZX81</b> .....	6
FORMCALC, ΕΠΕΞΕΡΓΑΣΙΑ ΚΕΙΜΕΝΟΥ, PR. NAME, MUSIC MACHINE, 3D-GRAPH	
• <b>JUPITER ACE</b> .....	19
ΕΛΛΗΝΙΚΟ ΑΛΦΑΒΗΤΟ, ΓΛΩΣΣΑ ΜΗΧΑΝΗΣ, ΣΑΡΑΝΤΑΠΟΔΑΡΟΥΣΑ	
• <b>SPECTRUM</b> .....	25
FLIGHT SIMULATION, ΜΟΝΟΜΑΧΙΑ ΤΩΝ ΙΠΠΟΤΩΝ, ΔΙΑΓΝΩΣΗ ΒΛΑΒΩΝ ΑΥΤΟΚΙΝΗΤΟΥ, ΤΡΟΝ ΖΟΓΚΛΕΡ, Ο ΠΟΛΕΜΟΣ ΤΩΝ ΑΣΤΡΩΝ, Η ΑΝΑΤΙΝΑΞΗ ΤΗΣ ΒΟΥΛΗΣ.	
• <b>ORIC</b> .....	44
ΕΠΙΚΙΝΔΥΝΗ ΠΡΟΣΓΕΙΩΣΗ, ΚΑΛΕΙΔΟΣΚΟΠΙΟ, Ο ΚΗΠΟΣ ΤΩΝ ΕΚΠΛΗΞΕΩΝ, BREAKOUT	
• <b>TI 99/4A</b> .....	55
ΠΡΟΣΕΔΑΦΙΣΗ ΣΤΟΝ ΤΙΤΑΝΑ, ΤΕΧ THELLO, Η ΓΗ ΣΕ ΚΙΝΔΥΝΟ	
• <b>TRS-80 color</b> .....	65
ΤΡΕΛΛΟΣ ΛΑΒΥΡΙΝΘΟΣ, GRAPHICS, ΠΟΛΕΜΟΣ ΣΤΗ ΘΑΛΑΣΣΑ, ΠΕΝΤΕ ΚΡΙΚΟΙ-ΕΝΑ ΤΑΛΛΗΡΟ, Ο ΚΥΒΟΣ ΤΟΥ RUBIC.	
• <b>VIC-20</b> .....	77
ALPHA ATTACK, ΤΟ ΦΙΔΙ, ΑΠΕΡΓΙΑ, 3D-ΛΑΒΥΡΙΝΘΟΣ, ΕΙΣΒΟΛΕΙΣ	
• <b>NEWBRAIN</b> .....	87
ΓΡΑΦΙΚΕΣ ΠΑΡΑΣΤΑΣΕΙΣ, ΣΤΟΙΧΕΙΑ ΣΥΜΜΕΤΡΙΚΩΝ ΔΙΑΤΟΜΩΝ	
• <b>BBC model B</b> .....	94
ΤΑΞΙΔΙ ΣΤ' ΑΣΤΕΡΙΑ, ΠΤΗΣΗ ΑΥΤΟΚΤΟΝΙΑΣ, ΤΟ ΡΑΛΛΥ ΤΗΣ ΚΑΤΑΣΤΡΟΦΗΣ, ΤΟ ΜΕΓΑΛΟ ΦΑΓΟΠΟΤΙ, ΤΟ ΚΟΨΙΜΟ ΤΗΣ ΧΛΟΗΣ, ΑΠΟΣΤΟΛΗ ΣΤΟΝ ΠΛΑΝΗΤΗ ΔΕΛΤΑ, ΕΠΙΚΙΝΔΥΝΟΙ ΜΕΤΕΩΡΙΤΕΣ	
• <b>EPSON HX-20</b> .....	106
Η BASIC ΤΟΥ EPSON ΚΑΙ Η MBASIC, ΥΠΕΝΘΥΜΙΣΤΗΣ, ΤΗΛΕΦΩΝΙΚΟΣ ΚΑΤΑΛΟΓΟΣ, ΕΠΙΤΟΚΙΑ/ΗΜΕΡΟΜΗΝΙΣ, ΤΟ ΠΛΟΙΟ.	
• <b>APPLE IIe</b> .....	113
MODIFIABLE DATABASE, APPLE PROMS, ANIMALS, ADDRESS, MUSIC SYNTHESIZER.	
• <b>OSBORNE 1</b> .....	123
ΔΙΑΧΕΙΡΙΣΗ ΣΕΙΡΙΑΚΩΝ ΑΡΧΕΙΩΝ ΣΤΗΝ MBASIC, ΔΥΟ ΠΡΟΓΡΑΜΜΑΤΑ, ΓΡΑΦΙΚΗ ΑΠΕΙΚΟΝΙΣΗ ΠΩΛΗΣΕΩΝ, ΕΛΕΓΧΟΣ ΠΡΟΜΗΘΕΙΩΝ	
• <b>PIXELMANIA 1</b> .....	136
ΚΕΡΔΙΣΤΕ ΕΝΑ JUPITER ACE	



Προγράμματα για τόν:

---

ZX-81



---

ΕΠΙΜΕΛΕΙΑ:  
Ματθαῖος Μηνδρινός

---



## FORMCALC

Σίγουρα όλοι οι αναγνώστες του περιοδικού έχουν κάποια ιδέα για τα προγράμματα τύπου VisiCalc (έχουν γίνει τόσες παρουσιάσεις σε προηγούμενα τεύχη του "COMPUTER ΓΙΑ ΟΛΟΥΣ"). Το FORMCALC είναι μια σμίκρυνση των προγραμμάτων αυτών στα επίπεδα του ZX81. Για όσους δεν γνωρίζουν λέμε ότι τα προγράμματα αυτά

είναι καθαρά επαγγελματικά και χρησιμοποιούν σε διάφορες εμπορικές βιομηχανικές και επιστημονικές εφαρμογές. Στην προκειμένη περίπτωση το FORMCALC μπορεί να μας βοηθήσει να δούμε τις μεταβολές που θα έχουμε στις εξαρτημένες μας μεταβλητές όταν μεταβάλλονται οι τιμές των διαφόρων ανεξάρτητων μεταβλητών μας. Τρέχοντας καταρχάς το πρόγραμμα θα φανεί το

κυρίως του μενού όπως το βλέπετε στο σχήμα, που σας ζητά να δώσετε το νούμερο (1-3) κάποιος εκ των τριών προσβάσεων. Όταν θέλετε να φτιάξετε ένα καινούργιο φύλλο εργασίας και ταυτόχρονα να "καθαρίζετε" όλα τα προηγούμενα δεδομένα δώστε το (1). Όταν θέλετε να χρησιμοποιήσετε τα δεδομένα που είχατε χρησιμοποιήσει την τελευταία φορά που κάνατε

SAVE το πρόγραμμα, δώστε το νούμερο (2), ενώ όταν θελήσετε να σβήσετε όλα τα προηγούμενα δεδομένα αλλά η FORMULA να παραμείνει η ίδια δώστε στο μενού το νούμερο (3). Όπως είναι λογικό θα ξεκινήσετε δίνοντας το νούμερο (1), οπότε θα εμφανιστεί στην οθόνη σας η ερώτηση "ENTER NO - OF ROWS REQUIRED" που σας ζητά τον αριθμό των γραμμών που θα χρησιμοποιήσετε στο φύλλο εργασίας σας (με άλλα λόγια τον αριθμό των τιμών που θα δώσετε σε κάθε μια απ' τις μεταβλητές σας).

Αφού δώσετε τον αριθμό των σειρών θα εμφανιστεί η ερώτηση "ENTER NO - OF COLUMNS REQUIRED" που σας ζητά τον αριθμό των στηλών του φύλλου εργασίας σας (δηλαδή τον αριθμό των μεταβλητών σας, εξαρτημένων/ανεξάρτητων). Ο αριθμός των στηλών βρίσκεται πάντοτε σε μια σχέση με τον αριθμό των γραμμών που δώσατε προηγουμένως και πάντοτε είναι μέσα στα όρια από 1 ως 100. Η επόμενη ερώτηση που θα εμφανιστεί είναι η "HOW MANY COLUMNS TO BE VISIBLE ON SCREEN" που σας ζητά τον αριθμό των στηλών που θέλετε να εμφανίζονται στην οθόνη. Η τιμή αυτή όπως είναι φυσικό το πολύ να είναι ίση με τον αριθμό που δώσατε στην δεύτερη ερώτηση. Εφόσον δοθεί και η τιμή αυτή τότε θα εμφανιστεί στην οθόνη σας το φύλλο εργασίας (σχήμα 2) δίνοντας στο πάνω δεξιό σημείο του σαν πληροφορίες τον αριθμό των γραμμών (R) και των στηλών (C) του φύλλου εργασίας (που είχατε προηγουμένως δώσει) καθώς και τα K της μνήμης που έχει καταλάβει το πρόγραμμα. Στο πάνω αριστερό σημείο της οθόνης υπάρχει η ένδειξη "COMMAND?" που σας λέει ότι το πρόγραμμα

περιμένει κάποια εντολή από σας. Ξαν πρώτη εντολή δώστε την H (Heading) για να δώσετε κάποια ονομασία στις στήλες σας (που μην ξεχνάτε ότι είναι οι μεταβλητές σας) καθώς και για να μπει μια επικεφαλίδα σ' αυτές. Αφού δοθεί η εντολή H και πατήσετε N/L το πρόγραμμα αυτόματα θα σας ζητήσει το νούμερο της στήλης που θα μπει η επικεφαλίδα. Αφού η παραπάνω εργασία γίνει για όλες τις στήλες ερχόμαστε στο επόμενο στάδιο, που είναι το πέρασμα των τιμών σ' όλες τις ανεξάρτητες μεταβλητές μας. Για το σκοπό αυτό χρησιμοποιείτε την εντολή C.

Το επόμενο στάδιο είναι το πέρασμα όλων των δεδομένων της στήλης (αριθμός δεδομένων = αριθμός γραμμών). Με ένα αστεράκι το πρόγραμμα μας δείχνει τη γραμμή στην οποία θα μπουں κάθε φορά τα δεδομένα μας. Αφού περαστούν όλα τα δεδομένα της στήλης το πρόγραμμα επιστρέφει σε επίπεδο COMMAND περιμένοντας από μας κάποια εντολή. Αν θέλουμε να χρησιμοποιήσουμε περισσότερες από μια ανεξάρτητες μεταβλητές πρέπει να επαναλάβουμε αντίστοιχα τόσες φορές την παραπάνω εργασία (πέρασμα δεδομένων). Και ήρθε λοιπόν η στιγμή να πάρουμε τα ποθούμενα αποτελέσματα από το πρόγραμμά μας. Για το σκοπό αυτό πρέπει να δώσουμε την εντολή RF για να εισάγουμε τη φόρμουλα. Στη συνέχεια πρέπει να δώσουμε τον αριθμό της στήλης που θα χρησιμοποιηθεί σαν εξαρτημένη μεταβλητή και κατόπιν με την ερώτηση "ENTER FORMULA" το πρόγραμμα μας ζητά να του δώσουμε τη φόρμουλα, δηλαδή

τη σχέση μεταξύ των ανεξάρτητων μεταβλητών και της εξαρτημένης που θα είναι π.χ. της μορφής  $(K1+5)*(K2+K3-8)$ . Μέσα σε ελάχιστο χρόνο το πρόγραμμα θα μας δώσει τις τιμές της εξαρτημένης μεταβλητής μας και θα δούμε πως αυτές επηρεάζονται καθώς μεταβάλλονται οι τιμές των ανεξάρτητων μεταβλητών μας. Αν η φόρμουλα είχε μια σταθερή τιμή π.χ. (10) τότε όλες οι γραμμές της στήλης που δώσατε μετά την εντολή RF θα πάρουν την τιμή αυτή. Αυτός είναι και ένας έξυπνος τρόπος να περάσουμε σε μια στήλη εύκολα τα δεδομένα μας αν είναι όλα τα ίδια και να μην χρειάζεται να τα πληκτρολογήσουμε ένα-ένα με την εντολή (C). Ένας άλλος εύκολος τρόπος για να περάσουμε τα δεδομένα μας (αν αυτά ακολουθούν κάποια αριθμητική πρόοδο) είναι να χρησιμοποιήσουμε στην φόρμουλα τη μεταβλητή (Προσοχή!! όχι εντολή) R (ROW). Έτσι αν θέλετε στις σειρές (π.χ. από 1 ως 4) κάποιας στήλης να δώσετε τις τιμές (5,6,7,8) αντίστοιχα, μπορείτε να χρησιμοποιήσετε τη FORMULA R+4, οπότε στη 1 γραμμή της στήλης θα πάει το νούμερο 1+4=5, στην δεύτερη (2) το νούμερο 2+4=6 κ.ο.κ. Αν θέλετε το άθροισμα των τιμών μιας στήλης τότε απλώς δώστε την εντολή S και κατόπιν το νούμερο της στήλης οπότε θα εμφανισθεί η τιμή του αθροίσματος. Αν θέλετε τα δεδομένα κάποιας στήλης να ταξινομηθούν κατά αύξουσα σειρά τότε μπορείτε να χρησιμοποιήσετε την εντολή SORT (και κατόπιν το νούμερο της στήλης, καθώς και το νούμερο της αρχικής και της τελικής γραμ-

μής που θέλετε να γίνει η ταξινόμηση). Αν θέλετε το φύλλο εργασίας να μετατοπισθεί στην οθόνη μια στήλη αριστερά και στο δεξιό σημείο να εμφανιστεί η επόμενη στήλη δώστε την εντολή 5, ενώ για να τοποθετηθεί μια στήλη δεξιά δώστε την εντολή 8. Αν θέλετε το φύλλο εργασίας σας να τοποθετηθεί από μια συγκεκριμένη στήλη στην οθόνη, τότε μπορείτε να χρησιμοποιήσετε τις εντολές 5N, 8N. Αυτές λειτουργούν όπως οι εντολές 5,8 μόνο που εδώ σας ζητάει και η αρχική στήλη που θα τοποθετηθεί στην οθόνη. Αν χρησιμοποιείτε πάνω από 17 γραμμές (που χωράει η οθόνη) τότε για να δείτε τις τιμές κάτω από την 17η γραμμή χρησιμοποιείτε την εντολή 7 που κάνει scrolling μια-μια γραμμή προς τα πάνω. Σε περίπτωση που θέλετε να δείτε τις τελευταίες 17 γραμμές δώστε την εντολή B (BOTTOM), ενώ για να επιστρέψετε στις 17 αρχικές γραμμές, χρησιμοποιείτε την εντολή T (TOP).

Ήρθε λοιπόν η ώρα να δούμε με ποιόν τρόπο μπορούμε να αλλάξουμε μερικές τιμές κάποιας στήλης χωρίς να είμαστε αναγκασμένοι να ξαναπεράσουμε τα νούμερα όλα απ' την αρχή. Για το σκοπό αυτό, υπάρχει η εντολή CH (CHANGE) η οποία μας ζητά τις συντεταγμένες της τιμής που θέλουμε να αλλάξουμε (αριθμός

στήλης, γραμμής) και κατόπιν τη νέα τιμή. Αν θέλετε τώρα να δείτε πως έχουν μεταβληθεί οι τιμές τις εξαρτημένες μας μεταβλητές εξαιτίας αυτών των αλλαγών χωρίς να χρειάζεται να ξαναδώσετε την προηγούμενη FORMULA δώστε την εντολή RF. Για να αλλάξετε κατά την διάρκεια που τρέχετε το πρόγραμμα τον αριθμό των στηλών που εμφανίζονται στην οθόνη σας χωρίς να καταστρέψετε το φύλλο εργασίας υπάρχει η εντολή CC (CHANGE COLUMN). Αν θέλετε να βθήσετε το φύλλο εργασίας χωρίς να καταστρέψετε την προηγούμενη FORMULA, υπάρχει για σας μια ακόμα εντολή η CL (CLEAR). Για να καθαριστούν όχι μόνο τα δεδομένα σας αλλά και η FORMULA, δώστε την εντολή SC (που στην πραγματικότητα τρέχει το πρόγραμμα από την αρχή φέρνοντας σας στο μενού). Αν θέλετε να κάνετε saving το πρόγραμμα καθώς και την FORMULA και τα δεδομένα σας, δώστε απλώς την εντολή SAVE, ενώ αν θέλετε να σώσετε το πρόγραμμα δώστε πρώτα την εντολή SC και κατόπιν την εντολή SAVE. Για να σταματήσει τέλος το πρόγραμμα υπάρχει η εντολή O.

Αυτές λοιπόν είναι όλες οι command inputs. Μπορεί να σας φαίνονται πολλές, αξίζει όμως τον κόπο να τις μάθετε για να μπορείτε να χειριστείτε ένα τέτοιο πρόγραμμα.

#### FORMCALC

- DO YOU WANT 1 NEW WORKSHEET
- 2 STORED DATA
- 3 STORED FORMULA





Cat Computers

Cat Computers

Cat Computers

Cat Computers

Cat Computers

Cat Computers

Cat Computers

Cat Computers

προγράμματα γενικά και εφαρμογών  
πολυτικού μηχανικού, αλλά και  
πολλές κασέτες με παιχνίδια,  
βιβλία, περιοδικά, περιφερειακά  
και όλα τα πετυχημένα και φθηνά  
home computers

Cat Computers

Αθήνα-Ιπποκράτους 57-τηλ. 3643044

COMMAND? 25R 10C 6.61K  
 ===K1===K2===K3===K4===K5===

101  
102  
103  
104  
105  
106  
107  
108  
109  
110  
111  
112  
113  
114  
115  
116  
117

```

1 FAST
4 REM " FORMCALC "
5 CLS
6 PRINT AT 5,10;"FORMCALC";AT
13,0;"DO YOU WANT 1 NEW WORKSHE
ET";AT 15,12;"2 STORED DATA";AT
17,12;"3 STORED FORMULA"
14 INPUT X$
15 CLS
16 IF X$="2" THEN GOTO 1315
17 IF X$="3" THEN GOTO 2000
18 PRINT AT 0,0;"ENTER NO OF R
OWS REQUIRED"
20 INPUT N1
24 LET N=N1+1
28 PRINT AT 0,0;"ENTER NUMBER
OF COLUMNS REQUIRED"
30 INPUT M
31 PRINT AT 0,0;"HOW MANY COLU
MNS TO BE VISIBLE ON SCREEN"
32 INPUT C1
35 DIM A$(M,50)
37 DIM H$(M,6)
40 DIM Q(N,M)
50 DIM C(M)
55 LET I$=""
60 LET T=0
65 FOR U=1 TO C1
66 LET C(U)=(U*INT (30/C1)-INT
(21/C1)+1)
67 NEXT U
70 IF I$="CC" THEN GOTO 1315
74 LET J=0
76 LET K=0
78 LET S=0
80 IF I$="T" THEN GOTO 1315
100 LET L=N1
105 LET S=0
106 IF L>17 THEN LET L=17
107 PRINT AT 3,0;
110 FOR X=1 TO L
120 LET S=S+1
130 PRINT S
140 NEXT X
145 PRINT AT 0,0;"
150 PRINT AT 0,0;"COMMAND?
";N1;"R";";M;"C";";INT
((PEEK 16386+256*PEEK 16387-PEE
K 16412-256*PEEK 16413)/10+.5)/1
00;"K"
151 PRINT "=====
=====
152 FOR C=1 TO C1
154 PRINT AT 1,C*INT (30/C1)-IN
T (12/C1);"K";C-T
155 NEXT C
157 IF I$="CL" THEN GOSUB 430
160 INPUT I$
165 PRINT AT 0,0;"
170 IF I$="T" AND N1<18 OR I$="
B" AND N1<18 THEN GOTO 150
180 IF I$="T" THEN GOTO 74

```

```

190 IF I$="5" OR I$="8" OR I$="
B" THEN GOTO 1300
200 IF I$="8N" OR I$="5N" THEN
GOTO 1500
210 IF I$="7" THEN GOTO 1930
220 IF I$="SORT" THEN GOTO 3000
230 IF I$="RF" THEN GOTO 1600
240 IF I$="RR" THEN GOTO 1700
250 IF I$="S" THEN GOTO 700
260 IF I$="C" THEN GOTO 300
270 IF I$="H" THEN GOTO 400
280 IF I$="CC" THEN GOTO 800
285 IF I$="CH" THEN GOTO 1100
285 IF I$="CL" THEN GOTO 2000
290 IF I$="SC" THEN GOTO 5
295 IF I$="O" THEN STOP
298 IF I$="SAVE" THEN GOTO 3500
299 IF I$(">")="S" THEN GOTO 150
300 REM ENTER INDIVIDUAL VALUES
305 PRINT AT 0,0;"ENTER COLUMN
NO"
310 INPUT C
312 PRINT AT 0,0;"ENTER VALUES
"
315 LET U=C+T
320 FOR R=1 TO N1
322 LET R1=R-K
325 LET A$(C)="Q(R,C)"
327 IF R1<1 THEN GOTO 340
330 PRINT AT R1+2,C(U);"*"
340 INPUT Q(R,C)
345 IF R1<1 THEN GOTO 360
350 GOSUB 2300
352 IF R=N1 THEN GOTO 150
355 IF R1>16 THEN GOSUB 1900
360 NEXT R
370 GOTO 150
400 REM COLUMN HEADINGS
402 PRINT AT 0,0;"COLUMN NUMBER
"
405 INPUT C
407 LET U=C+T
410 PRINT AT 0,0;"COLUMN HEADIN
G"
415 INPUT H$(C)
417 IF U<1 OR U>C1 THEN GOTO 42
2
420 GOSUB 2350
422 PRINT AT 0,0;"COMMAND ?
"
425 GOTO 160
430 PRINT AT 2,0;"
"
435 FOR C=ABS T+1 TO ABS T+C1
440 IF C>M THEN RETURN
445 IF H$(C,1 TO 2)=" " THEN G
OTO 455
447 LET U=C+T
450 GOSUB 2350
455 NEXT C
460 RETURN
700 REM SUM VALUE OF A COLUMN
705 PRINT AT 0,0;"ENTER COLUMN
NO TO BE SUMMED"
710 INPUT C
712 LET U=C+T
715 LET Q(N,C)=0
730 FOR R=1 TO N1
740 LET Q(N,C)=Q(N,C)+Q(R,C)
750 NEXT R
755 IF U<1 OR U>C1 THEN GOTO 77
0
760 GOSUB 2400
770 IF I$="RR" THEN GOTO 1765
790 GOTO 150
800 REM CHANGE COLUMN SPACING
810 PRINT AT 0,0;"HOW MANY COLU
MNS TO BE VISIBLE"
820 INPUT C1
830 GOTO 65
900 REM PROGRESSIVE SUM
910 LET A$(C)="Q(R,C-1)+(R<>1)*
Q(R-1*(R<>1),C)"
920 GOTO 1635
1100 REM CHANGE A VALUE
1102 LET X$="Y"
1105 PRINT AT 0,0;"ENTER COLUMN
NUMBER"
1110 INPUT C
1112 IF A$(C,1 TO 6)(">")="Q(R,C)" A
ND A$(C,1 TO 2)(">") THEN PRINT
AT 0,0;"RF COLUMN STILL CHANGE(
Y/N)"
1114 IF A$(C,1 TO 2)(">") " AND A
$(C,1 TO 6)(">")="Q(R,C)" THEN INPUT
X$

```

```

1116 IF X$((">")) THEN GOTO 150
1118 PRINT AT 0,0;"ENTER ROW NO
AND NEW NUMBER
1120 INPUT R
1125 LET R1=R-K
1130 LET U=C+T
1140 INPUT Q(R,C)
1150 LET A$(C)="Q(R,C)"
1160 IF U<1 OR U>C1 OR R1<1 OR R
1>17 THEN GOTO 1180
1170 GOSUB 2300
1180 IF Q(N,C)(<>) THEN GOTO 715
1190 GOTO 150
1300 REM LEFT/RIGHT SHIFT
1305 IF I$="5" THEN LET T=T-1
1310 IF I$="8" THEN LET T=T+1
1312 IF I$="B" THEN LET K=N1-17
1313 IF I$="B" THEN LET S=N1
1315 CLS
1320 FOR C=ABS T+1 TO ABS T+C1
1322 IF C>M THEN GOTO 1367
1325 LET U=C+T
1330 IF H$(C,1 TO 2)(">") THEN
GOSUB 2350
1335 IF A$(C,1 TO 2)=" " THEN G
OTO 1365
1340 FOR R=ABS K+1 TO ABS K+L
1342 LET R1=R-K
1345 GOSUB 2300
1350 NEXT R
1355 IF Q(N,C)=0 THEN GOTO 1365
1360 GOSUB 2400
1365 NEXT C
1367 IF I$="T" THEN GOTO 105
1370 LET S=S-L
1380 GOTO 107
1500 REM COLUMN AT FRONT
1505 PRINT AT 0,0;"ENTER COLUMN
NO TO BE AT FRONT"
1510 INPUT X
1520 LET T=1-X
1530 GOTO 1315
1600 REM FORMULA ENTRY
1605 PRINT AT 0,0;"ENTER COLUMN
NUMBER"
1610 INPUT C
1617 PRINT AT 0,0;"ENTER FORMULA

1620 INPUT A$(C)
1630 GOSUB 1800
1635 LET U=C+T
1640 FOR R=1 TO N1
1645 LET R1=R-K
1650 LET Q(R,C)=VAL A$(C)
1660 IF U<1 OR U>C1 OR R1<1 OR R
1>17 THEN GOTO 1680
1670 GOSUB 2300
1680 NEXT R
1685 IF A$(C,1)="R" AND A$(C,2)="
" THEN LET A$(C)="Q(R,C)"
1690 IF Q(N,C)(<>) THEN GOTO 715
1695 GOTO 150
1700 REM RE/CALCULATION
1702 FOR C=1 TO M
1705 IF A$(C,1 TO 6)="Q(R,C)" OR
A$(C,1 TO 2)=" " THEN GOTO 176
5
1710 LET U=C+T
1730 FOR R=1 TO N1
1732 LET R1=R-K
1735 LET X=Q(R,C)
1740 LET Q(R,C)=VAL A$(C)
1745 IF U<1 OR U>C1 OR R1<1 OR R
1>17 THEN GOTO 1760
1750 IF X<>Q(R,C) THEN GOSUB 230
0
1760 NEXT R
1763 IF Q(N,C)(<>) THEN GOTO 715
1765 NEXT C
1770 GOTO 150
1800 REM FORMULA ENCODE
1805 LET X=1
1810 LET B$=A$(C)
1815 LET C$=""
1820 IF X=LEN B$+1 THEN GOTO 189
0
1822 IF B$(X)="P" THEN GOTO 900
1825 IF B$(X)("<")"K" AND B$(X)(">")"S
" THEN GOTO 1850
1830 LET X$="R"
1835 IF B$(X)("<")"K" THEN LET X$="
N"
1840 IF B$(X+2)="*" OR B$(X+2)="
/" OR B$(X+2)="*" OR B$(X+2)="+
" OR B$(X+2)="-" THEN GOTO 1875
1845 LET C$=C$+"Q("X$+", "+B$(X+

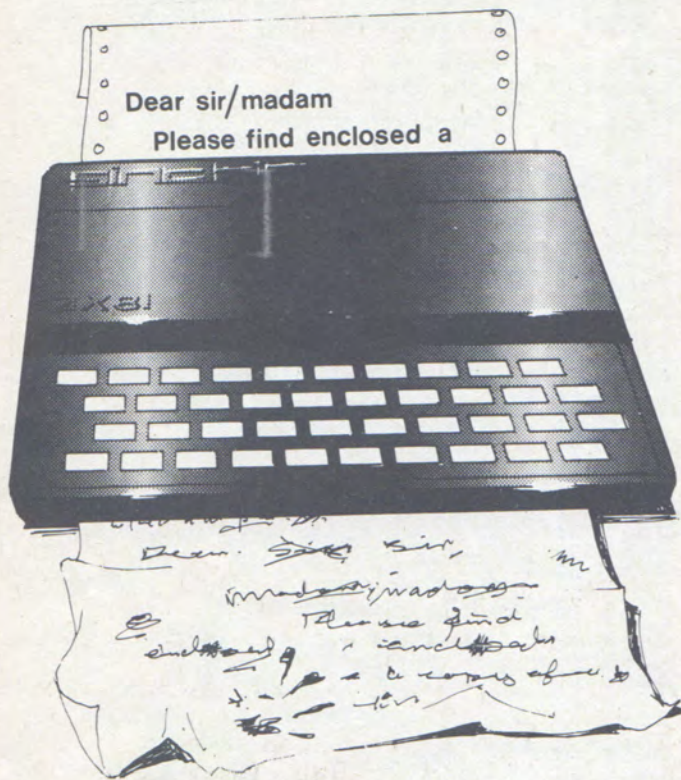
```

```

1 TO X+2)+")"
1850 LET X=X+3
1855 GOTO 1820
1860 LET C$=C$+B$(X)
1865 LET X=X+1
1870 GOTO 1820
1875 LET C$=C$+"Q("X$+", "+B$(X+
1)+")"
1880 LET X=X+2
1885 GOTO 1820
1890 LET A$(C)=C$
1895 RETURN
1900 REM SCROLL ROUTINE
1901 IF R1+2>N THEN RETURN
1905 LET K=K+1
1910 LET S=S+1
1915 PRINT AT 20,0;S
1920 SCROLL
1925 RETURN
1930 REM SCROLL SHIFT
1935 LET J=J+1
1940 IF S=N1 OR R=N1 THEN GOTO 1
50
1945 LET R=17+J
1950 LET K=K+1
1955 LET S=S+1
1960 PRINT AT 20,0;S;TAB 3;"
1970 SCROLL
1975 FOR C=ABS T+1 TO ABS T+C1
1976 IF C>M THEN GOTO 160
1977 IF A$(C,1 TO 2)=" " THEN G
OTO 1994
1980 LET U=C+T
1985 PRINT AT 19,C(U);INT (Q(R,C
)*100+.5)/100
1990 IF Q(N,C)=0 THEN GOTO 1994
1992 GOSUB 2400
1994 NEXT C
1998 GOTO 160
2000 REM CLEAR WORKSHEET
2001 CLS
2005 FOR C=1 TO M
2008 FOR R=1 TO N1
2009 LET Q(R,C)=0
2010 NEXT R
2020 IF Q(N,C)(<>) THEN LET Q(N,C
)=.001
2022 NEXT C
2035 GOTO 74
2300 REM PRINT ROUTINE
2305 PRINT AT R1+2,C(U);(INT (Q(
R,C)*100+.5)/100);"
2310 RETURN
2350 REM PRINT HEADINGS
2355 PRINT AT 2,C(U);H$(C)
2360 RETURN
2400 REM PRINT SUM ROUTINE
2405 PRINT AT 20,C(U);(INT (Q(N,
C)*100+.5)/100);"
2410 RETURN
3000 REM SHELL METZNER SORT
3001 PRINT AT 0,0;"ENTER COLUMN
TO BE SORTED
3002 INPUT C
3003 PRINT AT 0,0;"SORT FROM ?

3004 INPUT X
3005 PRINT AT 0,0;"SORT TO ? "
3007 INPUT Q
3009 LET R=1
3010 IF 2**R>N1 THEN GOTO 3025
3015 LET R=R+1
3020 GOTO 3010
3025 LET F=2**R-1
3030 LET F=INT (F/2)
3035 IF F=0 THEN GOTO 1315
3040 LET D=N1-F
3045 LET B=1
3050 LET R=B
3055 LET E=R+F
3060 IF Q(R,C)(>)Q(E,C) THEN GOTO
3074
3065 LET B=B+1
3070 IF B>D THEN GOTO 3030
3073 GOTO 3050
3074 FOR W=X TO Q
3075 LET T1=Q(R,W)
3080 LET Q(R,W)=Q(E,W)
3085 LET Q(E,W)=T1
3087 NEXT W
3090 LET R=R-F
3095 IF R<1 THEN GOTO 3065
3100 GOTO 3055
3500 SAVE "FORMCAL"
3501 GOTO 1

```



# ΕΠΕΞΕΡΓΑΣΙΑ ΚΕΙΜΕΝΟΥ

## WORD PROCESSING

- 1: SAVE ON TAPE
  - 2: SEND TO PRINTER
  - 3: WRITE PAGE
  - 4: EDIT PAGE
  - 5: CLEAR PAGE
- ENTER OPTION (1 TO 5)?

## COMPUTER GIA OLOYΣ

ΤΟ ΚΑΤΑΝΟΗΤΟ  
PERIODΙΚΟ

## GIA MIKRO-COMPUTER

MODE: EDITING PAGE: 7

```

10 REM *****
11 REM * WORD PROCESSOR *
12 REM *****
20 DIM A$(10,640)
30 LET B$=""
40 LET D$=""
50 LET POS=1
60 LET PAGE=1
65 GOTO 160
70 PRINT "WORDYS ENTER PROGRAM
NAME-"
110 INPUT Z$
115 IF Z$="" THEN GOTO 9050
120 PRINT "SET UP TAPE DECK THE
N PRESS (N/L)"
130 INPUT X$
140 SAVE Z$
150 CLS
170 PRINT "
180 PRINT " WORD PROCE

```

Είναι γνωστά σε όλους μας τα μειονεκτήματα που έχει η γραφομηχανή στο γράψιμο κάποιου κειμένου. Αν γίνει κάποιο λάθος ή παραληφθεί κάποιο γράμμα ή λέξη, είσαι αναγκασμένος να ξαναγράψεις το κείμενο από την αρχή, κάτι που συνεπάγεται χάσιμο πολύτιμου χρόνου. Άλλο ένα μειονέκτημα της γραφομηχανής είναι ότι μας δίνει μόνο ένα αντίτυπο του κειμένου και έτσι στην περίπτωση που χρειάζομαστε και άλλα αντίτυπα (δεν είναι λύση η φωτοτυπία) πρέπει να καθίσουμε να γράψουμε το κείμενο πάλι και πάλι απ'την αρχή.

Τα προβλήματα αυτά είχαν σαν φυσιολογική συνέπεια με την εξάπλωση των υπολογιστών να οδηγηθούν οι περισσότερες επιχειρήσεις στην λύση των κομπιούτερ (που εφο-

διασμένοι με ισχυρά προγράμματα) έχουν τη δυνατότητα επεξεργασίας κειμένου.

Ένα πρόγραμμα που μπορεί κάλλιστα να μετατρέψει τον ZX-81 σας σε ένα WORD PROCESSOR είναι αυτό που σας παρουσιάζουμε. Το πρόγραμμα είναι γραμμένο ολόκληρο σε BASIC κι'έτσι στην αρχή (προτού το περάσω και το τρέξω) μου είχε δημιουργήσει μερικές αμφιβολίες ως προς την αποτελεσματικότητα και την ταχύτητά του. Σας διαβεβαιώ όμως ότι έμεινα έκπληκτος! Η ευελιξία του είναι καταπληκτική και η ταχύτητά του ιδιαίτερα ικανοποιητική. Έτσι πιστεύω ότι με το πρόγραμμα αυτό αποδεικνύεται άλλη μια φορά ότι η BASIC του ZX-81 ενώ δεν μπορεί να πετύχει ικανοποιητικά αποτελέσματα στα video games τα καταφέρνει αρκετά

καλά σε επαγγελματικά προγράμματα.

Αε έρθουμε λοιπόν στο πρόγραμμά μας που έχει τις εξής δυνατότητες:

- 1) Γράψιμο (όπως είναι λογικό) του κειμένου στην οθόνη.
- 2) Editing (που είναι πραγματικά θαυμάσιο), δηλαδή τη δυνατότητα να διορθώσει ή να διαφοροποιήσει κάποιο σημείο του κειμένου.
- 3) Αυτόματο σώσιμο των κειμένων και του προγράμματος στην κασέτα.
- 4) "Dumping" του κειμένου (ή κάποιας "σελίδας") στον printer και
- 5) Αυτόματο "καθαρισμό" κάποιας σελίδας ή όλου του κειμένου.

Έτσι όταν πρωτοτρέξουμε το πρόγραμμα θα εμφανιστεί στην οθόνη μας το κυρίως μενού όπως φαίνεται στο σχήμα, που μας ζητά σε ποιο απ'τα πέντε σημεία θέλουμε να κάνουμε πρόσβαση. Στο σημείο αυτό θα ήθελα να ξεκαθαρίσω ότι το πρόγραμμα χρησιμοποιεί τα 14.7K της RAM διαθέτοντας 6.3K για αποθήκευση κειμένου που ισοδυναμούν με 10 σελίδες των 640 χαρακτήρων.

Ας αρχίσουμε από την πρόσβαση 3 (WRITE PAGE) στο επίπεδο δηλαδή εγγραφής του κειμένου. Στο σημείο αυτό το πρόγραμμα ζητάει το νούμερο της σελίδας (1-10) στην οποία θα κάνουμε την εγγραφή. Εφόσον αυτό δοθεί τότε η οθόνη μας θα πάρει τη μορφή μιας λευκής σελίδας που στο κατώτερο σημείο της γράφει σε ποιο επίπεδο βρίσκόμαστε, καθώς και τον αριθμό της σελίδας ενώ ο δρομέας θα βρίσκεται στο πάνω αριστερό σημείο της σελίδας. Έτσι στην πραγματικότητα ο ZX-81 σας έχει τώρα μετατραπεί σε μια γραφομηχανή με τη μόνη διαφορά ότι σαν πλήκτρο του SPACE χρησιμοποιείται το New line, ενώ το πλήκτρο του SPACE χρησι-

μοποιείται (λογικό είναι) για να κάνει BREAK και να βγαίνουμε από το πρόγραμμα.

Όταν μια γραμμή ξεπεράσει τους 32 χαρακτήρες αυτομάτως ο δρομέας κατεβαίνει στην αρχή της επόμενης ενώ στην περίπτωση που θέλετε ν'αλλάξετε γραμμή δεν χρειάζεται να γίνει αυτό πατώντας συνεχώς το πλήκτρο N/L αλλά απευθείας πατώντας Shift N/L. Εδώ θα ήθελα να τονίσω ότι το πρόγραμμα έχει την δυνατότητα επανάληψης κάποιου πλήκτρου (πατώντας π.χ. συνεχώς το πλήκτρο 0 θα γράφονται συνεχώς 0 στην οθόνη). Στην περίπτωση που θέλετε το σβήσιμο του τελευταίου χαρακτήρα που δώσατε, απλώς πατήστε (shift D), ενώ αν θέλετε να έρθετε από το επίπεδο writing που βρίσκεστε απευθείας σε επίπεδο editing πατείστε το πλήκτρο (shift W).

Αυτά λοιπόν όσον αφορά το επίπεδο writing ( τρίτη πρόσβαση στο μενού) και ας έρθουμε τώρα στο βασικότερο και δυσκολότερο σημείο του προγράμματος (πρέπει να θυμόμαστε τη λειτουργία 21 πλήκτρων για editing), δίνοντας το νούμερο 4 στο μενού. Το editing mode, όπως είπαμε και παραπάνω, χρησιμοποιείται για τη διόρθωση ή την αλλαγή κάποιου σημείου του κειμένου (λογικό είναι να υπάρχει κάποια γραμμένη σελίδα για να έλθουμε στο επίπεδο αυτό). Έτσι το πρώτο που θα εμφανιστεί όταν έρθουμε στο επίπεδο αυτό είναι μια ερώτηση που ρωτά σε ποια σελίδα του κειμένου θα επεμβούμε. Εφόσον δώσουμε το νούμερο της σελίδας (1-10), αμέσως θα εμφανιστεί η σελίδα αυτή με το δρομέα ν'αναβοσβήνει στο πάνω αριστερό σημείο της οθόνης. Για το editing καταρχάς μπορούμε να χρησιμοποιήσουμε τα

```
S S I N G "
190 PRINT "
200 PRINT AT 4,2;" 1:SAVE ON T
APE"
210 PRINT AT 6,2;" 2:SEND TO P
RINTER"
220 PRINT AT 8,2;" 3:WRITE PAG
E"
230 PRINT AT 10,2;" 4:EDIT PAG
E"
240 PRINT AT 12,2;" 5:CLEAR PA
GE"
245 PRINT AT 14,0;" ENTER OP
TION (1 TO 5)"
250 PRINT AT 14,25;" "
260 LET X$=INKEY$
270 IF X$="1" THEN GOTO 70
280 IF X$="2" THEN GOTO 350
295 PRINT AT 14,25;"?"
300 IF X$="3" THEN GOTO 500
310 IF X$="4" THEN GOTO 800
320 IF X$="5" THEN GOTO 4000
330 GOTO 250
360 PRINT "ENTER PAGE TO PRINTE
D OUT"
370 INPUT X
380 IF X<1 THEN GOTO 400
390 IF X=11 THEN GOTO 420
395 IF X>11 OR X<1 THEN GOTO 93
400 LPRINT A$(X)
410 GOTO 430
420 FOR X=1 TO 10
422 LPRINT A$(X)
424 NEXT X
430 PRINT " ANYMORE"
440 LET X$=INKEY$
480 IF X$="Y" THEN GOTO 350
490 IF X$="N" THEN GOTO 150
495 GOTO 440
510 PRINT
520 PRINT "ENTER PAGE FOR WRIT
ING ON"
530 INPUT X
540 IF X>10 OR X<1 THEN GOTO 93
545 LET POS=1
550 PRINT AT 21,0;" MODE:WRITTI
NG";AT 21,22;"PAGE:";X
580 PRINT AT 20,0;" "
600 PRINT AT 0,0;A$(X)
605 POKE 16441,20
606 POKE 16442,30
620 LET WL=INT (POS/32)
630 LET WC=POS-WL*32-1
635 IF POS/32=INT (POS/32) THEN
GOSUB 6000
640 PRINT AT WL,WC;" "
650 IF INKEY$=" " THEN GOTO 650
660 LET X$=INKEY$
670 IF X$=CHR$ 224 THEN GOTO 15
675 IF X$=CHR$ 121 THEN GOTO 73
680 IF X$=CHR$ 118 THEN GOTO 74
685 IF X$=CHR$ 228 THEN GOTO 30
686 IF X$=CHR$ 217 THEN GOTO 85
687 IF X$=CHR$ 121 THEN GOTO 85
690 LET A$(X,POS)=X$
700 PRINT AT WL,WC;A$(X,POS)
710 LET POS=POS+1
720 IF POS>640 OR POS<1 THEN GO
TO 380
730 GOTO 620
731 LET UU=(WL+1)*32-POS
732 LET POS=POS+UU+1
733 PRINT AT WL,WC;A$(X,POS-UU-
1)
734 GOTO 720
740 LET X$=" "
750 GOTO 690
800 PRINT
810 PRINT "ENTER PAGE FOR EDITI
NG"
820 INPUT X
830 IF X>10 OR X<1 THEN GOTO 90
840 LET POS=1
850 PRINT AT 20,0;" "
855 PRINT AT 21,0;" MODE:EDITIN
G";AT 21,22;"PAGE:";X
860 PRINT AT 0,0;A$(X)
870 LET WL=INT (POS/32)
```

πλήκτρα (5,6,7,8) για να οδηγήσουμε το δρομέα στο σημείο που θέλουμε. Έτσι τα πλήκτρα αυτά είναι τα πρώτα τέσσερα από τα 21 συνολικά που πρέπει να θυμάστε για το editing. Το πλήκτρο D είναι το πλήκτρο του σβησίματος (DELETE). Πατώντας το σβήνεται ο χαρακτήρας που είναι αριστερά από το δρομέα. Με το πλήκτρο S SPACE προστίθεται ένα κενό μεταξύ του χαρακτήρα που βρίσκεται αριστερά του δρομέα και εκείνου που εκείνη τη στιγμή καλύπτεται απ' αυτόν. Με το πλήκτρο H (HOME) ο δρομέας επιστρέφει αυτόματα στο πρώτο σημείο της σελίδας. Πατώντας το πλήκτρο 1 προστίθεται μια γραμμή (32 χαρακτήρες) από SPACE αριστερά από τη θέση του δρομέα. Έτσι όταν θέλετε μια γραμμή του κειμένου σας να ξεχωρίζει απ' τις άλλες απλά φέρνετε το δρομέα στο πρώτο σημείο της παρακάτω γραμμής και πατήστε το πλήκτρο L. Το πλήκτρο αυτό λειτουργεί με τον ίδιο τρόπο όπως και το πλήκτρο S με τη διαφορά ότι το (S) αφήνει 1 κενό (SPACE) ενώ το L αφήνει απευθείας 32 κενά. Όταν τώρα θελήσετε να αφαιρέσετε ένα αριθμό χαρακτήρων από το κείμενο (Ø-639) φέρτε καταρχάς το δρομέα αριστερά από τον πρώτο χαρακτήρα που θέλετε να σβήσετε και πατήστε το πλήκτρο 1. Αμέσως θα εμφανιστεί στο κάτω αριστερό σημείο της οθόνης η ερώτηση AMOUNT ? που ζητά τον αριθμό των χαρακτήρων που θέλετε να σβήσετε. Εδώ θέλω να τονίσω ένα σημείο πρόνοιας του προγράμματος. Πιθανόν να μην υπολογίσετε σωστά και σβήσετε κατά λάθος και χαρακτήρες που δεν έπρεπε να σβηστούν, το πρόγραμμα όμως κρατά σε μια buffer μνήμη τους χαρακτήρες που

σβήσατε και έτσι πατώντας το πλήκτρο P σας επιστρέφει αμέσως την σελίδα όπως ακριβώς ήταν προτού πατήσετε το πλήκτρο 1. Όταν κατά λάθος ενώ γράφατε το κείμενο ξεχάσατε κάποια λέξη ή ολόκληρη πρόταση τότε χρησιμοποιείτε το πλήκτρο (I) (INSERT) που τοποθετεί ένα text μεταξύ του χαρακτήρα που είναι (πάνω) απ' αυτόν. Για να διορθώσετε κάποιον χαρακτήρα απλά οδηγείστε το δρομέα πάνω σ' αυτόν και πατήστε το πλήκτρο T. Το πλήκτρο αυτό έχει σαν λειτουργία να μετατρέπει το χαρακτήρα που βρίσκεται κάτω από το δρομέα με αυτόν που θα πληκτρολογήσετε στη συνέχεια. Είχαμε αναφέρει προηγουμένως κάτι για την buffer (η οποία κρατά τους χαρακτήρες που αλλάχθηκαν την τελευταία φορά). Το πλήκτρο (O) χρησιμοποιείται για να "καθαρίσετε" αυτή τη μνήμη. Αν τώρα θέλετε ο δρομέας να κινείται σε επίπεδο FAST πατήστε το πλήκτρο (M). Για να επιστρέψετε στο SLOW απλά πατήστε N/L. Ως τώρα σίγουρα δεν πρέπει να είχατε κάποια δυσκολία στο να κατανοήσετε τις λειτουργίες των διαφόρων πλήκτρων του editing, ίσως όμως η λειτουργία του πλήκτρου (J) φανεί σε πολλούς κάπως δύστροπη. Πατώντας αυτό το πλήκτρο (Justification function) το πρόγραμμα ελέγχει τα κενά μεταξύ των λέξεων των γραμμών που επηρεάστηκαν από την τελευταία εγγραφή και τα αυξάνει συμμετρικά. Με άλλα λόγια κάνει ένα συμμετρικό αραίωμα των λέξεων (αναλογικό διάστημα). Εδώ θα πρέπει να εξηγήσουμε και την λειτουργία δυο control χαρακτήρων του (·) που δίνει στην εντολή J να καταλάβει ότι πρέπει να κάνει justification από το σημείο μόνο που βρίσκεται ο χαρακτήρας

```

880 LET WC=POS-WL*32-1
885 IF POS/32=INT (POS/32) THEN
GOSUB 8000
890 PRINT AT WL,WC;CHR$(128+CO
DE A$(X,POS))
900 PRINT AT WL,WC;A$(X,POS)
905 IF INKEY$="" THEN GOTO 890
910 LET X$=INKEY$
920 LET POS=POS+(X$="8")-(X$="5
")+32*(X$="6")-32*(X$="7")
930 IF POS>640 OR POS<1 THEN GO
TO 9070
940 IF CODE X$<37 AND CODE X$>3
20 THEN GOTO 870
950 IF X$="H" THEN LET POS=1
955 IF X$="D" THEN GOTO 2200
960 IF X$="S" THEN GOTO 1100
970 IF X$="L" THEN GOTO 1200
980 IF X$="A" THEN GOTO 1500
990 IF X$="I" THEN GOTO 1300
1000 IF X$="C" THEN GOTO 1400
1010 IF X$="B" THEN GOTO 1600
1020 IF X$="T" THEN GOTO 1700
1030 IF X$="1" THEN GOTO 1800
1040 IF X$="P" THEN GOTO 1900
1050 IF X$="R" THEN GOTO 2000
1060 IF X$="E" THEN GOTO 2400
1070 IF X$="O" THEN LET B$=""
1075 IF X$="J" THEN GOTO 7000
1080 IF X$="U" THEN GOTO 3500
1085 IF X$="M" THEN GOTO 8600
1090 GOTO 870
1100 LET B$=A$(X,640)+B$
1120 LET A$(X)=A$(X,1 TO POS-1)+
"+A$(X,POS TO 640)"
1130 LET POS=POS+1
1135 IF LEN B$>200 THEN GOTO 909
0
1140 GOTO 860
1210 LET B$=A$(X,640-32 TO 640)+
B$
1220 LET A$(X)=A$(X,1 TO POS-1)+
"+A$(X,POS TO 640)"
1230 IF LEN B$>200 THEN GOTO 909
0
1240 GOTO 860
1310 PRINT AT 21,0;"ENTER TEXT"
1320 INPUT X$
1330 LET A=LEN X$
1340 LET B$=A$(X,640-A TO 640)+B
$
1350 LET A$(X)=A$(X,1 TO POS-1)+
X$+A$(X,POS TO 640)
1360 IF LEN B$>200 THEN GOTO 909
0
1370 GOTO 860
1410 LET A$(X,POS)=" "
1420 GOTO 860
1510 GOSUB 2500
1515 IF X=10 THEN GOTO 0010
1520 LET X=X+1
1530 GOTO 850
1610 GOSUB 2500
1615 IF X=1 THEN GOTO 9010
1620 LET X=X-1
1630 GOTO 850
1710 FOR F=1 TO 100
1711 NEXT F
1720 IF INKEY$="" THEN GOTO 1720
1740 LET A$(X,POS)=INKEY$
1750 GOTO 860
1810 PRINT AT 21,0;" AMOUNT
?"
1820 INPUT A
1830 LET D$=A$(X,POS-A TO POS-1)
1840 LET A$(X)=A$(X,1 TO POS-A-1
)+A$(X,POS TO 640)
1850 GOTO 850
1910 LET A$(X)=A$(X,1 TO POS-1)+
D$+A$(X,POS TO 640)
1920 LET D$=""
1930 GOTO 850
2010 GOSUB 2500
2020 GOTO 850
2300 IF B$="" THEN LET A$(X)=A$(
X,1 TO POS-2)+A$(X,POS TO 640)
2310 IF B$<>"" THEN LET A$(X)=A$(
X,1 TO POS-2)+A$(X,POS TO 640)+
B$(1)
2320 IF B$<>"" THEN LET B$=B$(2
TO )
2330 LET POS=POS-1
2340 GOTO 860
2410 GOSUB 2500
2420 GOTO 150
2502 IF B$="" THEN RETURN
2505 FAST
2510 LET A=LEN B$
2520 FOR B=X+1 TO 9
2530 LET C$=A$(B,640-A TO 640)
2540 LET A$(B)=B$+A$(B,1 TO 640-

```

ως το τέλος και ο control χαρακτήρας (/) για να γίνει το justification μέχρι το σημείο αυτό και να επιστρέψει αμέσως μετά το πρόγραμμα σε edit mode. Προτού σας εξηγήσω τη λειτουργία του πλήκτρου A σας εξηγήσω τι ακριβώς συμβαίνει όταν τοποθετήσετε σε κάποιο σημείο του κειμένου ένα text (με το πλήκτρο I) έτσι ώστε όμως αυτό να έχει σαν αποτέλεσμα να χαθούν μερικοί χαρακτήρες από την τελευταία γραμμή της σελίδας, γιατί όπως είπαμε και στην αρχή μια σελίδα χωρά μέχρι 640 χαρακτήρες. Οι χαρακτήρες αυτοί δεν χάθηκαν αλλά απλώς τοποθετήθηκαν σαν πρώτες γραμμές της επόμενης σελίδας. Έτσι μην παραξανευτείτε όταν πατήσετε το πλήκτρο A που σας φέρνει την αμέσως επόμενη σελίδα από αυτήν που βρισκόσαστε και δείτε σ'αυτή κάτι γραμμένο, ενώ εσείς πιστεύατε ότι δεν είχατε γράψει τίποτα. Το πλήκτρο B κάνει ακριβώς την αντίθετη δουλειά από το A, δηλαδή σας επιστρέφει αυτόματα στην προηγούμενη σελίδα από αυτήν που βρίσκεστε.

Ένα πολύ χρήσιμο πλήκτρο είναι και το (W) που σας επιστρέφει (χωρίς να έλθετε πρώτα στο μενού) από το επίπεδο editing που βρίσκεστε τώρα απευθείας σε επίπεδο writing. Πως επιστρέφετε στο μενού; Απλώς πατείστε το πλήκτρο (E). Αν θέλετε να επανατοποθετήσετε στην buffer ότι περιείχε πρώτου την τελευταία αλλαγή πατείστε το πλήκτρο R (REPLACE), ενώ αν θέλετε να σβήσετε τον χαρακτήρα πάνω από τον οποίο βρίσκεται ο δρομέας πατείστε το πλήκτρο (C). Το πλήκτρο αυτό λειτουργεί όπως ακριβώς το πλήκτρο (T) με τη διαφορά ότι τοποθετεί ένα κενό (SPACE) αντί κάποιου άλλου

χαρακτήρα στη θέση του δρομέα. Επιτέλους τελειώσαμε με τα πλήκτρα του editing. Αν δεν το πιστεύετε μετρήστε τα: είναι πράγματι 21, (5,6,7,8,D,S,H,L,I,P,I,T,O,M,J,A,B,E,R,C,N).

Ερχόμαστε τώρα στις άλλες διευθύνσεις του μενού. Το 1 (SAVE TO TAPE) όπως καταλαβαίνετε είναι για να κάνει saving του προγράμματος στην κασέτα. Πρώτα δώστε το όνομα με το οποίο θα σώσετε το πρόγραμμά σας (δεν πρέπει να σώζετε το πρόγραμμα πάντοτε με το ίδιο όνομα για να μπορείτε να ξεχωρίζετε μεταξύ τους τα διάφορα κείμενα) και αφού ετοιμάσετε το κασετόφωνό σας πατείστε N/L. Εδώ σας προτείνω να ακολουθείτε πάντα αυτή την οδό για το σύστημα των κειμένων σας και ποτέ δίνοντας την εντολή SAVE γιατί με τον τρόπο αυτό το πρόγραμμα τρέχει απευθείας μόλις φορτωθεί και δεν περιμένει να του δώσετε την εντολή GOTO 150. Αν κατά λάθος στην περίπτωση αυτή δίνετε RUN θα είχατε σβήσει όλο το κείμενο.

H option 2 του μενού (SEND TO PRINTER) χρησιμοποιεί για να τυπώσετε κάποια σελίδα του κειμένου (1-10) στον printer. Δίνοντας εδώ σαν νούμερο σελίδας το 11 θα τυπωθούν ταυτόχρονα και οι 10 σελίδες, ενώ δίνοντας το νούμερο 12 επιστρέφετε στο μενού αν την τελευταία στιγμή αποφασίσετε να μην τυπώσετε κάποια σελίδα. Και σαν τελευταία να εξηγήσουμε την option (5) που χρησιμοποιεί για να καθαρίζουμε κάποια σελίδα δίνοντας απλά το νούμερο της (1-10).

Όπως βλέπετε το πρόγραμμα έχει πραγματικά θαυμάσιες δυνατότητες και σίγουρα αξίζει τον κόπο να το πληκτρολογήσετε και να μετατρέψετε τον ZX-81 σας σε ένα word processor.

```

0000 1)
0005 LET B#=C#
0010 NEXT B
0015 LET B#=""
0020 SLOW
0025 RETURN
0030 LET POS=POS-1
0035 LET A$(X,POS+1)=" "
0040 GOTO 600
0050 GOSUB 2500
0055 GOTO 550
0060 LET X=X+1
0065 IF X>10 THEN GOTO 9010
0070 LET POS=1
0075 GOTO 550
0080 PRINT " PAGE TO CLEAR ?"
0085 INPUT X
0090 IF X>10 OR X<1 THEN GOTO 90
0095
0100 LET A$(X)=" "
0105 GOTO 150
0110 LET GH=0
0115 LET AS=2
0120 FAST
0125 FOR Z=1 TO 640-32 STEP 32
0130 FOR Y=Z+31 TO Z STEP -1
0135 IF A$(X,Y)="" THEN GOTO 71
0140 IF A$(X,Y)="" THEN GOTO 71
0145 IF A$(X,Y)="" THEN GOTO 7
0150 LET GH=GH+1
0155 NEXT Y
0160 LET A$(X)=A$(X,1 TO Z+31-GH)
0165 +A$(X,Z+32 TO 640)
0170 FOR Y=Z TO Z+31
0175 IF GH=0 THEN GOTO 7100
0180 IF A$(X,Y)="" THEN GOSUB 7
0185
0190 NEXT Y
0195 IF GH<>0 THEN GOTO 7800
0200 LET GH=0
0205 LET AS=2
0210 NEXT Z
0215 SLOW
0220 GOTO 850
0225 LET A$(X)=A$(X,1 TO Y-1)+"
0230 +A$(X,Y TO 640)
0235 LET GH=GH-1
0240 LET Y=Y+AS
0245 RETURN
0250 LET AS=AS+1
0255 GOTO 7070
0260 LET WC=31
0265 LET WL=WL-1
0270 RETURN
0275 FAST
0280 LET WL=INT (POS/32)
0285 FAST
0290 LET WC=POS-WL*32-1
0295 IF POS/32=INT (POS/32) THEN
0300 GOSUB 8000
0305 PRINT AT WL,WC;" "
0310 SLOW
0315 IF INKEY#="" THEN GOTO 8650
0320 PRINT AT WL,WC;A$(X,POS)
0325 LET X#=INKEY#
0330 IF X#=CHR$ 118 THEN GOTO 85
0335
0340 LET POS=POS+(X#="8")-(X#="5
0345 ") +32*(X#="6") -32*(X#="7")
0350 GOTO 8610
0355 PRINT AT 21,0;" <ERROR> YOU
0360 HAVE USED ALL FILES "
0365 GOTO 9910
0370 PRINT AT 21,0;" <ERROR> THER
0375 E IS NO SUCH PAGE "
0380 GOTO 9910
0385 PRINT AT 21,0;" <ERROR> PROG
0390 RAM NAME IS INVALID "
0395 GOTO 9910
0400 PRINT AT 21,0;" <ERROR> RUNN
0405 ING INTO NEXT PAGE "
0410 GOTO 9910
0415 PRINT AT 21,0;" <ERROR> WORK
0420 ING BUFFER IS FULL "
0425 GOSUB 2500
0430 GOTO 9910
0435 PRINT AT 21,0;" <ERROR>TEXT I
0440 NSERTION IS TO LARGE"
0445 FOR F=1 TO 20
0450 NEXT F
0455 GOTO 850
0460 FOR F=1 TO 100
0465 NEXT F
0470 GOTO 150

```



## PR. NAME

Σε όλους μας έχει τύχει σε κάποια στιγμή να μην γνωρίζουμε το όνομα ενός προγράμματος που είχαμε γράψει παλιότερα σε μια κασέτα. Έτσι για να δούμε ποιο πρόγραμμα ήταν, έπρεπε να το φορτώσουμε (και αν ήταν απ'τα πολύ μεγάλα μπορούσαμε να περιμένουμε και 8 λεπτά) και πιθανότατα τελικά αυτό να μην ήταν αυτό που ψάχναμε.

Το προγραμματάκι αυτό που σας παρουσιάζουμε έχει σαν σκοπό του να βρίσκει και να μας δίνει το όνομα με το οποίο είχαμε παλιότερα κάνει SAVE το πρόγραμμα. Έτσι μπορούμε να ανακαλύψουμε γρήγορα (σ'ελάχιστα δευτερόλεπτα) και χωρίς καμιά "παρενέργεια" (είτε στο κύριο πρόγραμμα είτε στο ίδιο το PR. NAME) το πρόγραμμα που μας ενδιαφέρει. Το πρόγραμμα αυτό λειτουργεί περίπου όπως η εντολή LOAD (δηλαδή φορτώνει το

πρόγραμμα απ'το κασετόφωνο) αλλά με τη διαφορά ότι φορτώνει μόνο τα 90 πρώτα bytes του προγράμματος στα οποία είναι φυλαγμένο το όνομά του.

Το πρόγραμμα μόλις βρει το όνομα κάποιου προγράμματος αμέσως το "επιστρέφει" και περιμένει να πατήσουμε N/L για να συνεχίσει σε περίπτωση που θέλουμε να βρει το όνομα κάποιου άλλου προγράμματος. Για εκείνους που έχουν μια κασέτα και έχουν χάσει τις σημειώσεις της και θέλουν να βρουν ποια προγράμματα έχουν σ'αυτήν φυλαγμένο τότε σας αφαιρέσουν τη γραμμή 40 και αφού φέρουν την κασέτα στην αρχή της ασ αφήσουν το πρόγραμμα να τρέχει συνέχεια μέχρι να τελειώσει η κασέτα. Αμέσως μόλις κάνετε BREAK θα υπάρχουν στην οθόνη σας όλα τα ονόματα των προγραμμάτων με την σειρά όπως παλιότερα τα είχατε κάνει SAVE.

```

1 REM LN 7?LN 0RND/CLS :#Y
2 <= RETURN PEEK COPY 3ABS #*3)
3 ( LET LET LN 0RNDACS ??4 NOT *K
4 POKE /1STA$ 2 1<= RETURN #ACS
5 ??S PRINT ( PRINT SGN 4 RETURN
6 ?K005 ZACS )K?TAN TAN M
7 10 RAND USR 16514
8 200 CLEAR
9 30 PRINT " ";
10 40 INPUT A$
11 50 GOTO 10
12 60 SAVE "PR.NAME"
13 70 LIST

```

16514	20
16515	30
16516	150
16517	200
16518	1000
16519	04
16520	4
16521	51
16522	4
16523	1
16524	0
16525	0
16526	0
16527	107
16528	10
16529	04
16530	11
16531	5
16532	1
16533	1
16534	0
16535	3
16536	3
16537	3
16538	0
16539	17
16540	0
16541	41
16542	41
16543	0
16544	0
16545	4
16546	0
16547	0
16548	1
16549	0
16550	1
16551	15
16552	0
16553	48
16554	44
16555	4
16556	0
16557	13
16558	0
16559	48
16560	0
16561	6
16562	0
16563	10
16564	4
16565	0
16566	0
16567	100
16568	100
16569	0
16570	45
16571	0
16572	45
16573	0
16574	0
16575	4
16576	54
16577	0
16578	0
16579	0
16580	0
16581	0
16582	17
16583	48
16584	105
16585	201
16586	201





## MUSIC MACHINE

Σίγουρα όλοι σας θα θέλατε ο ZX-81 να είχε ήχο, έστω και εκείνο το BEEP που έχει ο SPECTRUM, όμως η πραγματικότητα είναι σκληρή. Να όμως ένα πρόγραμμα που σας δίνει την δυνατότητα όχι μόνο να ακούσετε κάποιο ήχο αλλά και μουσική (όχι βέβαια HiFi) από τον μικρό σας ZX-81. Εδώ σίγουρα θα αναρωτηθείτε "μα καλά εγώ ξέρω ότι ο ZX-81 δεν έχει ούτε ένα μεγαφωνάκι πως...". Η απάντηση είναι ότι η μουσική θα ακούγεται

μέσα από το μεγάφωνο της τελεόρασής σας.

Το πρόγραμμα μετατρέπει τον ZX-81 σε ένα ασ πούμε υποτυπώδες πιανάκι με έξη πλήκτρα (1,2,3,4,5,6) καθένα από τα οποία δίνει μια διαφορετική νότα. Για να ακούγεται καλύτερα η μουσική σας προτείνω να αποσυντονίσετε λίγο το κανάλι της τηλεόρασής σας.

Βέβαια μην περιμένετε να γίνετε πιανίστας με αυτό το πρόγραμμα αλλά θα διασκεδάσετε και θα έχετε μια ευχάριστη εμπειρία.

```

1 REM MUSIC MACHINE
2 FAST
3 SLOW
4 FAST
5 IF INKEY$="3" THEN GOTO COD
6 "C" IF INKEY$="4" THEN GOTO VAL
7 "C" IF INKEY$="5" THEN GOTO COD
8 "C" IF INKEY$="6" THEN GOTO VAL
9 "200"
10 GOTO CODE "C"
11 FAST
12 SLOW
13 SLOW
14 FAST
15 IF INKEY$="2" THEN GOTO COD
16 "C" IF INKEY$="4" THEN GOTO VAL
17 "C" IF INKEY$="5" THEN GOTO COD
18 "C" IF INKEY$="6" THEN GOTO VAL
19 "200"
20 GOTO CODE "C"
21 FAST
22 FAST
23 SLOW
24 SLOW
25 SLOW
26 SLOW
27 SLOW
28 FAST
29 IF INKEY$="2" THEN GOTO COD
30 "C" IF INKEY$="3" THEN GOTO COD
31 "C" IF INKEY$="5" THEN GOTO COD
32 "C" IF INKEY$="6" THEN GOTO VAL
33 "200"
34 GOTO VAL "80"
35 FAST
36 SLOW
37 SLOW
38 SLOW
39 SLOW
40 FAST
41 IF INKEY$="4" THEN GOTO VAL
42 "C" IF INKEY$="2" THEN GOTO COD
43 "C" IF INKEY$="3" THEN GOTO COD
44 "C" IF INKEY$="6" THEN GOTO VAL
45 "200"
46 GOTO CODE "C"
47 FAST
48 FAST
49 SLOW
50 SLOW
51 SLOW
52 SLOW
53 FAST
54 IF INKEY$="2" THEN GOTO COD
55 "C" IF INKEY$="3" THEN GOTO COD
56 "C" IF INKEY$="4" THEN GOTO VAL
57 "C" IF INKEY$="6" THEN GOTO COD
58 "C" GOTO VAL "200"

```

# 3D — GRAPH

Πολλοί από σας σίγουρα θα έχετε ακουστά για τα 3D (three dimensions) προγράμματα που έχουν κυκλοφορήσει πολλές εταιρίες. Λίγοι όμως θα είχατε την ευκαιρία να έχετε μια "τριδιάστατη εμπειρία".

Να λοιπόν που τώρα έχετε την ευκαιρία να αποκτήσετε την μοναδική εμπειρία της τρισδιάστατης απεικόνισης. Βέβαια το πρόγραμμα αυτό δεν έχει γραφτεί με σκοπό να κάνει απλώς όμορφες τρισδιάστατες απεικονίσεις, αλλά καθαρά σαν μαθηματικό πρόγραμμα απεικόνισης της γραφικής παράστασης οποιασδήποτε εξίσωσης της μορφής  $Z=f(x,y)$ , (3 διαστάσεις  $X, Y, Z$ ). Σίγουρα θα ικανοποιηθεί και εκείνος που θέλουν απλώς να φτιάξουν μια ωραία εικόνα στην οθόνη ή στον printer τους.

Το πρόγραμμα σας ζητά καταρχάς τις ελάχιστες και τις μέγιστες τιμές (μέσα στα όρια των οποίων θα πάρουμε τη γραφική παράσταση) καθώς και το βήμα που θα παίρνουν οι δυο ανεξάρτητες μεταβλητές  $X, Y$ . Κατόπιν θα σας ζητηθεί η εξίσωση που θέλετε να σχηματισθεί μέσα στα όρια των παραπάνω τιμών. Εδώ σας λέμε ότι την εξίσωση θα την δώ-

σετε όπως ακριβώς είναι δηλαδή (π.χ.  $X+Y, \sin(X+Y)+\cos(X/2+Y), X*X+Y*Y$ ). Εφόσον τη δώσετε θα περιμένετε λίγο να γίνουν (σε FAST) ορισμένοι απαραίτητοι υπολογισμοί και τότε θα αρχίσει να σχηματίζεται στην οθόνη σας η τρισδιάστατη γραφική παράσταση. Θα πρέπει να σημειώσουμε ότι όποτε θελήσετε να δώσετε μια εξίσωση στην οποία το  $X$  ή το  $Y$  θα υψώνεται σε κάποια δύναμη (π.χ.  $X**2+Y$ ), είναι προτιμότερο να αποφεύγετε την ύψωση στην δύναμη (λόγω του ότι ο ZX-81 δεν μπορεί να υψώσει αρνητικό αριθμό σε δύναμη) και έτσι η εξίσωση του προηγούμενου παραδείγματος να δοθεί υπό την μορφή  $(X*X+Y)$ . Θέλω να πω επίσης, ότι το πρόγραμμα μπορεί να χρησιμοποιηθεί και από τους κατόχους Spectrum αφαιρώντας απλώς τις εντολές FAST και SLOW και αλλάζοντας τα νούμερα 62,42 (γραμμές 440, 450) σε 255, 175.

Για όσους τέλος δεν ξέρουν με ποια εξίσωση να κάνουν αρχή προτείνω την  $(X*X+Y*Y)$  με κατώτερο όριο των  $X, Y$  το -3 και με ανώτερο το 3, έχοντας βήμα το .2 που σίγουρα σε πολλούς θα θυμίσει το δίχτυ που παλιότερα ο μπακάλης έβαζε τα σαλάμια.

```

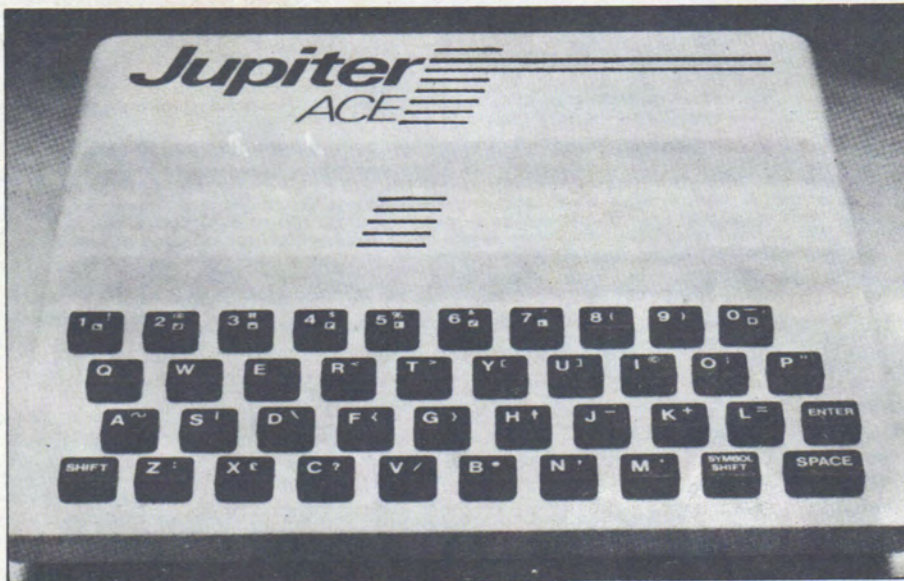
1 REM 3D-GRAPH
2 REM (C) BY M.MHNDRINOS
10 LET A=SQR 2/2
20 LET MINXA=10**38
30 LET MAXXA=-MINXA
40 LET MAXZA=MAXXA
50 LET MINZA=MINXA
60 PRINT "DOSE MOY TO MIN(X) :
70 INPUT X1
80 PRINT X1
90 PRINT "DOSE MOY TO MAX(X) :
100 INPUT X2
110 PRINT X2
120 PRINT "DOSE MOY TO BHMMA TOY
X :
130 INPUT DX
140 PRINT DX
150 PRINT
160 PRINT "DOSE MOY TO MIN(Y) :
170 INPUT Y1
180 PRINT Y1
190 PRINT "DOSE MOY TO MAX(Y) :
200 INPUT Y2
210 PRINT Y2
220 PRINT "DOSE MOY TO BHMMA TOY
Y :
230 INPUT DY
240 PRINT DY
250 PRINT
260 PRINT "DOSE MOY THN EZISVSH
270 INPUT A$
280 PRINT "F(X) = ";A$
290 PRINT
300 PRINT "PRESS N/L TO START :
310 INPUT B$
320 FAST
330 FOR Y=Y1 TO Y2 STEP DY
340 FOR X=X1 TO X2 STEP DX
350 LET Z=VAL A$
360 LET XA=X-Y*A
370 LET ZA=Z-Y*A
380 IF ZA>MAXZA THEN LET MAXZA=
ZA
390 IF ZA<MINZA THEN LET MINZA=
ZA
400 IF XA>MAXXA THEN LET MAXXA=
XA
410 IF XA<MINXA THEN LET MINXA=
XA
420 NEXT X
430 NEXT Y
440 LET LX=62/(MAXXA-MINXA)
450 LET LZ=42/(MAXZA-MINZA)
460 CLS
470 SLOW
480 FOR Y=Y1 TO Y2 STEP DY
490 FOR X=X1 TO X2 STEP DX
500 LET Z=VAL A$
510 LET XA=X-Y*A
520 LET XX=(XA-MINXA)*LX
530 LET ZZ=(ZA-MINZA)*LZ
540 PLOT XX,ZZ
550 NEXT X
560 NEXT Y
570 STOP

```



Προγράμματα για τόν:

# JUPITER ACE



ΕΠΙΜΕΛΕΙΑ:  
Φώτης Καρατζιάς

Από τους 256 χαρακτήρες που μπορεί ν' απεικονίσει ο Ace οι πρώτοι 128 είναι κανονικοί (λευκοί σε μαύρο φόντο) και οι υπόλοιποι inverse video (μαύροι σε λευκό φόντο). Ο κομπιούτερ δίνει τη δυνατότητα στο χρήστη να προγραμματίσει ο ίδιος τη μορφή κάθε χαρακτήρα. Επειδή με κάθε τέτοιον αναπρογραμματισμό αλλάζει όχι μόνο ο κανονικός χαρακτήρας αλλά κι ο αντίστοιχος inverse video, στη συνέχεια θ' αναφερθούμε σ' αυτούς που έχουν κωδικό ASCII από 0 μέχρι 127. Προκειμένου να δώσουμε στον Ace ελληνικό αλφάβητο, αφού εξετάσουμε στο εγχειρίδιο τα σύμβολα που έχουν ανατεθεί σε καθέναν απ' αυτούς τους κωδικούς, βλέπουμε ν' ανοίγονται μπροστά μας δυο εναλλακτικές λύσεις: Η πρώτη είναι να "καταστρέψουμε" τα πεζά αγγλικά (που έχουν κωδικούς ASCII 97-122) και να βάλουμε στη θέση τους τα κεφαλαία ελληνικά. Το πλεονέκτημα αυτής της μεθόδου είναι ότι μπορούμε να έχουμε κανονικό ελληνικό πληκτρολόγιο με τα σωστά γράμματα στις σωστές θέσεις, στο οποίο θα μπορούμε να μεταπηδάμε μ' ένα απλό πάτημα του "CAPS LOCK". Η δεύτερη λύση είναι να αναθεσουμε τα 10 ελληνικά κεφαλαία που δεν έχουν την ίδια μορφή με κανένα αγγλικό γράμμα σε "κενούς" κωδικούς ASCII. Με τη μέθοδο αυτή, τα γράμματα δεν είναι τοποθετημένα στα σωστά πλήκτρα. Από την άλλη μεριά όμως, αποκτάμε ελληνικό αλφάβητο χωρίς να χάσουμε τίποτα απ' το ενδοκτισμένο ρεπερτόριο χαρακτήρων του Ace. Όταν δεν πρόκειται λοιπόν να συντάξουμε εκτεταμένα κείμενα ο δρόμος αυτός είναι προτιμότερος και γι' αυτό τον ακολουθούμε εδώ.

## ΕΛΛΗΝΙΚΟ ΑΛΦΑΒΗΤΟ

ΕΛΛΗΝΙΚΟΣ ΧΑΡΑΚΤΗΡΑΣ	ΠΛΗΚΤΡΟ	ΚΩΔΙΚΑΣ ASCII
Π	A	1
Σ	B	2
Ψ	C	3
Δ	D	4
Θ	E	5
Φ	F	6
Γ	G	7
Ξ	J	10
Λ	L	12
Ω	O	15

Οι χαρακτήρες 0-31 εκτυπώνονται είτε με τη βοήθεια της λέξης EMIT, είτε με τα αντίστοιχα πλήκτρα αν προηγουμένως έχει πατηθεί το "GRAPHICS". Ιδιαίτερα για τους χαρακτήρες 1-26 η αντίστοιχα πλήκτρο-χαρακτήρα (σε GRAPHICS mode πάντα) είναι ιδιαίτερα απλή, αφού στο πλήκτρο A αντιστοιχεί ο χαρακτήρας 1, στο B ο 2, στο C ο 3 κλπ. Οι χαρακτήρες 0 και 13, αν και μπορούν να προγραμματιστούν, χρησιμοποιούνται σαν χαρακτήρες ελέγχου και καλό είναι να μη τους πειράζουμε. Διαφορετική, κάθε φορά που θα αλλάζει γραμμή θα εμφανίζονται στην οθόνη ανεπιθύμητα τα σύμβολα που θα έχουμε αναθέσει στους δυο χαρακτήρες. Επίσης δεν πρέπει να πειράζουμε τους χαρα-

κτήρες 16-23 που περιέχουν τα γραφικά σύμβολα, που χρησιμοποιούνται και από τη λέξη PLOT. Οι χαρακτήρες 1-12, 14-15 και 24-31 αν και περιέχουν αντίγραφα των γραφικών συμβόλων μπορούν ν' αναπρογραμματιστούν χωρίς να χάσουμε κανέναν από τους ενδοκτισμένους χαρακτήρες του Ace. Χρησιμοποιήσαμε 10 από αυτούς για τα ελληνικά γράμματα, κάνοντας την ανάθεση κατά τρόπο λίγο-πολύ τυχαίο. Φυσικά, αν θέλει ο αναγνώστης, μπορεί πάρα πολύ εύκολα να αναθέσει τους χαρακτήρες σε διαφορετικούς κωδικούς (κι επομένως σε διαφορετικά πλήκτρα).

Στο εγχειρίδιο του Ace περιγράφεται ένας τρόπος προγραμματισμού των χαρακτήρων, αλλά οι πληροφορίες τόσο για τη φύλαξη

τους σε κασέτα, όσο και για τη φόρτωση τους από αυτήν στον υπολογιστή δεν είναι όσο πρέπει λεπτομερείς. Κάνοντας μερικές τροποποιήσεις στη μέθοδο που περιγράφει το εγχειρίδιο, οι χαρακτήρες μπορούν να φυλαχτούν και να ξαναφορτωθούν σαν οποιοδήποτε άλλο λεξιλόγιο.

Πρώτ' απ' όλα τυπώσαμε την πολύ χρήσιμη λέξη GR που αναφέρεται στο εγχειρίδιο (σελίδα 71).

```
: GR
8 * 11263 + DUP
8 +
DO
I C! - 1
+ LOOP
;
```

Η GR περιμένει 9 αριθμούς από το σωρό. Αυτός που είναι στην κορυφή αντιπροσωπεύει τον κώδικα ASCII του χαρακτήρα που σχεδιάζουμε. Εδώ πρέπει να σημειώσουμε ότι κάθε χαρακτήρας σχηματίζεται στην οθόνη από 8 σειρές φωτεινών σημείων. Οι επόμενοι 8 αριθμοί που περιμένει η GR περιγράφουν την κατάσταση καθενιάς απ' αυτές τις σειρές (ποια φωτεινά σημεία πρέπει να είναι λευκά και ποια όχι). Αν σε κάθε λευκό σημείο αντιστοιχήσουμε τον αριθμό 1 και σε κάθε μαύρο το 0, βλέπουμε ότι κάθε σειρά περιγράφεται από ένα δυαδικό αριθμό των 8 ψηφίων. Είναι ιδιαίτερα βολικό να προγραμματίζουμε κάποιο χαρακτήρα χρησιμοποιώντας δυαδικούς αριθμούς γιατί έτσι έχουμε άμεση εποπτεία της μορφής του. Έστω λοιπόν αν θέλουμε να φτιάξουμε ένα "γάμα κεφαλαίο". Η αντίστοιχη λέξη είναι η ακόλουθη:

```
: GAMA (2 BASE C!)
00000000
01111110
01000000
01000000
01000000
01000000
01000000
```

01000000  
00000000 (DECIMAL) 7  
GR;

Τυπώνοντας έτσι ακριβώς τη λέξη βλέπουμε καθαρά τους άσσους να σχηματίζουν το Γ. Οι λέξεις [ και ] (που περιγράφονται στη σελίδα 134 του εγχειριδίου) μας επιτρέπουν να επιστρέψουμε προσωρινά αν και βρισκόμαστε στο εσωτερικό του ορισμού μιας λέξης στον interpret mode. Αυτό σημαίνει ότι οτιδήποτε υπάρχει ανάμεσα στις [ και ] εκτελείται άμεσα και δεν αποτελεί μέρος του ορισμού. Στη συγκεκριμένη περίπτωση, οι λέξεις [ 2 BASE C! ] δίνουν στον κομπιούτερ να καταλάβει ότι οι αριθμοί που θα εισαχθούν στη συνέχεια είναι δυαδικόι, ενώ οι [ DECIMAL ] ότι επιστρέφουμε στο δεκαδικό σύστημα αρίθμησης. Ο αριθμός 7 που ακολουθεί είναι ο κωδικός

ASCII που θ'αναθέσουμε στο Γ. Σε GRAPHICS mode αντιστοιχεί στο πλήκτρο G, σύμφωνα με τον κανόνα που αναφέραμε πιο πάνω. Τυπώνοντας τώρα GAMA "ENTER" για να μπει το Γ στη μνήμη χαρακτήρων του Ace. Για να δούμε να εμφανίζεται στην οθόνη τυπώνουμε είτε 7 EMIT "ENTER" είτε "GRAPHICS" G. Τυπώνοντας LIST G "ENTER" παρατηρούμε ότι η λέξη έχει μεταφερθεί στο λεξιλόγιο με την εξής μορφή:

: GAMA 0 126 64 64 64 64 64 0 7 GR;

Δηλαδή όλοι οι δυαδικοί έχουν μετατραπεί σε δεκαδικούς και οι λέξεις που εκτελούνται σε interpret mode έχουν εξαφανιστεί. Ο αναγνώστης μπορεί να εισάγει στον υπολογιστή του τη λέξη απευθείας με αυτή τη μορφή. Στη συνέχεια δί-

νομε τις αντίστοιχες λέξεις για τα υπόλοιπα 9 γράμματα.

Π: : PI 0 126 66 66 66 66 66 0 1 GR;

Σ: : SIGMA 0 126 96 48 48 96 126 0 2 GR;

Ψ: : PSI 0 73 73 42 28 8 8 0 3 GR;

Δ: : DELTA 0 24 36 66 66 66 126 0 4 GR;

Θ: : THTA 0 60 66 126 126 66 60 0 5 GR;

Φ: : FI 0 8 62 42 42 62 8 0 6 GR;

Ξ: : KSI 0 126 0 60 60 0 126 0 10 GR;

Λ: : LAMDA 0 24 36 66 66 66 66 0 12 GR;

Ω: : OMEGA 0 126 66 66 126 0 126 0 5 GR;

Μόλις τυπώσουμε τη λέξη GR και τις 10 λέξεις για τα γράμματα, τυπώνουμε και τη λέξη OLA:

: OLA PI SIGMA PSI DELTA THTA FI GAMA

KSI LAMDA OMEGA 16 1 DO CR I . ." \*\*\*" I EMIT LOOP ;

Η λέξη αυτή εισάγει όλα τα γράμματα στη μνήμη χαρακτήρων. Κατόπιν εκτυπώνει τους χαρακτήρες 1-15 δίνοντας έτσι την επιβεβαίωση ότι όλα έχουν λειτουργήσει σωστά.

Η φύλαξη του ελληνικού αλφαβήτου σε κασέτα γίνεται τυπώνοντας κανονικά SAVE ELN και το φόρτωμά του στον υπολογιστή με το LOAD ELN. Μόλις γίνει αυτό, τυπώνουμε OLA "ENTER". Όταν η OLA μας εμφανίσει στην οθόνη τους χαρακτήρες, μπορούμε ν'απαλλαγούμε από τις περιττές λέξεις και να ελευθερώσουμε τη μνήμη τυπώνοντας FORGET GR "ENTER". Τελειώνοντας δίνουμε την αντιστοίχια κωδικού ASCII, πλήκτρου (με το "GRAPHICS" πατημένο) και ελληνικού χαρακτήρα.

## Jupiter ACE

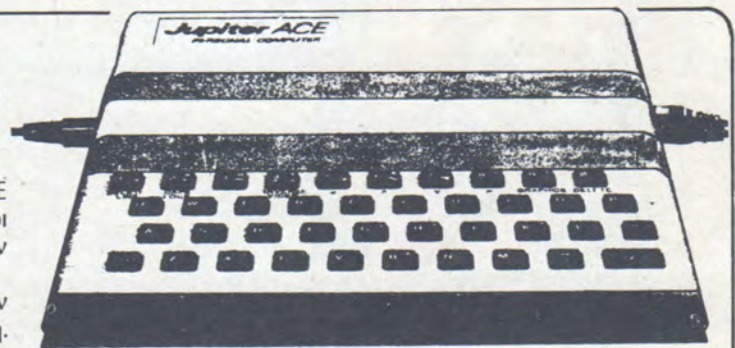
### Ο ΜΙΚΡΟΎΠΟΛΟΓΙΣΤΗΣ ΠΟΥ ΔΕΝ ΘΑ ΧΡΕΙΑΣΤΕΙ ΝΑ ΑΝΤΙΚΑΤΑΣΤΗΣΕΤΕ ΤΟΥ ΧΡΟΝΟΥ !!

Σχεδιασμένος για το μέλλον ο Jupiter-ACE θα ... εκτελεί το καθήκον του όταν οι άλλοι σύγχρονοι του θα βρίσκονται στο καλάθι των αχρήστων.

• Η γλώσσα FORTH που χρησιμοποιεί τον κάνει κατάλληλο για οποιαδήποτε εφαρμογή. Δεν είναι μια τροποποιημένη γλώσσα (όπως συμβαίνει με όλους τους άλλους μικροϋπολογιστές που χρησιμοποιούν τη δική τους παραλλαγή) αλλά ακολουθεί πιστά τη μοναδική τυποποιημένη FORTH-79.

• Το εγχειρίδιο (μεταφρασμένο τώρα και στα ελληνικά) είναι ένα τέλειο βοήθημα για όσους θέλουν να μάθουν FORTH ακόμη και αν δεν έχουν Jupiter-ACE.

• Η διαθέσιμη μνήμη του Jupiter-ACE φθάνει με την επέκταση 48K RAM που ήδη κυ-



κλοφόρησε στην ελληνική αγορά, τα 51K! Ποιός άλλος μικροϋπολογιστής έχει τέτοια δυνατότητα;

• Η βιβλιοθήκη SOFTWARE συνεχώς επεκτείνεται. Υπάρχουν πάνω από 40 τίτλοι με θέματα επιστημονικά, εκπαιδευτικά, utilities, παιχνίδια κλπ. και το σπουδαιότερο: Όλα **δοκιμασμένα** ώστε να φορτώνονται χωρίς προβλήματα.

• Τα αναμενόμενα interfaces για σύνδεση με εκτυπωτή και δισκέττα (πραγματική virtual memory!), τα joysticks, το color PCB, το voice synthesiser κλπ. θα κάνουν τον Jupiter-ACE τον τελειότερο (αν και φθηνότερο) μικροϋπολογιστή της ελληνικής αγοράς.

**ΑΝΤΙΠΡΟΣΩΠΟΣ:**

**Α.ΠΑΠΑΣΩΤΗΡΙΟΥ**

Στουρνάρα 23 - Αθήνα 106 82  
τηλ.: 36.41.826 - 36.09.821



# ΛΙΓΗ ΓΛΩΣΣΑ ΜΗΧΑΝΗΣ ΔΕ ΒΛΑΠΤΕΙ

## SEARCH & REPLACE

(ASCII CODE SEARCH  
ASCII CODE REPLACE- )

RST 24	223		
LD C E	75		
RST 24	223		
LD HL (9216)	33	0	36
LD B (23)	6	23	
PUSH BC	197		
LD B (32)	6	32	
LD A (HL)	126		
CP E	187		
JRNZ (+2)	32	2	
LD A C	121		
LD (HL) A	119		
INC HL	35		
DJNZ (-9)	16	247	
POP BC	193		
DJNZ (-15)	16	241	
JP (IY)	253	233	

## INVERT

( - )			
LD HL (9216)	33	0	36
LD B (23)	6	23	
PUSH BC	197		
LD B (32)	6	32	
LD A (HL)	126		
ADD A (128)	198	128	
LD (HL) A	119		
INC HL	35		
DJNZ (-7)	16	249	
POP BC	193		
DJNZ (-13)	16	243	
JP (IY)	253	233	

Δεν υπάρχει αμφιβολία ότι η FORTH είναι ταχύτερη. Σπάνια θα σας κάνει να νοιώσετε την ανάγκη γλώσσας μηχανής. Έπειτα υπάρχει και η λέξη FAST του Ace που κάνει την εκτέλεση ακόμα ταχύτερη.

Μια (και ίσως η μόνη) περίπτωση που ο κώδικας μηχανής προσθέτει ορατή για τον άνθρωπο βελτίωση ταχύτητας παρουσιάζεται όταν θέλουμε να αλλάξουμε στιγμιαία ολόκληρη την οθόνη. Για να καταλάβετε τι εννοούμε, χρησιμοποιήστε τη λέξη FILL, που γεμίζει την οθόνη με ένα χαρακτήρα αφού πάρει τον ASCII κωδικό του από το σωρό:

```
: FILL
9952 9216
DO
DUP I C!
LOOP
DROP
;
```

Δίνοντας για παράδειγμα 42 FILL η οθόνη θα γεμίσει με αστέρια. Το "γέμισμα" γίνεται ταχύτατα (ιδίως σε FAST mode) αλλά όχι στιγμιαία. Η ρουτίνα SEARCH & REPLACE που δίνουμε στη συνέχεια μπορεί να εκτελέσει την ίδια εργασία με τέτοια ταχύτητα που να νομίζουμε ότι όλοι οι χαρακτήρες εμφανίστηκαν ταυτόχρονα. Είναι γραμμένη σε γλώσσα μηχανής και για να την εισάγουμε στον κομπιούτερ ακολουθούμε την ακόλουθη διαδικασία: Πληκτρολογούμε πρώτα:

```
DEFINER CODE
DOES >
CALL
;
και κατόπιν
CODE SEARCH & REPLACE
```

223 C, 75 C, 223 C, 33 C, 0 C, 36 C, και λοιπά.

Πριν εισάγουμε τον κώδικα μηχανής κάνουμε έναν έλεγχο αν τον έχουμε τυπώσει σωστά. Ένα λάθος μπορεί να μας σβύσει τα πάντα.

Η SEARCH & REPLACE περιμένει από το σωρό δυο κωδικούς ASCII: Του χαρακτήρα που θα αντικαταστήσει και του χαρακτήρα που θα βάλει στη θέση του. Έτσι μπορούμε για παράδειγμα να αλλάξουμε όλα τα "E" και να τα κάνουμε "I" ή οτιδήποτε άλλο, ή έχοντας μια μαύρη οθόνη να την μετατρέψουμε ακαριαία σε άσπρη (ολικό inverse video).

Μια δεύτερη ρουτίνα σε γλώσσα μηχανής είναι η INVERT. Για να την εισάγουμε, τυπώνουμε πάλι:

```
DEFINER CODE
DOES >
CALL
;
```

και κατόπιν

```
CODE INVERT
33 C, 0 C, 36 C, 6 C,
23 C, κλπ.
```

Η INVERT απεικονίζει όλους τους χαρακτήρες της οθόνης σε μορφή inverse video. Μ'αυτή μπορούμε να δημιουργήσουμε εντυπωσιακά εφέ. Δοκιμάστε για παράδειγμα την ακόλουθη λέξη:

```
: BANG
21 1
DO
INVERT I
10 * 60 BEEP
LOOP
;
```

Οι δυο ρουτίνες έχουν άπειρες εφαρμογές. Βάλτε τη φαντασία σας να δουλέψει και φτιάξτε τα δικά σας video games. Καλή διασκέδαση!

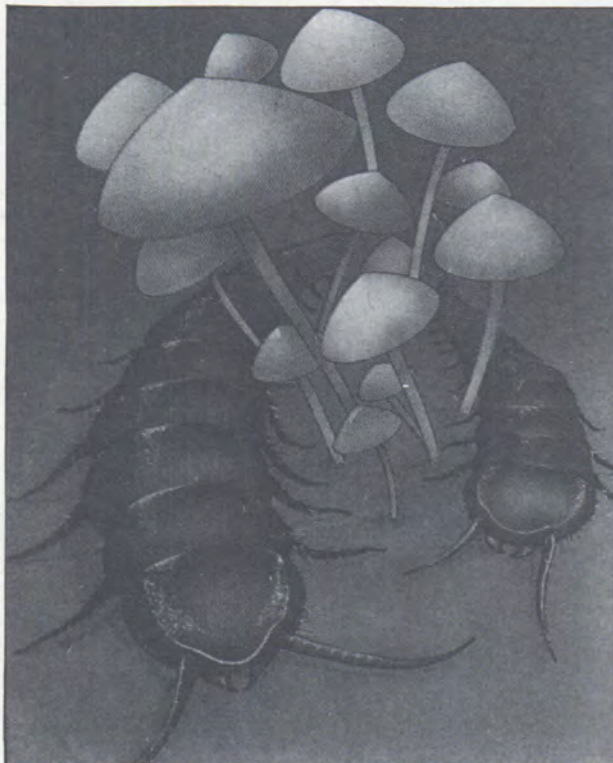
Το πρόγραμμα που ακολουθεί, απαιτεί την επέκταση μνήμης για τον Ace. Πριν αρχίσετε να το πληκτρολογείτε, φορτώστε το ελληνικό αλφάβητο που περιγράφουμε σ' αυτό το τεύχος. Κατόπιν τρέξτε τη λέξη OLA για να οριστούν οι χαρακτήρες, αλλά μη δώσετε FORGET GR.

Κατά την πληκτρολόγηση, προσέξτε ιδιαίτερα τη λέξη ONCE-MORE. Επειδή περιέχει τη λέξη MILLIPEDE που δεν έχει ακόμα οριστεί, τυπώνουμε αρχικά : ONCE-MORE; Μόλις ολοκληρώσουμε την πληκτρολόγηση όλου του προγράμματος δίνουμε: EDIT ONCE-MORE. Κατόπιν τυπώνουμε τη λέξη όπως περιγράφεται στο listing. Τέλος δίνουμε: REDEFINE ONCE-MORE. Αν δεν έχει γίνει κάποιο λάθος στην πληκτρολόγηση, το πρόγραμμα τρέχει όταν εισάγουμε τη λέξη MILLIPEDE.

Το παιχνίδι ΣΑΡΑΝΤΑΠΟΔΑΡΟΥΣΑ παίζεται με τον ακόλουθο τρόπο: Μέσα σ' ένα τούνελ από το πάνω μέρος της οθόνης προς τα κάτω, τρέχει μια σαρανταποδαρούσα. Με το πλήκτρο Z μπορούμε να την κάνουμε να κινηθεί προς τ' αριστερά και με το M προς τα δεξιά. Μέσα στο τούνελ εμφανίζονται ζωφια, που αν τα φάει η σαρανταποδαρούσα κερδίζουμε βαθμούς. Πρέπει όμως να προσέξουμε, γιατί αν χτυπήσει τη μύτη της στα τοιχώματα το παιχνίδι τελειώνει.

Το παιχνίδι έχει επίπεδα δυσκολίας από 2 μέχρι το 10. Όμως, πάνω από το πέμπτο επίπεδο απαιτείται πολύ γρήγορα αντανακλαστικά. Στο δέκατο

## ΣΑΡΑΝΤΑ - ΠΟΔΑΡΟΥΣΑ



επίπεδο, κι ο καλύτερος παίκτης δεν τα καταφέρνει να κρατήσει πάνω από μερικά δευτερόλεπτα.

Το συνολικό σκορ εξαρτάται από το χρόνο που κράτησε το παιχνίδι (VARIABLE S), από τα ζωφια που έφαγε η σαρανταποδαρούσα (VARIABLE G) και από το βαθμό δυσκολίας (VARIABLE H). Συγκεκριμένα, το σκορ καθορίζεται από την τιμή που έχει πάρει το τέλος του παιχνιδιού η έκφραση:  $(S/10+G)*H$ . Οι μεταβλητές A και B καθορίζουν το εύρος του τούνελ. Αν σας φανεί πολύ στενό ή πολύ φαρδύ, μπορείτε ν' αλλάξετε τις αρχικές τιμές τους. Η μεταβλητή M καθορίζει τη διακύ-

μανση των τοιχωμάτων του τούνελ, σε συνδυασμό με το βαθμό δυσκολίας. Τέλος η μεταβλητή P καθορίζει τη θέση που τυπώνεται η μύτη της σαρανταποδαρούσας. Μια πολύ χρήσιμη λέξη, που δε θα τη βρείτε στο εγχειρίδιο του Ace είναι η SCREEN-PEEK. Αυτή περιμένει από το σωρό τη γραμμή και τη στήλη μιας θέσης της οθόνης και επιστρέφει τον κώδικα ASCII του χαρακτήρα που περιέχει η θέση αυτή.

Το πρόγραμμα περιέχει επεξηγήσεις στα ελληνικά σ' όλες τις φάσεις του. Όταν φορτώνετε από το κασετόφωνο, πριν αρχίσετε το παιχνίδι, μην ξεχάσετε να τρέξετε τη λέξη OLA για να σχηματιστούν οι

ελληνικοί χαρακτήρες. Καλή διασκέδαση!

Ø VARIABLE SEED  
Ø VARIABLE A  
Ø VARIABLE B  
Ø VARIABLE G  
Ø VARIABLE H  
Ø VARIABLE M  
Ø VARIABLE P  
Ø VARIABLE S

```
: SETUP 8 A ! 22 B ! Ø
G ! Ø M ! 15 P ! Ø S
;

: LONG-DELAY 2ØØØØ Ø
DO LOOP ;

: SHORT-DELAY 2ØØ Ø DO
LOOP ;

: 2DUP OVER OVER ;

: PRINT AT ." ΣΑΡΑΝΤΑ-
ΠΟΔΑΡΟΥΣΑ" ;

: RIGHT 17 6 DO I DUP
PRINT LOOP ;

: CENTRE 17 6 DO I 11
PRINT LOOP ;

: LEFT 23 Ø DO I DUP
22 SWAP - PRINT LOOP
;

: TITLE 1Ø Ø DO CLS
RIGHT SHORT-DELAY
CLS CENTRE SHORT-
DELAY CLS LEFT
SHORT-DELAY LOOP
CLS CENTRE ;

: HI-SOUND 3Ø 1ØØ BEEP
;

: LOW-SOUND 6Ø 1ØØ
BEEP ;

: PAUSE BEGIN INKEY
UNTIL HI-SOUND LOW-
SOUND ;

: HARDNESS CLS CR CR
." ΔΩΣΕ ΒΑΘΜΟ ΔΥΣΚΟ-
ΛΙΑΣ (2-10)" CR
QUERY LINE DUP DUP
11 < SWAP 1 > AND
IF H ! 1ØØØ Ø DO
LOOP ELSE DROP
HARDNESS THEN ;

: INSTRUCTIONS CLS
." ΠΑΤΗΣΕ ΟΠΟΙΟΔΗΠΟ-
ΤΕ ΠΛΗΚΤΡΟ" CR CR
." ΓΙΑ ΝΑ ΣΥΝΕΧΙ-
ΣΕΙ" CR CR ." ΦΑΓΕ
ΤΑ ΖΩΦΙΑ" CR CR
." ΧΩΡΙΣ ΝΑ ΧΤΥΠΗ-
```

```

ΣΕΙΣ" CR CR ." TH
MYTH ΣΟΥ ΣΤΟ ΤΟΥΝΕΛ"
CR CR ." ΑΡΙΣΤΕΡΑ
'Z'" 8 SPACES ." ΔΕ-
ΞΙΑ 'M'" CR PAUSE ;

: MIL 60 189 102 165
102 165 126 60 8 ;

: BUG 24 60 153 126 60
255 60 219 9 ;

: GRAPH 8 * 11263 +
DUP 8 + DO I C! -1
+LOOP ;

: DEF-GRAPH MIL GRAPH
BUG GRAPH ;

: WALL-PRINT 20 DUP @
AT 144 EMIT B @ AT
144 EMIT ;

: RANDOMISE SEED @ 75
U* 75 0 D+ 2DUP U<
-- 1- DUP SEED ! ;

: RND RANDOMISE U*
SWAP DROP ;

: BUG-PRINT 10 RND 1+
9 > IF 20 A @ DUP B
@ SWAP -2 / + AT 9
EMIT THEN ;

: SCREEN-PEEK AT 15388
@ C@ ;

: ONCE-MORE CLS ." ΘΕ-
ΛΕΙΣ ΝΑ ΞΑΝΑΠΑΙΞΕΙΣ
;" CR CR ." N=NAI
O=OXI" CR BEGIN
INKEY ?DUP UNTIL DUP
110 = IF DROP MILLI-
PEDE ELSE 111 = IF
CLS CR CR ." ΓΕΙΑ
ΣΟΥ" CR LONG-DELAY
ABORT ELSE ONCE-MORE
THEN THEN ;

: ENDIT CLS CR 5 0 DO
." ΧΤΥΠΗΣΕΣ ΤΗ ΜΥΘΗ
ΣΟΥ" CR CR ." ΣΤΟΝ
ΤΟΙΧΟ ΤΟΥ ΤΟΥΝΕΛ" CR
CR LOOP 2 0 DO LONG-
DELAY LOOP CLS 6 0
AT ." ΣΗΜΕΙΩΣΕΣ" S @
10 / DUP ." ΒΑΘ-
ΜΟΥΣ" CR CR ." ΕΦΑ-
ΓΕΣ " G @ DUP ."
ΣΚΑΡΑΒΑΙΟΥΣ" CR CR
." ΣΥΝΟΛΙΚΑ ΕΧΕΙΣ" +
H @ * ." ΠΟΝΤΟΥΣ"
5 0 DO LONG-DELAY
LOOP ONCE-MORE ;

: CHECK-OUT SCREEN-
PEEK 144 = IF ENDIT
THEN ;

: NOSE-SNUB 15 P @
CHECK-OUT 16 P @
CHECK-OUT ;

: G+ G @ 1+ G ! ;

: EAT-BUG 16 P @
SCREEN-PEEK 32 = 0 =
IF G+ LOW-SOUND HI-
SOUND THEN ;

: MIL-PRINT 15 P @ AT
8 EMIT ;

: SCROLL 22 32 AT 32
EMIT CR CR ;

: SUPDATE S @ 1+ S ! ;

: MUPDATE H @ DUP RND
SWAP 2 / - ?DUP IF 3
RND + M ! ELSE
MUPDATE THEN ;

: AUPDATE A @ DUP -1
> IF M @ + THEN A ! ;

: BUPDATE B @ DUP 31<
IF M @ + THEN B ! ;

: A2UPDATE A @ DUP 1<
IF M @ ABS + A ! B @
M @ ABS + B ! ELSE
DROP THEN ;

: B2UPDATE B @ DUP 28
> IF M @ ABS - B ! A
@ M @ ABS - A ! ELSE
DROP THEN ;

: UPDATE SUPDATE
MUPDATE AUPDATE
BUPDATE A2UPDATE
B2UPDATE ;

: MIL-CLEAR 15 P @ AT
32 EMIT ;

: GETKEY INKEY DUP 122
= IF DROP P @ 1 - P !
ELSE 109 = IF P @ 1+
P ! THEN THEN ;

: MULLIPEDE SETUP
TITLE HARDNESS
INSTRUCTIONS DEF-
GRAPH BEGIN WALL
PRINT BUG-PRINT
NOSE-SNUB EAT-BUG
MIL-PRINT SCROLL
UPDATE MIL-CLEAR
GETKEY 0 UNTIL ;

```

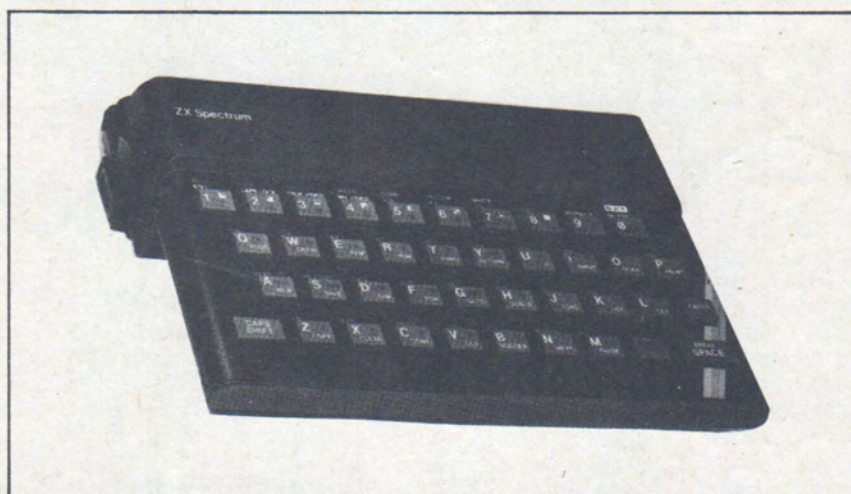




---

Προγράμματα για τον:

# ΣPECTRUM

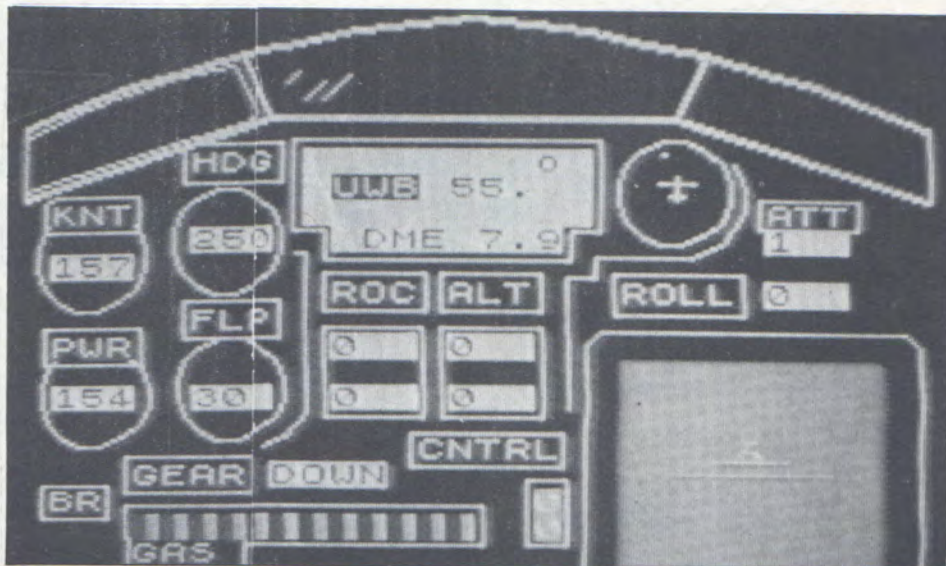
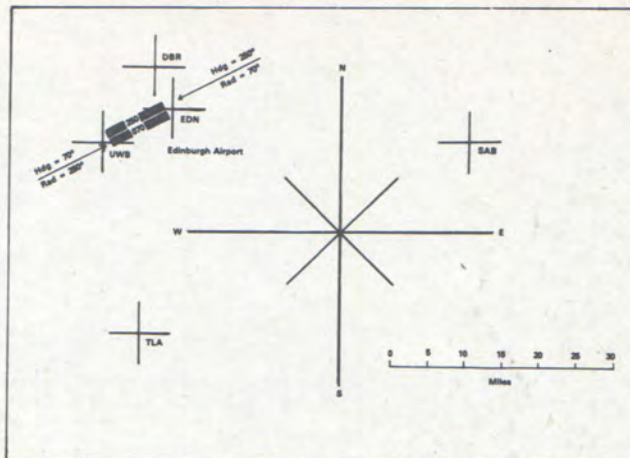


ΕΠΙΜΕΛΕΙΑ:  
Δημήτρης Τσουροπλής

---

Να "πετάξετε" με τα φτερά του SPECTRUM θα σας επιτρέψει το πρόγραμμα αυτό εξομίωσης πτήσης. Όπως βλέπετε πρόκειται για μεγάλο και περίπλοκο πρόγραμμα, που αφήνει ελάχιστη μνήμη ελεύθερη για την έκδοση των 16K RAM. Σαν συνέπεια, θα χρειαστεί πολύωρη και μεθοδική πληκτρολόγηση πριν ο υποψήφιος πιλότος βρεθεί στον αέρα!

Πρώτα απ' όλα πληκτρολογείτε το κυρίως πρόγραμμα του listing 1 και "σώστε" το με την εντολή SAVE "BA 111" LINE 1 ώστε να "τρέχει" αυτόματα μετά το "φόρτωμα". Αφού βεβαιωθείτε ότι η εγγραφή είναι εντάξει σειρά έχει η πληκτρολόγηση του κώδικα μηχανής. Τα bytes μπορούν να εισαχθούν ένα-ένα με την βοήθεια του σύντομου προγράμματος του listing 2. Καθώς οι αριθμοί που εισάγετε, τυπώνονται μαζί με τις διευθύνσεις τους, είναι εύκολο να τους επαληθεύσετε. Αφού τελειώσει η εισαγωγή των bytes, "σώστε" τον κώδικα αμέσως μετά το κυρίως πρόγραμμα, με την εντολή SAVE "m" CODE 23296, 236 και ελέγξτε αν η εγγραφή είναι εντάξει (VERIFY). Αφού γίνουν αυτά, πληκτρολογείτε το πρόγραμμα σχεδίασης της οθόνης (listing 3) και "τρέξτε" το. Μόλις η οθόνη σχεδιαστεί και το πρόγραμμα σταματήσει, δώστε την εντολή SAVE "SC" SCREEN\$ και "σώστε" έτσι την οθόνη αμέσως μετά τις δυο προηγούμενες εγγραφές στην κασέτα. Με τον τρόπο αυτό η οθόνη θα "φορτώνεται" αυτόματα από το κυρίως πρόγραμμα. Για μεγαλύτερη σιγουριά μπορείτε να "σώσετε" σε μια άλλη κασέτα το πρόγραμμα του listing 3 ώστε να μπορείτε να ξανασχεδιάσετε την οθόνη αν χρειαστεί.



## FLIGHT SIMULATION

Ας έρθουμε τώρα σε μερικές οδηγίες πάνω στο πρόγραμμα. Καταρχήν μην περιμένετε να είναι εύκολο το να πετάξει ένα αεροπλάνο (έστω και σε εξομίωση). Οι λειτουργίες των διαφόρων πλήκτρων φαίνονται στον πίνακα 4 και χρειάζεται να εξοικειωθείτε μ'αυτές πριν την πτήση. Με το σύστημα ανίχνευσης που χρησιμοποιείται, κάθε πάτημα αναγνωρίζεται και εκτελείται. Σημειώστε ακόμα ότι μερικά πλήκτρα χρησι-

μούν σαν διακόπτες ON-OFF. Η έλλειψη RAM δεν επιτρέπει στο πρόγραμμα να είναι απρόσβλητο από bugs, αλλά αρκετά περίεργα πράγματα έχουν προβλεφθεί. Για παράδειγμα δεν μπορείτε να χρησιμοποιήσετε φρένα στον αέρα ή να σηκώσετε τις ρόδες στο έδαφος!

Η εικόνα 5 δίνει μια άποψη της περιοχής της πτήσης και η ίδια εικόνα σας δίνεται και στο "μόνιτορ" πάνω στο ταμπλώ με τα όργανα

πιέζοντας το "M" (map). Η πλοήγηση γίνεται βάσει των στοιχείων που σας δίνουν οι ραδιοφάροι και οι πληροφορίες είναι δυο αριθμοί: Η απόσταση από τον ραδιοφάρο σε μίλια (DME) και η γωνία σε μοίρες ως προς τον Βορρά, που έχει το αεροπλάνο ως προς τον ραδιοφάρο. Πιο συγκεκριμένα η γωνία αυτή είναι η γωνία που σχηματίζεται με κορυφή τον ραδιοφάρο, μια πλευρά την ευθεία που περνά από τον ραδιοφάρο και έχει διεύθυνση

0° Βόρειο και δεύτερη πλευρά την ευθεία που περνά από τον ραδιοφάρο και το αεροπλάνο μας. Τα στοιχεία αυτά δίνονται στο πάνω κεντρικό μέρος του ταμπλώ, ενώ δεξιά τους βρίσκεται ένα όργανο στο οποίο φαίνεται η θέση του ραδιοφάρου ως προς την διεύθυνση πορείας του αεροπλάνου (η τελεία που αναβοσβήνει είναι ο ραδιοφάρος). Πιέζοντας το "C" ο ραδιοφάρος που σας δίνει πληροφορίες εναλλάσσεται διαδοχικά μεταξύ αυτών που υπάρχουν στο χάρτη. Οι χρησιμότεροι ραδιογράφοι είναι οι END και UWV γιατί βρίσκονται πλησιέστερα στο αεροδρόμιο.

Το μόνιτορ που αναφέραμε χρησιμεύει κατά τρεις τρόπους. Κανονικά μας παρουσιάζει την θέση του αεροπλάνου ως προς τον τεχνητό ορίζοντα. Δεύτερον μας δίνει το χάρτη της περιοχής και την θέση του αεροπλάνου μας εκεί. (Είναι η κουκίδα μέσα στο τετράγωνο που αναβοσβήνει.) Λάβετε υπόψη ότι όσο ο χάρτης είναι "αναμμένος" η θέση του αεροπλάνου δεν αλλάζει στην απεικόνιση. Τρίτο και τελευταίο βοήθημα από το "μόνιτορ" είναι η απεικόνιση του διαδρόμου κατά την προσγείωση.

Μερικές ακόμα λεπτομέρειες πάνω στο ταμπλώ είναι τα προειδοποιητικά φώτα για φρένα, μη κατεβασμένες ρόδες κατά την προσγείωση, τέλος καυσίμων, κάθοδο του αεροπλάνου (η ROC γίνεται

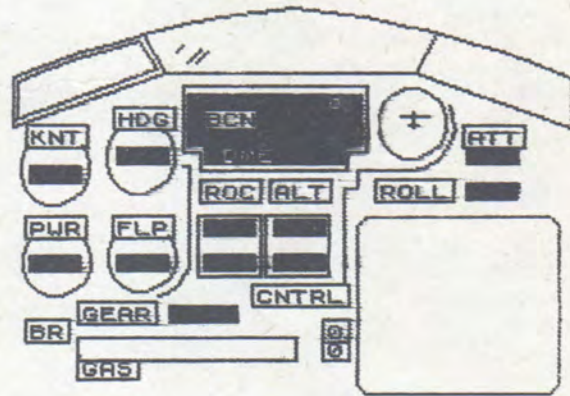
ROD δηλαδή κλίση καθόλου αντί ανόδου) και τέλος κατάσταση εκτός ελέγχου. Τα όργανα κλίσης (ROC) και ύψους (ALT) δείχνουν χιλιάδες πόδια στο πάνω παραθυράκι και πόδια στο κάτω. Η απογείωση γίνεται με ταχύτητα 100 κόμβους και πάνω, το δε υψόμετρο πρέπει να περάσει τα 10 πόδια. Για την προσγείωση οι σωστές γωνίες δίνονται στο χάρτη. Πρέπει επίσης να κατεβαίνετε με την σωστή κλίση, χάνοντας δηλαδή ύψος με ρυθμό 600-800 πόδια το λεπτό (ROD). Ο διάδρομος θα εμφανιστεί στο μόνιτορ μόνο εφόσον η διεύθυνσή σας βρίσκεται σε απόκλιση  $\pm 10\%$  από τη σωστή.

Για να εξασκηθείτε στην προσγείωση μπορείτε να ορίσετε σαν αρχική θέση (define location) την παρακάτω:  
 Beacon ? = EDN  
 Radial ? = 70  
 Heading ? = 250  
 DME ? = 2  
 Αυξήστε την γωνία των FLAPS καθώς πέφτει η ταχύτητα για να διατηρηθεί η άνοση και φρενάρτε αρκετά γρήγορα με την προσγείωση (ένδειξη ALT=10) γιατί ο διάδρομος δεν είναι ατέλειωτος... Τέλος θυμίζουμε ότι οι ενδείξεις heading (HDG) δηλαδή διεύθυνσης ως προς τον ορίζοντα είναι: 0°. Ανατολή 90°, Νότος 180°, Δύση 270°. Για να σας διευκολύνουμε σε τυχόν διορθώσεις σας δίνουμε τα διάφορα τμήματα του προγράμματος και τη λειτουργία τους συνοπτικά στον πίνακα 6.

```

: PRINT TAB 11; INVERSE 1;
16 CIRCLE 20,60,16: CIRCLE 19,
102,16: CIRCLE 59,62,16: CIRCLE
58,107,16: CIRCLE 152,122,16
17 DATA 14,19,1,8,6,14,6,12,1
1,14,11,10,26,6,26,12,15,14,15
20 RESTORE 17: FOR n=1 TO 10:
READ a,b: PRINT AT a,b: INVERSE
1: NEXT n
22 DATA "0",19,18,"0",18,18,"G
AS",20,4,"",19,4,"BR
",18,1,"PWR",12,1,"KNT",7,1,"HDG
",6,1,"FLP",12,6,"GEAR",17,4,"RO
C",10,11,"ROLL",10,21,"ATT",7,26
10,"BCN",6,11,"CNTRL",16,14,"ALT",
10,15
24 RESTORE 22: FOR N=1 TO 16:
READ A$,A,B: PRINT AT A,B;A$: LE
T X=LEN A$*8+4: PLOT (B-1)*8+6,(
22-A)*8+1: DRAW X,0: DRAW 0,-10:
DRAW -X,0: DRAW 0,10: NEXT N
26 PLOT 85,81: GO SUB 100: PLO
T 116,81: GO SUB 100
28 PLOT 150,4: DRAW 4,-4: DRAW
84,0: DRAW 4,4: DRAW 0,72: DRAW
-4,4: DRAW -84,0: DRAW -4,-4: D
RAW 0,-72: FOR N=13 TO 20: PRINT
AT N,21: PAPER 3: NEXT N
30 PRINT INVERSE 1;AT 8,12;"DM
E":AT 5,18;"0":AT 17,9;"
32 PLOT 86,101: DRAW 67,0: DRA
W 0,8: DRAW 9,0: DRAW 0,30: DRAW
-84,0: DRAW 0,-30: DRAW 8,0: DR
AW 0,-8
36 DATA 1,1,255,1,1,1,7,1,0,0,
254,0,0,0,192,0
38 RESTORE 36: FOR n=0 TO 7: R
EAD a: POKE USR "k"+n,a: NEXT n:
FOR n=0 TO 7: READ a: POKE USR
"l"+n,a: NEXT n: PRINT AT 6,22;"
+
40 PLOT 76,103: DRAW 3,0: DRAW
0,-42: DRAW -14,-17,-1,2: PLOT
200,130: DRAW -16,-29,-1,95: DRA
W -24,0: DRAW 0,-8: DRAW -6,0: D
RAW 0,-44
99 STOP
100 DRAW 29,0: DRAW 0,-26: DRAW
-29,0: DRAW 0,26: RETURN

```



listing 3

```

1 REM
5 BORDER 0: PAPER 0: INK 9: C
LS
10 PLOT 0,120: DRAW 70,25: DRA
W 114,0: DRAW 71,-25: DRAW 0,23:
DRAW -127,32,3: DRAW -128,-32,
3: DRAW 0,-23: PLOT 2,125: DRAW
0,16: DRAW 56,19,-2: DRAW 8,-1
3: DRAW -66,-24
12 PLOT 70,145: DRAW -10,18: P
LOT 184,145: DRAW 10,18: PLOT 80
150: DRAW 6,6: PLOT 84,150: DRA
W 5,5: PLOT 75,155: DRAW 3,3
14 FOR n=5 TO 7: PRINT AT n,10
: INVERSE 1: NEXT n

```

ΠΙΝΑΚΑΣ 6 ΑΝΑΛΥΣΗ ΠΡΟΓΡΑΜΜΑΤΟΣ

ΓΡΑΜΜΕΣ

5-240	Κύριος βρόχος προγράμματος
2000-2070	FLAP
2100-2108	Ισχύς κινητήρα
2110-2116	Στροφή
2120-2129	Διεύθυνση καρίνας αεροπλάνου (Attitude)
2500-2515	Τροχοί
2520-2546	Χάρτης
2580-2595	Ανανέωση μεταβλητών
8500-8810	Προαγείωση
9000-9020	Φρένα
9510-9511	Ραδιοφάροι
9520-9522	Εκτός ελέγχου
9530-9545	Τεχνητός ορίζων
9550-9560	Απογείωση
9640-9644	Καύσιμα
9550-9665	Υπορουτίνα αρχής
9680-9682	Μηνύματα στο τέλος της πτήσης

ΠΙΝΑΚΑΣ 4

ΠΛΗΚΤΡΟ ΛΕΙΤΟΥΡΓΙΑ

B	Φρένα ON-OFF
C	Αλλαγή ραδιοφάρου
D/U	Κλίση κάτω-πάνω
R/L	Στροφή δεξιά-αριστερά
J/K	Μείωση/αύξηση των FLAP
P	Αύξηση ισχύος κινητήρα-σταμάτημα μείωσης
O	Μείωση ισχύος - σταμάτημα αύξησης
M	Χάρτης ON-OFF
G	Ρόδες προαγείωσης πάνω-κάτω

listing 2

```
1000>FOR n=1 TO 236
1010 PRINT 23295+n,
1020 INPUT "Enter byte",i: POKE
23295+n,i: PRINT i
1030 NEXT n
```

175,33,129,92,54,1,33,20,  
79,6,7,197,229,6,8,14,  
11,36,229,119,35,13,32,251,  
225,16,244,225,17,32,0,25,  
193,16,232,33,129,92,190,200,

119,6,3,33,180,71,24,219,  
32,33,34,193,72,32,10,207,  
64,18,18,33,72,32,15,139,  
72,22,12,203,72,22,10,6,  
72,27,16,90,72,7,14,198,  
72,27,10,143,72,18,12,207,  
72,18,10,16,72,17,16,62,  
12,50,145,92,6,11,42,75,  
92,17,0,0,25,221,33,51,  
91,197,229,33,131,92,6,2,  
221,126,0,221,35,35,119,221,  
126,0,221,35,35,119,35,35,  
16,238,225,6,3,126,221,229,  
229,197,215,193,225,221,225,35,  
16,243,193,16,212,175,50,145,  
92,201,0,0,0,0,0,  
0,1,255,16,124,16,16,124,  
198,69,255,0,0,0,0,0,  
0,0,254,0,253,203,2,134,  
33,163,90,34,54,92,33,24,  
80,34,132,92,33,9,8,34,  
136,92,1,3,0,17,48,91,  
205,60,32,33,24,90,6,3,  
54,31,35,16,251,33,0,60,  
34,54,92,201,0,0,0,0,

listing 1

```
4>LOAD ""CODE: BORDER 0: PAP
ER 0: INK 7: CLEAR: LOAD ""SCRE
EN#: DIM s$(11,3): GO SUB 9550
6 INPUT "TAKE-OFF? (Y/N) ";X
#: IF X$="Y" THEN GO TO 9550
8 INPUT "OK...STARTING ALTITU
DE? ";AL: LET AL=AL*P: INPUT "W
IND DIRECTION? ";WD: LET WD=WD*R
AD: INPUT "WINDSPEED? ";WS: LET
WS=WS/2E3: INPUT "DEFINE LOCATIO
N? (Y/N)";X$: IF X$<>"Y" THEN GO
TO D*P
25 INPUT "BEACON? ";X$: FOR Q=
T TO 7: IF X$=M$(Q) THEN GO TO 2
8
26 NEXT Q: GO TO 25
26 LET B=0-T: LET B=B+7*(B=0):
GO SUB 9510: INPUT "RADIAL? ";
RD: LET RD=RD*RAD
30 INPUT "HEADING? ";HD: LET
HD=HD*RAD: INPUT "DME? ";DM: LE
T S=DM*COS RD: LET W=DM*SIN RD
40 LET S=S+SF*COS HD+WS*COS WD
: LET W=W+SF*SIN HD+WS*SIN WD: L
ET DM=SQR (S*S+W*W): LET RD=ACS
(S/DM): IF W<0 THEN LET RD=Y*PI-
RD
50 LET HD=HD+AL/(SF*1.5E4): IF
HD>DEG>360 THEN LET HD=HD-Y*PI
55 IF HD<0 THEN LET HD=HD+Y*PI
60 IF C AND DM<6 THEN LET B=T:
GO TO EX
65 LET G=G-(PR+20)/M1: IF G<50
0 THEN GO SUB 9644-M1
70 LET G$=A$( TO G/1E3)
80 IF C THEN LET AL=0: LET RC=
0: LET E=PR: LET KN=KN+E: GO TO
105
90 LET I=2E4/KN: IF J<>I AND N
OT M THEN GO SUB 9530: LET J=I
102 LET RC=AT*KN/60-ST*20: LET
E=PR-(FL*KN/4E3)-(GR*KN/400)-RC/
P: LET KN=KN+E*2E3/KN*(KN<4E3 OR
SGN E=-1): IF KN<(1.5E3-FL*P) T
HEN GO SUB 9520: GO TO 108
103 LET ST=0: POKE 23054,7: GO
```

```

TO 108
105 IF AT>8 THEN LET C=0: GO TO
102
111 LET SF=KN/25E3: PRINT AT 19
,D: PAPER Y;G$: LET A1=0: LET AL
=AL+AC*3/Y: LET FS=KN/2E4: GO SU
B 2580
146 RANDOMIZE USR 23391: GO SUB
9500: IF TH THEN GO SUB 2095+TH
*85
20200 IF PEEK 23560<65 THEN GO TO
20216
20210 LET SI=PEEK 23560-64: IF SI
>0 AND SI<22 THEN GO SUB S(SI)
20215 POKE 23560,0
20216 IF C THEN GO TO 40
20229 IF AL<5 THEN LET B=Y: GO TO
EX
2030 IF B>Y OR ABS S>.4 OR ABS W
>.4 THEN GO TO 40
2036 IF B=T AND (HD<4.19 OR HD>4
.64) THEN GO TO 40
2038 IF B=Y AND (HD<1.05 OR HD>1
.4) THEN GO TO 40
2040 GO TO 8500
20000 IF FL<30 THEN LET FL=FL+5
20005 GO TO 2015
20010 IF FL>0 THEN LET FL=FL-5
20015 LET S$(8)=STR$ FL: RETURN
20070 RETURN
2100 IF TH=Y THEN LET TH=0: RETU
RN
2102 IF PR<42 THEN LET PR=PR+D:
LET TH=T: RETURN
2105 IF TH=T THEN LET TH=0: RETU
RN
2107 IF PR>-20 THEN LET PR=PR-D:
LET TH=Y: RETURN
2108 RETURN
2110 IF LA THEN LET C=C+Y: GO TO
2117
2111 IF NOT C THEN LET RL=RL+P*(
RL<60)
2112 GO TO 2116
2113 IF LA THEN LET C=C-Y: GO TO
2117
2115 IF NOT C THEN LET RL=RL-P*(
RL>-60)
2116 GO SUB 9530: RETURN
2117 LET HD=HD+C/65
2120 IF C=T OR AT<-D THEN RETURN
2122 LET AT=AT-T: GO TO 2127
2125 IF (KN<1E3 AND NOT LA) OR A
T>10 THEN RETURN
2126 LET AT=AT+T
2127 PRINT AT 8,26; INVERSE T; AT
+T: " IF M OR LA THEN RETURN
2129 GO SUB 9530: RETURN
2500 IF GR AND NOT C THEN PRINT
AT 17,P-T; INVERSE T;"UP ": LET
GR=0: RETURN
2510 IF NOT GR AND (NOT C OR M=4
00) THEN PRINT AT 17,9; INVERSE
T;"DOWN": POKE 23076,7: LET GR=T
: RETURN
2515 RETURN
2520 IF M THEN GO TO 2545
2521 LET M=T: POKE 23560,0: REST
ORE 9550: FOR Q=0 TO 7: PRINT AT
13+Q,21; PAPER T;" ": N
EXT Q
2525 PRINT PAPER 1 AT 20,25;"t1a
" AT 18,22;"fkk" AT 16,21;"uwb"
AT 16,25;"eda" AT 18,27;"sab" AT
13,25;"pth" AT 15,23;"dbr"
2530 LET Q1=198: LET Q2=40
2535 FOR Q=0 TO 6: PLOT Q1,Q2
2537 IF B-T=0 THEN LET P1=Q1: LE
T P2=Q2: LET P1=P1+W/1.2: LET P2
=P2+S/1.2: IF P1<255 AND P2<78 P
ND P2>-T AND P1>160 THEN PLOT FL
ASH T,P1,P2: PRINT P3=T
2540 READ M$(Q+T),J,I: LET Q1=Q1
-1/1.2: LET Q2=Q2-U/1.2: NEXT Q:
RETURN
2545 IF P3=T THEN LET P3=0: PLOT
OVER T; FLASH O,P1,P2
2546 LET M=0: FOR Q=0 TO 7: PRIN
T AT 13+Q,21; PAPER 3; FLASH O;"
": NEXT Q: GO TO 9535
2580 LET AL=AL/P: LET AL1=INT (A
L/1E3)+T*(SGN AL=-T): LET S$(9)=
STR$ AL1: LET AL1=INT (AL-AL1*1E
3): LET S$(P)=STR$ AL1: LET AL=A
L*P
2583 LET S$(3)=STR$ (KN/P)
2584 LET S$(Y)=STR$ (DEG*RD)
2585 LET S$(6)=STR$ (DEG*HD)
2586 LET S$(7)=STR$ AL
2587 LET S$(11)=STR$ DM
2588 LET S$(T)=STR$ (PR+110)
2589 LET S$(D)=STR$ (INT (RC/100

```

```

)+T*(SGN RC=-T)): LET S$(5)=STR$
(INT (P*RC-VAL S$(D)*1E3))
2592 IF SGN RC=-T THEN LET S$(5)
=-S$(5,Y TO 3)+"0": PRINT FLASH T
AT P,13;"D": FLASH O: RETURN
2594 PRINT AT P,13;"C"
2595 RETURN
2596 LET G=-T: PRINT AT P,15;"HG
T": LET S$(D)=" ": LET S$(5)="
": LET LA=T: LET E1=E
2595 LET S(Y)=9E3: LET S(3)=2070
2596 LET n=P*P: LET s=-N: LET e=
22E3: LET w=13E3: LET a=50: LET
21=AL/P-RND*P
2570 LET C=(HD*DEG-250+100*(B=Y)
)*5
2610 LET Z=0: LET X=204: LET K=3
5: LET J=D*P: LET M=J*P: FOR Q=0
TO 7: PRINT AT 13+Q,21;"
": NEXT Q: IF NOT GR THEN POKE
23076,164
2640 LET F=KN/35+A: LET E=E-F: L
ET U=W-F: LET S=S-C/P: LET N=S+2
00: IF AL>8 THEN LET AL=AL+D*AT:
GO TO 8645
2643 LET AL=0: PRINT AT P,13;"C"
: LET S$(D)=" ": LET S$(5)="
"
2645 IF KN<100 THEN LET KN=0
2650 LET R=M*N/U-C+X: LET L=M*S/
U-C+X: LET V=M*N/E-C+X: LET I=M*
S/E-C+X: LET U=-M*AL/W-Y*AT+K: L
ET T1=-M*AL/E-Y*AT+K
2665 IF ABS (R-X)>J THEN LET R=S
GN R*J+X
2670 IF ABS (U-X)>J THEN LET U=S
GN U*J+X
2675 IF L<X-J THEN LET L=X-J
2676 IF L>X+J THEN LET L=X+J
2680 IF ABS (I-X)>J THEN LET I=S
GN I*J+X
2685 IF ABS (T1-K)>K THEN LET T1
=SGN T1*K+K
2690 IF ABS (U-K)>K THEN LET U=S
GN U*K+K
2695 IF U>=35 AND W<M THEN LET U
=3: LET T1=35
2710 RANDOMIZE USR 23296: PLOT 1
58,D: DRAW D,-D: DRAW 84,0: DRAW
D,D
2715 PLOT R,U: DRAW L-R,0: PLOT
L,U: DRAW I-L,T1-U: PLOT I,T1: D
RAW U-I,0: PLOT U,T1: DRAW R-U,U
-T1
2721 IF E1>2E3 THEN GO TO 2723
2722 IF NOT AL AND C<Y THEN PRIN
T AT 16,21;" "
2724 IF NOT AL THEN LET RC=AL: G
O TO 2730
2725 LET RC=Y*AT*KN/120: LET E1=
PR-FL*KN/4E3-GR*KN/500-RC/P: LET
KN=KN+E1*2E3/KN
2730 IF AL>8 THEN LET AL=AL*P: G
O SUB 2580: LET AL=AL/P: GO TO 8
734
2732 GO SUB 2583
2734 LET S$(Y)=" ": LET S$(11)
=" "
2735 RANDOMIZE USR 23391
2736 IF G=T THEN GO SUB 9010
2737 IF TH THEN GO SUB 2095+TH*5
2741 IF AL<20 THEN GO TO 2750
2742 IF KN<(1.5E3-FL*P) THEN GO
SUB 9520: GO TO 2747
2745 POKE 23054,7: LET ST=0
2750 IF PEEK 23560<65 THEN GO TO
2790
2760 LET SI=PEEK 23560-64: IF SI
>0 AND SI<22 THEN GO SUB S(SI)
2770 POKE 23560,0
2790 IF W<X THEN LET W=X: IF E<2
E3 THEN GO TO 8630
2800 IF AL<20 AND NOT Z THEN GO
TO 9570
2810 GO TO 8640
2830 IF AL>P OR KN THEN LET B=T:
GO TO EX
2840 LET B=D: GO TO EX
2900 LET G=-G: POKE 23105,7+177*
(G=T)
9010 IF AL>P THEN RETURN
9018 IF KN>100 THEN LET KN=KN-10
0: RETURN
9020 LET KN=0: RETURN
9500 LET R=RD+PI: LET A1=HD-R: P
LOT OVER T;X3,Y1: DRAW OVER T;T,
0
9501 POKE 23200,0
9502 LET X3=182-12*SIN A1: LET Y
1=122+12*COS A1: PLOT X3,Y1: DRA
W T,0: RETURN
9510 LET S=S+A(B): LET U=W+B(B):
LET B=B+T: IF B=8 THEN LET B=T

```

```

9511 PRINT AT 5,11;M$(B): RETURN
9520 IF C OR AL<50 THEN RETURN
9521 LET ST=ST+T: POKE 23054,184
: IF ST<20 THEN RETURN
9522 LET B=D+T: GO TO EX
9525 DATA "OUT OF RNWY..", "NOSE
DIVE..", "NICE WORK CAP'N!", "O
UT OF CONTROL..", "IMPACT BEFORE
RNWY", "CRASH R OF RNWY", "IMPACT
L OF RNWY", "UNDERCARRIAGE?", "LAN
DING TOO STEEP", "OUT OF FUEL.."
9530 IF M THEN RETURN
9531 PLOT OVER T;180,AH: DRAW OV
ER T;46,AY
9535 LET AH=26+I-AT-RL/2: LET AY
=RL
9540 RANDOMIZE USA 23483
9542 IF AH<9 THEN LET AH=9
9545 PLOT 180,AH: DRAW 46,AY: RE
TURN
9550 GO SUB 9651: LET W=-.8: LET
S=0: LET FL=P: LET C=T: LET KN=
T/P: LET I=D: LET GR=T: LET AT=0
: LET RL=AT
9551 PRINT AT 17,P-T; INVERSE T;
"DOWN"
9552 LET PR=0: PRINT INVERSE T;A
T 14,T;"
9553 LET WS=0: LET WD=0: LET B=T
: GO SUB S(3)
9554 LET HD=250*PI: LET SF=1E-1
0
9555 GO TO D*P
9570 IF E>9E3 THEN LET B=D+Y: GO
TO EX
9575 IF N<20 THEN LET B=7: GO TO
EX
9580 IF N>180 THEN LET B=D+D: GO
TO EX
9585 IF NOT GR THEN LET B=9: GO
TO EX
9590 IF AT<-3 THEN LET B=P: GO T
O EX
9625 LET Z=T: LET A=120: GO TO 8
640
9640 LET G=12490: LET M1=Y: PRIN
T AT 21,D: FLASH T;"F": RETURN
9644 LET B=P+T: GO TO EX
9650 LET P=10: DIM S$(12): LET E
X=9680: LET C=0: LET O=C: LET AT
=C: LET P3=C

```

```

9651 LET LA=0: LET FL=0: LET TH=
0: LET M=0: LET ST=0: LET X3=0:
LET Y1=0: LET RC=0: LET PR=0: LE
T RL=0: LET AT=0: LET D=4: LET E
=1: LET Y=E+E: LET J=Y+D: LET I=
J: LET M1=0: LET GR=E: LET DEG=1
80/PI: LET RAD=1/DEG: LET G=1249
0: LET A$="": POKE 2
3658,D+D: LET B=INT (RAND*6)+E
9652 LET T=E: LET SF=.05: LET KN
=2500: LET S=RND*P: LET W=-RND*P
: LET HD=RAD*(RND*360)
9654 RESTORE 9660: DIM A(7): DIM
B(7): DIM M$(7,3): FOR Q=T TO 7
: READ M$(Q),A(Q),B(Q): NEXT Q
9655 DIM S(21): FOR Q=T TO 21: L
ET S(Q)=2078: NEXT Q: LET S(3)=9
510: LET S(D)=2120: LET S(7)=2500
0: LET S(P)=2010: LET S(P+T)=2E3
0: LET S(P+Y)=2113: LET S(13)=262
0: LET S(15)=2105
9656 LET S(16)=2100: LET S(18)=2
110: LET S(21)=2125
9657 FOR Q=0 TO D+D: PRINT AT 13
+Q,P+P;"": NEXT Q: PL
OT 158,D: DRAW Q,-D: DRAW 64,0:
DRAW D,D: FOR Q=0 TO D+D-T: PRIN
T AT 13+Q,P+P+T; PAPER Y+T;"":
NEXT Q: PRINT AT P+P+T,0
;"
9658 PLOT 30,7: DRAW 26,0
9659 DATA "EDN",4.9,6.4,"UWB",-
1,-50.2,"SAB",27.6,44.38,"TLA",-
63.1,.8,"PTH",46.30.8,"FKK",-22.
4,-32.3,"DBR",5.1,-1.9
9661 GO SUB 9535: GO SUB S(3): G
O SUB 2500
9662 PRINT AT 8,26; INVERSE T;"1
": IF NOT AT THEN RETURN
9665 LET AT=0: GO TO D+Y
9680 RESTORE 9625: FOR Q=T TO 8:
READ X$: NEXT Q: PRINT AT P,16;
"ALT",AT 21,0; FLASH T;X$: INPUT
"ANOTHER GO? ";X$: IF X$="Y" TH
EN LET AT=T: LET C=0: PLOT OVER
T;X3,Y1: DRAW OVER T;T,0: POKE 2
3105,D+D-T: POKE 23054,7: POKE 2
3076,7
9681 IF X$="Y" THEN LET P3=Y: GO
TO 9651
9682 PRINT AT 21,0; INVERSE T;"
ROGER AND OUT": STOP

```

ΜΙΚΡΟΪΠΟΛΟΓΙΣΤΕΣ  
**SINCLAIR**  
**ZX 81, ZX SPECTRUM**  
**IBM PC, MEMOTECH**



MICROCOMPUTER PROGRAMS  
& SYSTEMS, **ΠΟΛΥΤΕΧΝΕΙΟΥ 47**  
(ΠΡΟΕΚΤ. ΤΣΙΜΙΣΚΗ) 2ος ΟΡΟΦΟΣ  
**ΘΕΣ/ΝΙΚΗ ΤΗΛ. (031) 540246**

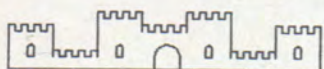
ΣΤΗ ΘΕΣΣΑΛΟΝΙΚΗ ΚΑΙ Σ' ΟΛΗ ΤΗ Β. ΕΛΛΑΔΑ

**ΜΙΚΡΟΪΠΟΛΟΓΙΣΤΕΣ:** ΑΠΟ 25.000 ΔΡΧ  
**ΠΕΡΙΦΕΡΕΙΑΚΑ:** PRINTERS, ΜΝΗΜΕΣ  
ΠΛΗΚΤΡΟΛΟΓΙΑ, INTERFACES (RTTY  
ZX LPRINT, KEMPSTON), DIGITAL  
TRACER, JOUSTICKS κ.ά  
**ΠΡΟΓΡΑΜΜΑΤΑ:** ΕΚΠΑΙΔΕΥΤΙΚΑ, ΕΠΕ-  
ΞΕΡΓΑΣΙΑΣ ΚΕΙΜΕΝΩΝ, ΑΡΧΕΙΩΝ, ΕΙ-  
ΚΟΝΩΝ, ΕΠΑΓΓΕΛΜΑΤΙΚΑ (ΑΠΟΘΗΚΗ,  
ΠΕΛΑΤΕΣ, ΑΡΧΕΙΟΘΗΚΗ, ΜΙΣΘΟΔΟΣΙΑ  
ΠΟΛΙΤΙΚΟΥ ΜΗΧ/ΚΟΥ, ΤΟΠΟΓΡΑΦΟΥ  
ΜΗΧ/ΚΟΥ, ΘΕΡΜΟΜΟΝΩΣΗΣ ΚΑΙ ΠΕΡΙΣ-  
ΣΟΤΕΡΑ ΑΠΟ 50 ΠΑΙΧΝΙΔΙΑ  
**ΓΛΩΣΣΕΣ ΠΡΟΓΡΑΜΜΑΤΙΣΜΟΥ:** BASIC  
FORTH, PASCAL, MACHINE CODE,  
BASIC COMPILERS.  
**UTILITIES:** ZX TOOLKIT, ZX SLOW  
LOADER, ASSEMBLER / DISSASSEMBLER

ΓΙΑ ΟΠΟΙΑΔΗΠΟΤΕ ΠΛΗΡΟΦΟΡΙΑ  
ΕΠΙΣΚΕΦΤΕΙΤΕ ΜΑΣ Η ΤΗΛΕΦΩΝΗΣΤΕ

Ένα πρόγραμμα πιο ρομαντικό από τα συνηθισμένα, επιχειρεί να μας μεταφέρει στην εποχή των ιπποτών - έστω και με ηλεκτρονικά μέσα. Εσείς αντιπροσωπεύετε στο παιχνίδι από τον άσπρο ιππότη ενώ ο κομπιούτερ ελέγχει τον μαύρο. Στην αρχή πρέπει να διαλέξετε τον εξοπλισμό σας για την μονομαχία. Έχετε μόνο 100 χρυσά νομίσματα διαθέσιμα οπότε δεν μπορείτε ταυτόχρονα να έχετε τον καλύτερο οπλισμό και πανοπλία.

Το ρόπαλο (mace) είναι το ισχυρότερο επιθετικό όπλο ενώ το σπαθί (sword) είναι το ασθενέστερο. Εν τούτοις το σπαθί χρησιμεύει και για άμυνα ενώ ο πέλεκυς (axe) όχι. Μόλις κάνετε την επιλογή σας, αναγγέλεται και ο εξοπλισμός που διάλεξε ο αντίπαλος και μετά αρχίζει η μάχη. Στην αρχή οι ιππότες είναι έφιπποι, μέχρις ότου κάποιος



ρίξει τον άλλον στο έδαφος με κτύπημα. Μόλις ένας ιππότης πέσει από το άλογο, ο άλλος θα κατέβει και η μάχη θα συνεχιστεί χωρίς άλογο. Επιτίθεστε στον αντίπαλο μετακινώντας τον δικό σας ιππότη αριστερά-δεξιά με τα "5" και "8" και τον κτυπάτε πιέζοντας ένα πλήκτρο από το 1-4 ανάλογα με την δύναμη που θέλετε να έχει το κτύπημα. Ο αριθμός που διαλέγετε για κτύπημα



# Η ΜΟΝΟΜΑΧΙΑ ΤΩΝ ΙΠΠΟΤΩΝ

αφαιρείται από τη "ζωτικότητα" σας που είναι ο αριθμός που βρίσκεται κάτω-αριστερά στην οθόνη. Όταν ο αριθμός αυτός φτάσει το 0 (είτε για σας, είτε για τον αντίπαλο) ο αντίστοιχος ιππότης πεθαίνει. Απομακρυνόμενος από τον αντίπαλο μπορείτε σταδιακά να αποκτήσετε το μέγιστο της ισχύος σας.

Η μονομαχία λοιπόν συνίσταται στο να τρέχετε πάνω στον αντίπαλο, να τον κτυπάτε και μετά να απομακρύνετε ώστε να ανακτήσετε τις δυνάμεις που χάνετε. Είναι ουσιαστικό να προσέχετε την ισχύ σας για να ξέρετε πότε να υποχωρείτε.

Στο ίδιο το πρόγραμμα δεν πρέπει να συναντήσετε δυσκολία και σας θυμίζουμε ότι όλα τα κεφαλαία γράμματα μέσα σε εισαγωγικά είναι οι αντίστοιχοι graphics χαρακτήρες, που θα πάρουν την κανονική τους μορφή αφού το πρόγραμμα "τρέξει". Αν λοιπόν δείτε κάποιον ιππότη καβάλα σε γράμματα του αλφαβήτου... κάπου ξεχάσατε να περάσετε σε graphics mode!

```

1 REM Knight Fight
10 PAPER 4: BORDER 4: CLS : GO
SUB 7000
100 INK ec: IF eh=1 THEN PRINT
AT y-1,x-1;" US
110 PRINT AT y,x;"ABC ": IF x/2
=INT (x/2) THEN PRINT AT y+1,x;"
DEF "
120 IF x/2<>INT (x/2) THEN PRIN
T AT y+1,x;"GHI "
130 INK ac: IF ah=1 THEN PRINT
AT y-1,29-x;" TU"
140 PRINT AT y,28-x;" LKJ": IF
x/2=INT (x/2) THEN PRINT AT y+1,
28-x;" ONM"
150 IF x/2<>INT (x/2) THEN PRIN
T AT y+1,28-x;" RQP"
160 LET x=x-1: BEEP .003,10: PA
USE 2: BEEP .003,5: PAUSE 3: BEE
P .003,0: PAUSE 5
170 IF x=16 AND ec=7 THEN GO TO
200
180 IF x<2 THEN GO TO 600
190 GO TO 100
200 LET ad=le+INT (RND*6)-ar-(a
r=2)-INT (RND*6): LET ed=la+INT
(RND*4)-er-(er=2)-INT (RND*4)
210 IF ad<=0 AND ed<=0 THEN GO
TO 100
220 IF ad>0 AND ed>0 THEN GO TO
500+100*(ad>ed)
230 IF ad>0 THEN GO TO 600
240 GO TO 500
500 PRINT AT y-1,x-4;" T/\": BE
EP .1,20: BEEP .1,30
510 PRINT AT y-1,x-4;" ": BEEP
.1,20: BEEP .1,10
520 RESTORE 9100: FOR f=0 TO 1:
FOR g=0 TO 7: READ a: POKE USA
CHR# 163+g,a: NEXT g: PRINT AT y
+1,x-6+f;CHR# 163: NEXT f
530 PRINT " ";AT y,x+3;"
";AT y,28-x;" "
540 LET x=x-1: LET ex=x+9: LET
y=y-1: LET ah=0: LET em=em-INT (
RND*6)-1: LET se=se-8: GO TO 100

```

```

600 PRINT AT y-1,x;"^/5": BEEP
.1,50: BEEP .1,30
610 PRINT AT y-1,x;" " : B
EEP .1,50: BEEP .1,30
620 INK 7: RESTORE 9100: FOR f=
0 TO 1: FOR g=0 TO 7: READ a: PO
KE USR CHR# 162+g,a: NEXT g: PRI
NT AT y+1,x+4+f;CHR# 162: NEXT
f
630 PRINT AT y+1,x-4;" " "
;AT y,x+3;" " ;AT y,28-x;" "
640 LET x=x-1: LET ex=x+9: LET
y=y-1: LET eh=0: LET em=em-INT (
RND*4): LET sa=sa-5: GO TO 100
800 IF eh=1 AND ah=1 THEN GO TO
900
810 LET x=20: LET ex=10: LET y=
y+1
820 IF eh=1 THEN LET x=5
830 IF ah=1 THEN LET ex=25
840 GO TO 1000
900 PRINT AT y-1,x;" " ;AT y-
1,28-x;" "
910 LET x=26: LET y=10: LET dc=
ac: LET ac=ec: LET ec=dc: GO TO
1000
1000 PRINT AT y-1,0;" " ;TAB 31;"
"; INK 7;AT y,0;" LKJ"; INK 0;T
AB 20;" ABC"; INK 7;AT y+1,0;" T
NM"; INK 0;AT y+1,20;" GHI"
1010 FOR f=1 TO 2: PRINT AT y-f,
0;" " ;AT y-f,27;" " : NEXT
f
1020 RESTORE 9050: FOR f=1 TO 2:
FOR g=0 TO 7: READ a: POKE USR
CHR# (143+f)+g,a: NEXT g: NEXT f
1030 RESTORE 9100: FOR f=3 TO 12
: FOR g=0 TO 7: READ a: POKE USR
CHR# (143+f)+g,a: NEXT g: NEXT
f
1040 PRINT AT y+1,5;" "
1100 PRINT INK 0;AT y,ex;v$;"A "
;AT y+1,ex;" E "; INK 7;AT y,x-1
;" B";v$;AT y+1,x-1;" F "
1110 IF ex>x+2 THEN PRINT AT y,e
x-1;" " ;AT y,x+3;" "
1120 LET x=x+(INKEY#="8")-(INKEY
#="9")+(x<6)-(x>25): LET x=x-(x>
ex-2)
1130 LET ex=ex+5GN (- (ex>x+1 AND
RND<.5)+ (ex<5)-(ex>25)+(RND<.3)
+(se<3 AND ex<25)): IF ex<x+2 TH
EN GO TO 1400
1200 LET a$=INKEY$: IF a$<"1" OR
a$>"4" THEN LET a$="0"
1210 LET ez=INT (RND*4)+1: IF se
<ez THEN LET ez=se-1
1220 IF se<3 THEN LET ez=0
1230 IF a$="0" THEN GO TO 1300
1240 LET sa=sa-VAL a$: IF sa<1 T
HEN GO TO 4000
1250 IF VAL a$+wa+(RND*7)+1>ez+e
se+er-1+(we=1) THEN LET em=em-1
1260 IF em<1 THEN GO TO 4500
1300 BEEP .03,45: LET se=se-ez:
IF se<1 THEN GO TO 4500
1310 IF ez+we+INT (RND*3)>VAL a$
+sh+ar+(wa=1) THEN LET am=am-1
1320 IF am<1 THEN GO TO 4000
1330 IF sa>am THEN LET sa=am
1340 IF se>em THEN LET se=em
1400 PRINT AT 20,0;sa;" " ;AT 20,
20;se;" " : IF ex-x<3 THEN GO TO
1100
1410 FOR f=1 TO 2: IF sa<am THEN
LET sa=sa+1: NEXT f
1420 IF se<em THEN LET se=se+1
1430 GO TO 1100
4000 PRINT EXP 0;AT y,x;" " ;v$
;"A " ; INK 7;AT y+1,x;" CD"; INK
0;" " ;E
4010 PRINT AT 20,0;"You die.": G
O TO 9200
4500 PRINT INK 7;AT y,x;" B";v$
;" " ;AT y+1,x;" F "; INK 0;"CD
";
4510 PRINT AT 20,20;"He dies.":
GO TO 9200
7000 PRINT "Knight Fight"
7050 RANDOMIZE : LET ah=1: LET e
h=1: LET x=26: LET y=10
7060 LET cash=100: LET sh=0: LET
ac=0: LET ec=7
7100 PRINT "(1) Chain-mail cost
$ 40 coins, " "(2) plate-mail 50.
"
7110 PRINT "(1) Light lances c
ost 10 " "(2) medium lances 20 "
"(3) heavy lances 30. "
7120 PRINT "(1) Swords cost 20 "
"(2) axes 20 " "(3) maces 30. "
Shields 20. "
7130 FOR f=1 TO 21: FOR g=0 TO 7

```

```

: READ a: POKE USR CHR# (f+143)+
g,a: NEXT g: NEXT f
7200 PRINT AT 21,0;"What armour
do you buy?": LET a$=INKEY$: IF
a$<"1" OR a$>"2" THEN GO TO 7200
7210 BEEP .5,0: LET ar=VAL a$: L
ET cash=cash-30-10*ar
7220 PRINT AT 21,0;"What lunce d
o you buy? ": LET a$=INKEY$: IF
a$<"1" OR a$>"3" THEN GO TO 7220
7230 BEEP .5,0: LET la=VAL a$: L
ET cash=cash-10*la
7240 PRINT AT 21,0;"What other w
eapon do you buy?": LET a$=INKEY
$: IF a$<"1" OR a$>"3" THEN GO T
O 7240
7250 BEEP .5,0: LET wa=VAL a$: L
ET w$=CHR# (152+wa): IF cash-20-
10*(wa=3)<0 THEN GO TO 7240
7260 LET cash=cash-10-10*(wa=3):
IF cash<20 THEN GO TO 7300
7270 PRINT AT 21,0;"Do you buy a
shield? (y/n) ": LET a$=INKEY#
: IF a$<"y" AND a$<"n" THEN GO
TO 7270
7280 BEEP .5,0: LET cash=cash-20
*(a$="y"): LET sh=(a$="y")
7300 LET sa=25: LET se=20: LET a
m=25: LET em=20
7360 CLS : PRINT "Your opponen
t chooses: "
7370 GO TO 7400+INT (RND*2)*50
7400 PRINT "Defence: Medium armou
r. " "Attack: Heavy lance and a
mace. "
7410 LET er=2: LET es=0: LET we=
3: LET le=3: LET v$="I": GO TO 8
000
7450 PRINT "Defence: Heavy armou
r and shield " "Attack: Medium l
ance and sword. "
7460 LET er=2: LET es=1: LET we=
1: LET le=2: LET v$="G"
8000 PRINT " (Press ENTER to st
art) ": PAUSE 0: CLS
8010 PLOT 0,120: DRAW 250,0: PLO
T 0,120: DRAW 0,30: GO SUB 8500
8020 DRAW 0,-20: GO SUB 8500: DR
AW 0,30: GO SUB 8500
8030 DRAW 0,-10: GO SUB 8500: DR
AW 0,10: GO SUB 8500
8040 DRAW 0,-30: GO SUB 8500: DR
AW 0,20: GO SUB 8500: DRAW 0,-30
8050 PLOT 115,120: DRAW 0,10: DR
AW 20,0,-PI: DRAW 0,-10
8060 FOR f=15 TO 230 STEP 70: PL
OT f,130: DRAW 5,0: DRAW 0,7: DR
AW -5,0,PI: DRAW 0,-7: NEXT f: G
O TO 9000
8500 FOR f=1 TO 3: DRAW 0,5: DRA
W 5,0: DRAW 0,-5: DRAW 5,0: NEXT
f: DRAW 0,5: DRAW 5,0: DRAW 0,-
5: RETURN
9000 DATA 1,10,15,23,31,59,51,33
,236,61,191,255,255,255,255,255,
0,120,224,240,240,244,243,224
9010 DATA 3,6,8,4,3,0,0,255,22
7,128,0,0,0,0,102,102,112,56,8
,4,2,6
9020 DATA 1,1,0,1,3,2,6,0,254,24
0,192,126,1,2,2,0,224,192,64,128
,0,0,0,0
9030 DATA 120,60,240,232,240,220
,204,132,55,100,253,255,255,255,
255,255
9040 DATA 0,1,7,15,31,47,207,7,1
20,120,0,120,102,64,96,0,127,31,
3,1,128,64,64,0,7,3,2,1,0,0,0,0
9050 DATA 192,96,16,16,52,192,0,
0,255,199,1,0,0,0,0,0,3,3,14,28,
16,32,64,96
9060 DATA 120,240,112,56,40,246,
214,116,30,31,14,26,20,111,107,4
0,0,0,0,0,0,0,255,255
9100 DATA 0,0,32,63,0,71,120,0,3
,4,0,251,255,240,0,240
9110 DATA 60,60,60,60,54,34,34,1
82,60,60,60,60,100,68,68,54
9120 DATA 0,64,32,16,10,4,10,3,0
,0,48,112,72,6,4,3,16,24,60,24,1
6,0,4,3
9130 DATA 0,2,4,5,60,32,60,192,0
,0,12,14,18,16,32,192,6,24,60,24
,16,32,64,192
9140 RETURN
9200 PRINT AT 16,2;"PRESS Y TO P
LAY AGAIN";AT 17,2;"OR N TO STOP
"
9210 IF INKEY#="y" THEN RUN
9220 IF INKEY#="n" THEN RANDOMIZ
E USR 0
9230 IF INKEY#="" OR INKEY#<"y"
OR INKEY#<"n" THEN GO TO 9200

```



Το όνομα του προγράμματος, αλλά και μια ματιά στο listing κάνουν φανερό το αντικείμενό του. Εκείνο που ίσως σας παραξενέψει είναι τα ελληνικά κείμενα και η μυστηριώδης γραφή των αγγλικών εντολών. Η εξήγηση είναι ότι το πρόγραμμα αυτό, βασίζεται σ'ένα παλιότερο αφιέρωμα του "COMPUTER ΓΙΑ ΟΛΟΥΣ". Ο τίτλος του

άρθρου ήταν "Μιλείστε ελληνικά με τον Spectrum" και δημοσιεύτηκε στο 4ο τεύχος. Αν έχετε ήδη πληκτρολογήσει εκείνο το πρόγραμμα θα ξέρετε την διαδικασία που πρέπει να ακολουθήσετε ώστε να αντικαταστήσετε το set των λατινικών χαρακτήρων. Όσοι από σας δεν έτυχε να "τρέξουν" εκείνο το πρόγραμμα έχουν την ευ-

καιρία να το κάνουν τώρα προμηθευόμενοι εκείνο το τεύχος, αν δεν το έχουν.

Αφού λοιπόν "μάθετε" τον Spectrum ελληνικά δεν έχετε παρά να πληκτρολογήσετε το listing που σας δίνουμε, με λίγη προσοχή βέβαια μια και οι εντολές είναι σε "ελληνική διάλεκτο" της BASIC... Η διάγνωση των βλαβών βασίζεται σ'ένα κύριο

μενού, που περιλαμβάνει τα εξής:

1. Μηχανή / μίζα
2. Φώτα / ηλεκτρικά
3. Φρένα
4. Σύστημα οδήγησης
5. Κιβώτιο ταχυτήτων

Διαλέγοντας μια κατηγορία σας ζητούνται ορισμένες πληροφορίες και βάσει αυτών εκτιμάται το αίτιο της βλάβης (όχι πάντα και με μεγάλη σιγουριά...).

## ΔΙΑΓΝΩΣΗ ΒΛΑΒΩΝ ΑΥΤΟΚΙΝΗΤΩΝ

```

10 ΨΛΣ
30 PRINT "
40 PRINT " ΔΙΑΓΝΩΣΗ ΒΛΑΒΩΝ Α
ΥΤΟΚΙΝΗΤΟΥ"
50 PRINT "
60 PRINT " ΠΙΕΣΤΕ ΤΟΝ ΑΝΤΙΣ
ΤΟΙΧΟ ΑΡΙΘΜΟ
ΟΥ ΒΡΙΣΚΕΤΑΙ
ΛΗΜΑ"
210 PRINT "
1. ΜΗΧΑΝΗ/ΜΙΖΑ
2. ΦΩΤΑ/ΗΛΕΚΤΡ
3. ΦΡΕΝΑ
4. ΣΥΣΤΗΜΑ ΟΔΗ
ΓΗΣΗΣ"
240 PRINT "
250 PRINT "
270 ΠΛΟΤ 0,167: ΔΡΑΥ 255,0: ΔΡΑ
Υ 0,-151: ΔΡΑΥ -255,0: ΔΡΑΥ 0,15
1
300 LET I=4000 INKEY$: IF I<49
OR I>53 THEN GO TO 300
315 ΨΛΣ
320 GO TO 1000+(I-48)
600 LET I$=INKEY$: IF I$<>"0" A
NA I$<>"N" THEN GO TO 500
510 PRINT ("ΝΑΙ" AND I$="N")+("
OXI" AND I$="0")
520 RETURN
1000 REM
1010 PRINT "ANABEI H ΜΗΧΑΝΗ; (
N/O)"
1020 GO TO 500
1040 IF I$="N" THEN GO TO 1500
1050 PRINT "ΓΥΡΙΖΕΙ Η ΜΙΖΑ; (N
/O)"
1060 GO TO 500
1070 IF I$="N" THEN GO TO 1250
1080 ΨΛΣ
1090 PRINT "1. ΤΑ ΦΩΤΑ ΔΕΝ ΑΝΑ
ΒΟΥΝ"
1100 PRINT "2. ΤΑ ΦΩΤΑ ΑΝΑΒΟΥΝ
ΑΜΥΔΡΑ"
1110 PRINT "3. ΤΑ ΦΩΤΑ ΕΙΝΑΙ Κ
ΑΝΟΝΙΚΑ"
1120 INPUT A
1130 IF A<1 OR A>3 THEN GO TO 11
20
1135 ΨΛΣ
1140 IF A=2 THEN GO TO 1190
1145 IF A=3 THEN GO TO 1220

```

```

1150 PRINT AT 10,0;"1. ΑΔΕΙΑ ΜΠΑ
ΤΑΡΙΑ"
1160 GO TO 9000
1190 PRINT AT 10,0;"1. ΑΔΕΙΑ ΜΠΑ
ΤΑΡΙΑ"
1195 PRINT "2. ΧΑΛΑΡΕΣ Η ΒΡΩΜΙΚΕΣ ΣΥ
ΝΔΕΣΕΙΣ ΜΠΑΤΑΡΙΑΣ"
1200 GO TO 9000
1210 PRINT "4. ΚΟΛΛΗΜΕΝΗ ΜΗΧΑΝΗ"
1220 PRINT AT 10,0;"1. ΧΑΛΑΡΗ, Β
ΡΩΜΙΚΗ Η ΚΑΚΗ ΣΥΝΔΕΣΗ ΣΤΟ ΚΥΚΛΩ
ΜΑ ΕΚΚΙΝΗΣΗΣ"
1225 PRINT "2. ΧΑΛΑΡΩΜΕΝΟ ΤΥΛΙΓΜΑ
ΣΤΗ ΜΙΖΑ"
1230 PRINT "3. ΕΛΛΑΤΩΜΑΤΙΚΗ ΜΙΖΑ
ΡΕΛΑΙ ΜΙΖΑΣΗ ΔΙΑΚΟΠΤΗΣ ΕΚΚΙΝΗΣΗ
Σ"
1240 GO TO 9000
1250 ΨΛΣ
1260 PRINT AT 10,0;"1. ΧΑΛΑΡΟΙ, Β
ΡΩΜΙΚΟΙ Η ΔΙΑΒΡΩΜΕ-ΝΟΙ ΑΚΡΟΔΕΚΤ
ΕΣ ΜΠΑΤΑΡΙΑΣ Η ΓΕΙΩΣΗ"
1270 PRINT "2. ΕΛΛΑΤΩΜΑΤΙΚΗ ΜΙΖΑ
Η ΡΕΛΑΙ ΜΙ-ΖΑΣ"
1280 GO TO 9000
2000 REM
2010 ΨΛΣ
2020 PRINT "ΤΙ ΔΕΝ ΔΟΥΛΕΥΕΙ ΣΟ
ΣΤΑ;"
2030 PRINT "1. ΕΝΔΕΙΚΤΙΚΑ ΟΡΓΑΝ
Α"
2040 PRINT "2. ΕΞΩΤΕΡΙΚΑ ΦΩΤΑ"
2050 PRINT "3. ΟΛΟ ΤΟ ΗΛΕΚΤΡΙΚΟ
ΣΥΣΤΗΜΑ"
2060 INPUT A
2070 IF A<1 OR A>3 THEN GO TO 20
60
2080 IF A=2 THEN GO TO 2400
2090 IF A=3 THEN GO TO 2700
2100 ΨΛΣ
2110 PRINT "1. ΕΝΔΕΙΚΤΙΚΟ ΕΚΙΝΝ
ΗΣΗΣ ΜΗΧΑΝΗΣ 2. ΕΝΔΕΙΚΤΙΚΟ ΠΙΕΣΗ
Σ ΛΑΔΙΟΥ"
2120 INPUT A
2130 IF A<1 OR A>3 THEN GO TO 21
20
2140 IF A=2 THEN GO TO 2200
2145 ΨΛΣ
2150 PRINT AT 10,0;"1. Ο ΙΜΑΝΤΑΣ
ΤΟΥ ΔΥΝΑΜΟ ΕΙΝΑΙ ΧΑΛΑΡΟΣ Η ΚΟ
ΜΜΕΝΟΣ"
2160 PRINT "2. ΚΑΠΟΙΑ ΑΣΦΑΛΕΙΑ Ε
ΙΝΑΙ ΚΑΜΜΕΝΗ"
(ΔΕΝ ΓΙΝΕΤΑΙ ΦΩΡ

```



Στριμώξτε τον αντίπαλό σας σε μια γωνιά - αυτός είναι ο σκοπός του παιχνιδιού που σας παρουσιάζουμε εδώ. Οι δυο παίκτες μπορούν να παίζουν ταυτόχρονα στο πληκτρολόγιο χάρη στη χρήση εντολών IN. Οι παίκτες κινούνται μέσα στην οθόνη αφήνοντας πίσω τους ένα θανατηφόρο ίχνος που έχει τη μορφή τοίχου. Για να επιζήσετε πρέπει να ρίξετε τον αντίπαλό σας σε αδιέξοδο ώστε να καταστραφεί. Η "δράση" απεικονίζεται στην οθόνη από γωνία 45° χρησιμοποιώντας την BRIGHT ώστε η κορυφή του ίχνους να φαίνεται πιο φωτεινή από το κάτω μέρος. Ο παίκτης που θα καταφέ-

ρει να νικήσει 10 φορές είναι και ο τελικός νικητής. Έχετε τη δυνατότητα να διαλέξετε το χρώμα που θα αφήνει το ίχνος κάθε παίκτη ακόμα δε και να έχετε το ίδιο χρώμα (για μεγαλύτερη δυσκολία).

Σκόπιμο είναι να πληκτρολογήσετε τις γραμμές 9000 ως 9030 και να τις "τρέξετε" ώστε να δείτε τους graphics χαρακτήρες όπως εμφανίζονται στο listing. Εκτός από τους χαρακτήρες (graphics) A, B και C, οι υπόλοιποι χαρακτήρες που χρησιμοποιούνται, είναι οι κανονικοί χαρακτήρες του SPECTRUM (π.χ. 32 graphics "8" στην γραμμή 10).

# TRON



```

9000 GO SUB 9000
9010 GO SUB 9000
9020 LET s1=0: LET s2=0
9030 PRINT AT 0,0: "
9040 PRINT AT 20,0: "
9050 PRINT AT 21,0: "
9060 PRINT AT 1,0: "
9070 FOR f=2 TO 19: PRINT AT f,0
; PAPER 4: "
9080 NEXT f
9090 LET x=4: LET y=10
9100 LET a=27: LET b=10
9110 LET d=2: LET c=4

```

```

95 GO TO 100+50*(RND<.5)
110 LET a$=("1" AND IN 63436<>2
55)+("2" AND IN 64510=253)+("3"
AND IN 65022<>255)+("4" AND IN 6
4510=254)
120 IF LEN a$=1 THEN LET d=VAL
a$
125 LET x=x+(d=2)-(d=4): LET y=
y+(d=3)-(d=1)
126 IF SCREEN$(y,x)<>"" THEN G
O TO 2000
130 PRINT AT y,x: BRIGHT 1: INK
c1: "
140 IF SCREEN$(y+1,x)="" THEN
PRINT AT y+1,x: PAPER c1: INK c1
; "
145 BEEP .01,0
150 LET a$=("1" AND IN 61438<>2
55)+("2" AND IN 57342=254)+("3"
AND IN 49150<>255)+("4" AND IN 5
7342=253)
170 IF LEN a$=1 THEN LET c=VAL
a$
175 LET a=a+(c=2)-(c=4): LET b=
b+(c=3)-(c=1)
176 IF SCREEN$(b,a)<>"" THEN G
O TO 1000
180 PRINT AT b,a: BRIGHT 1: INK
c2: "
190 IF SCREEN$(b+1,a)="" THEN
PRINT AT b+1,a: PAPER c2: INK c2
; "
195 BEEP .01,10
200 GO TO 100
1000 PRINT AT b,a: FLASH 1: " "
1010 LET s1=s1+1
1020 GO TO 3000
2000 PRINT AT y,x: FLASH 1: " "
2010 LET s2=s2+1
3000 FOR f=30 TO 60: BEEP .001,f
: NEXT f
3010 PRINT INVERSE 1: AT 9,11: "PL
ayer 1: ";s1
3020 PRINT INVERSE 1: AT 11,11: "P
layer 2: ";s2
3030 FOR f=60 TO 30 STEP -1: BEE
P .001,f: NEXT f
3035 IF s1>9 OR s2>9 THEN GO TO
4000
3040 IF INKEY$<>"" THEN GO TO 30
40
3050 PRINT AT 20,3: "Press any ke
y to CONTINUE"
3060 IF INKEY$<>"" THEN GO TO 10
3070 GO TO 3050
4000 PRINT FLASH 1: AT 13,10: "PLA
YER ";(s2=10)+1: " WINS"
4010 POKE 23617,237: INPUT "Anot
her go (y/n) "; LINE a$
4020 IF a$<>"n" AND a$<>"N" AND
a$<>"y" AND a$<>"Y" THEN GO TO 4
010
4030 IF a$(1)<>"n" AND a$(1)<>"N
" THEN RUN
4040 STOP
8000 BORDER 0: PAPER 0: INK 7: C
L5
8020 PRINT "TAB 12: "CHASER""
8030 PRINT "The object of CHASE
R is to forcethe other player in
to a wall."
8040 PRINT "Each player has a
chaser which leaves a trail, the
chasers can not stop until one
is destroyed."
8050 PRINT "TAB 12: "CONTROLS"
8055 PRINT "TAB 14: "^ < > v"
8060 PRINT "Player 1. 2 0 W
A"
8070 PRINT "Player 2. 0 0 F
L"
8080 PRINT "TAB 14: "^ < > v"
8100 POKE 23617,237: INPUT "Play
er's 1 colour (1 TO 7) ";c1
8110 IF c1<1 OR c1>7 THEN GO TO
8100
8120 INPUT "Player's 2 colour (1
TO 7) ";c2
8130 IF c2<1 OR c2>7 THEN GO TO
8120
8140 RETURN
9000 FOR f=0 TO 7 STEP 2: POKE U
SR "a"+f,170: POKE USR "a"+f+1,6
5: NEXT f
9010 FOR f=0 TO 6: POKE USR "b"+
f,126: NEXT f: POKE USR "b"+7,25
5
9020 RESTORE : FOR f=0 TO 7: REA
D a: POKE USR "c"+f,a: POKE USR
"d"+7-f,a: NEXT f
9030 DATA 0,0,8,20,34,0,0,0
9040 RETURN

```

# Ο ΖΟΓΚΛΕΡ

Σκοπός του παιχνιδιού αυτού είναι να κρατήσετε τις μπάλλες που κινούνται, μέσα στην οθόνη έτσι ώστε να μην σας φύγουν από το κάτω μέρος που είναι ελεύθερο. Εσείς μετακινείτε μια ρακέτα αριστερά-δεξιά στο βάθος της οθόνης και πρέπει να αποκρούσετε μ'αυτήν τις μπάλλες όταν φεύγουν προς τα

κάτω.

Στην αρχή υπάρχει μια μόνο μπάλλα αλλά κάθε 25 δευτερόλεπτα, μια επιπλέον μπάλλα κάνει την εμφάνισή της. Το σκορ ανανεώνεται συνέχεια και είναι ανάλογο και με το πόσες μπάλλες υπάρχουν στην οθόνη. Οι οδηγίες του παιχνιδιού τυπώνονται από το πρόγραμμα μ'έναν ευχάριστο

και πρωτότυπο τρόπο. Το ενδιαφέρον του προγράμματος προκύπτει κυρίως από την μεγάλη ταχύτητα της κίνησης, που εξασφαλίζεται με την χρήση κώδικα μηχανής. Η πληκτρολόγηση θέλει ιδιαίτερη προσοχή γιατί τα λάθη σε κώδικα μηχανής μπορούν να σας κάνουν να χάσετε το πρόγραμμα. Αν τα στοιχεία του κώδικα

μηχανής δεν είναι όσα πρέπει να είναι το πρόγραμμα θα σταματήσει απλώς με την ένδειξη "DATA ERROR" που σημαίνει ότι κάποιος αριθμός έχει παραληφθεί ή επαναληφθεί. Σας συμβουλεύουμε πάντως να "σώσετε" το πρόγραμμα προτού το "τρέξετε" ώστε να μην διακινδυνεύσετε τον κόπο της πληκτρολόγησης.

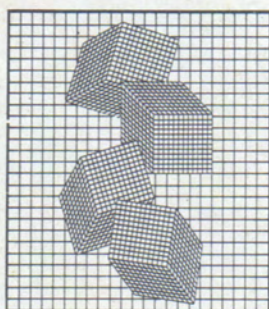
```

20 GO TO 9000
20 LET A$="": LET X=10: GO SUB
      JUGGLER
200 LET A$="": S<-
  -> S: LET X=12: GO SUB
200 LET A$="": H=HOLD
  S=START: LET X=14: GO SUB
200 GO TO 300
200 FOR A=1 TO LEN A$: PRINT AT
  X,0: BRIGHT 1: A$(LEN A$-A+1 TO
  X): BEEP .01, A+X: NEXT A: RETURN
300 FOR A=1 TO 10: POKE 32346+4
  *A, INT (RND*6)+3: POKE 32347+4**A
  , INT (RND*20)+2: POKE 32348+4**A,
  1-(2 AND RND>.5): POKE 32349+4**A
  , 1-(2 AND RND>.5): NEXT A
310 LET A$="": PRESS S
      : LET X=21: GO SUB
200 GO SUB 500
330 CLS: PRINT BRIGHT 1: FLASH
  1: AT 10,11: "BE READY": PAUSE 50
  : CLS: PRINT AT 0,0: LET L=USR
  32000
340 LET r=0: LET p=0: GO SUB 40
0 350 IF INKEY$="h" THEN GO SUB 5
00 360 IF PEEK 23673>=5 THEN GO SU
  B 400
380 LET p=p+r: PRINT AT 0,27: P
  APER 1:p: PAPER 0: LET L=USR 325
  27: IF PEEK 23296<>99 THEN GO TO
  350
390 GO TO 300
400 POKE 23672,0: POKE 23673,0:
  LET r=r+(1 AND r<10): POKE 325
  31,r: POKE 32575,r: POKE 32578,3
  0-2*r: PRINT AT 0,1: PAPER 1: "RO
  UND":r: AT 0,20: "POINTS":p: RET
  URN
500 LET t=PEEK 23672: LET t1=PE
  EK 23673
510 IF INKEY$<>"s" THEN GO TO 5
  10
520 POKE 23672,t: POKE 23673,t1
  : RETURN
9000 DATA 62,17,215,62,1,215,62,
  5,215,62,6,215,6,21,62,22,215,12
  0,215,62,6,215,6,2,32,215,6,2,2
  15,120,215,6,2,31,215,6,2,215,1
  5,-24,33,175,90,34,176,92,6,3,54
  ,67,35,16,-5
9010 DATA 2001
9020 DATA 62,0,50,129,92,1,254,2
  39,237,120,203,87,32,14,56,176,9
  2,254,188,40,7,67,1,50,129,92,24
  ,24,1,254,247,237,120,203,183,32
  ,14,56,176,92,254,161,40,7,62,23
  
```

```

5,50,129,92,24,1,2001
9030 DATA 62,50,129,92,254,1,32,16
  42,176,92,54,7,99,92,254,1,32,16
  4,43,34,176,92,92,92,42,176,92,2,43
  4,43,34,176,92,92,92,42,176,92,7,2001
9050 DATA 62,0,50,129,92,254,1,32,16
  150,92,215,215,0,0,0,1,1,0,0,0,7,1
  120,92,215,215,0,0,0,1,1,0,0,0,7,1
  120,92,215,215,0,0,0,1,1,0,0,0,7,1
  233,33,0,91,112,95,115,62,22,215,
  120
9060 DATA 215,123,215,62,16,215,
  92,4,215,62,144,215,201,33,0,91,
  92,2,91,134,71,95,94,205,244,126
  ,254,7,92,16,56,9,91,93,50,9
  1,91,24,90,33,6,91,93,7,50,0,91,
  9070 DATA 92,16,100,0,91,7,50,0,91,
  901,91,24,12,33,6,91,126,237,68,5
  119,95,126,237,6,91,119,33,0,91,50
  ,91,134,119,3,5,9,91,134,119
  ,43,70,35,94,62,22,215,120,215,1
  23,215,62,16
9080 DATA 215,62,4,215,62,144,21
  93,0,1,17,20,205,181,3,201
9081 DATA 197,205,6,10,197,99,50
  ,205,6,197,205,17,4,6,205,181,3,
  909,50,0,91,201
9090 CLEAR 31999: LET A$="": LET
  PLEASE WAIT
  X=10: GO SUB 200
9095 LET T=0: FOR A=32000 TO 323
  42: READ B: LET T=T+B: POKE A,B:
  NEXT A: IF T<>31720 THEN GO TO
  9999
9100 RESTORE 9105: LET T=0: FOR
  A=32500 TO 32593: READ B: LET T=
  T+B: POKE A,B: NEXT A: IF T<>105
  80 THEN GO TO 9999
9105 DATA 197,213,229,213,62,0,1
  84,40,9,17,32,0,33,0,88,25,16,-3
  ,209,22,0,25,126,225,209,193,201
9110 DATA 33,94,126,6,1,197,229,
  17,0,91,237,160,237,160,237,160,
  237,160
9120 DATA 205,142,125,225,17,0,9
  1,235,126,254,21,202,255,126,237,
  160,237,160,237,160,237,160,235,
  193,16,-39
9130 DATA 205,52,125,6,2,197,6,3
  0,197,6,0,16,-2,193,16,-2,205,52
  ,125,193,16,-17,201
9140 RESTORE 9140: FOR A=0 TO 7:
  READ B: POKE USR "A"+A,B: NEXT
  A: DATA 60,126,255,255,255,255,1
  20,60
9150 GO TO 20
9999 PRINT AT 10,7: FLASH 1: "
  DATA ERROR": STOP
  
```

# PASCAL-FORTH COMPILERS-ASSEMBLERS



**HOME/PERSONAL  
& BUSINESS COMPUTERS**

**ΕΤΟΙΜΑ ΠΡΟΓΡΑΜΜΑΤΑ • ΠΑΙΓΝΙΔΙΑ  
ΒΙΒΛΙΑ • ΠΕΡΙΟΔΙΚΑ • CASSETES • DISKS**

**ΤΣΑΚΑΛΩΦ & ΠΙΝΔΑΡΟΥ 25 • ΑΘΗΝΑ**



listing 3

```

: 14,28,28,28,0,24,60,126,
: 28,28,28,0,24,60,126,80,
: 28,28,0,24,60,126,80,40,
: 28,0,24,60,126,80,40,40,
: 0,24,60,126,80,40,40,56,
: 24,60,126,80,40,40,56,0,
: 60,126,80,40,40,56,0,24,
: 126,80,40,40,56,0,24,36,
: 80,40,40,56,0,24,36,126,
: 40,40,56,0,24,36,126,7,
: 40,56,0,24,36,126,7,7,
: 56,0,24,36,126,7,7,15,
: 0,24,36,126,7,7,15,31,
: 24,36,126,7,7,15,31,27,
: 36,126,7,7,15,31,27,14,
: 126,7,7,15,31,27,14,14,
: 7,7,15,31,27,14,14,14,
: 7,15,31,27,14,14,14,160,
: 15,31,27,14,14,14,160,160,
: 31,27,14,14,14,160,160,112,
10 FOR f=USR "a" TO USR "t"+7
20 INPUT P: PRINT f,P: POKE f,
P: NEXT f

```

## Ο ΠΟΛΕΜΟΣ ΤΩΝ ΑΣΤΡΩΝ

Με την ευκαιρία της προβολής της ταινίας "Η επιστροφή των Jedi" - τελευταίο μέρος του τρίπτυχου "Ο πόλεμος των άστρων" και "Η αυτοκρατορία αντεπιτίθεται" - σας δίνουμε την ευκαιρία να αντιμετωπίσετε τις φοβερές μηχανικές καμήλες του πλανήτη Hoth. Το πρόγραμμα είναι γραμμένο για τον 16K Spectrum αλλά αυτόματα υπολογίζει τις

διευθύνσεις των user defined χαρακτήρων και "τρέχει" και σε 48K.

Ο ρόλος σας είναι να καταστρέψετε τις καμήλες της Αυτοκρατορίας, πετώντας με το τελευταίο διαστημικό καταδιωκτικό σκάφος που έχει απομείνει. Η οδήγηση του σκάφους σας είναι απλή και σαν βοήθημα έχετε όργανα που δείχνουν το υψόμετρο και τον ρυθμό ανόδου-καθόδου του

σκάφους. Το τελευταίο αυτό όργανο βρίσκεται στο κέντρο του πίνακα ελέγχου και είναι βαθμολογημένο σε ενδείξεις των 8 m/sec. Το καταδιωκτικό σας μπορεί επίσης να αλλάξει διεύθυνση πορείας αλλά όχι πάντα εύκολα. Η πτήση σας γίνεται ακόμα πιο δύσκολη από τα εχθρικά πυρά. Μόλις το σκάφος κτυπηθεί από laser, αρχίζει να χάνει ύψος και αν δεν το ανεβάσετε, θα καταστραφεί στο έδαφος.

Η καταστροφή μιας καμήλας απαιτεί ταλέντο στην πλοήγηση. Πρέπει να πετάτε σε ύψος 28 μέτρων (ακριβώς πάνω από τα εχθρικά πυρά) και να πυροβολείτε στο κεφάλι. Στην αρχή του παιχνιδιού σας ζητείται το επίπεδο δυσκολίας του παιχνιδιού (1-5). Αυτό αν-

```
10 CLEAR '32393: LET A=32400
20 INPUT "CODE : "; B: PRINT A,
B: POKE A,B: LET A=A+1: GO TO 20
```

```
32400
6,64,17,0,64,213,225,35,197,1,3
1,0,26,237,176,43,119,0,35,35,19
,193,16,240,201,
```

```
32425
6,64,17,255,71,213,225,43,197,1,
31,0,26,237,184,35,119,0,43,43,2
7,193,16,240,201,
```

```
32450
6,64,17,0,72,213,225,35,197,1,31
,0,26,237,176,43,54,0,35,35,19,1
93,16,240,201,
```

```
32475
6,64,17,255,79,213,225,43,197,1,
5,0,26,237,184,35,54,0,43,43,27
,193,16,240,201,
```

```
32500
6,1,197,33,0,1,17,4,0,229,205,18
1,3,225,17,16,0,167,237,82,32,24
0,193,16,233,201,0,0,0,0,
```

```
32530
6,1,197,33,0,3,17,1,0,229,205,18
1,3,225,17,16,0,167,237,82,32,24
0,193,16,233,201,0,0,0,0,
```

```
32560
6,2,197,33,15,0,17,1,0,229,205,1
81,3,225,17,4,0,167,237,90,125,2
54,255,32,237,193,16,230,201,0,
```

listing 4

```
100 PAPER 7: CLS : FOR L=16 TO
21: PRINT PAPER 2; AT L,0; " : NEX
T L
110 INK 6: PLOT 0,0: DRAW 255,0
: DRAW 0,46: DRAW -255,0: DRAW 0
,-46: DRAW 2,2: DRAW 251,0: DRAW
0,42: DRAW -251,0: DRAW 0,-42
120 PLOT 3,3: DRAW 87,0: DRAW 0
,16: DRAW -87,0: DRAW 0,-16: PLO
T 3,27: DRAW 87,0: DRAW 0,16: DR
AW -87,0: DRAW 0,-16
130 PLOT 7,7: DRAW 0,9: DRAW 81
,0: DRAW 0,-9: DRAW -81,0: PLOT
7,31: DRAW 0,9: DRAW 81,0: DRAW
0,-9: DRAW -81,0
140 PRINT FLASH 1: INK 2: AT 17,
1: "HEIGHT": AT 20,1: "SCORE": CHR#
8: OVER 1: " <
150 PLOT 96,3: DRAW 103,0: DRAW
0,40: DRAW -103,0: DRAW 0,-40
160 INK 7: LET A=168: FOR F=0 T
O PI STEP PI/270
170 PLOT A,17: DRAW 23*COS F,22
*SIN F
180 IF F>=PI/2 THEN LET A=127
190 NEXT F
200 INK 1: LET A=127: FOR F=PI
TO 2*PI STEP PI/200: PLOT A,15:
DRAW 23*COS F,7*SIN F: IF F>=PI+
PI/2 THEN LET A=168
210 NEXT F
220 INK 6: FOR F=7 TO 39 STEP 4
: PLOT 128,F: DRAW 7,0: PLOT 160
,F: DRAW 7,0: NEXT F
230 FOR F=9 TO 37 STEP 4: PLOT
128,F: DRAW 4,0: PLOT 163,F: DR
AW 4,0: NEXT F
240 PLOT 204,3: DRAW 48,0: DRAW
0,16: DRAW -48,0: DRAW 0,-16: P
LOT 207,7: DRAW 0,9: DRAW 41,0:
DRAW 0,-9: DRAW -41,0
250 PLOT 204,27: DRAW 48,0: DR
AW 0,16: DRAW -48,0: DRAW 0,-16:
PLOT 207,31: DRAW 0,9: DRAW 41,0
: DRAW 0,-9: DRAW -41,0
260 PRINT AT 17,26: PAPER 1: IN
K 7: BRIGHT 1: "OR": INK 3: "O"
270 INK 6: PLOT 203,2: DRAW 0,4
1: PLOT 91,2: DRAW 0,41
280 FOR F=USR "A" TO USR "D"+7:
READ A: POKE F,A: NEXT F: DATA
0,16,56,123,255,255,255,0,0,0,8,
29,191,255,255,0,0,40,120,255,25
5,255,255,0,0,24,60,126,255,0,0,
0
290 PRINT AT 0,0: BRIGHT 1: PAP
ER 5: INK 7: "ACBBCBACBACB": I
NK 6: "BC": INK 7: "CBCBBCBACBAA
C"
300 PRINT AT 2,5: INK 2: "BF
ITTTLE BF
ON ON H
OTH. H
310 PRINT FLASH 1: BRIGHT 1: AT
14,3: INK 4: PAPER 6: " * Press an
y key to begin *"
320 SAVE "sc"SCREEN$
```

listing 2

τιστοιχεί στον αριθμό των κτυπημάτων που απαιτούνται για να καταστραφεί μια καμήλα. Μετά σας ζητείται ο διαθέσιμος συνολικά χρόνος για την αποστολή. Μια τιμή γύρω στα 400 είναι καλή για τα πρώτα παιχνίδια. Αν το παιχνίδι σας φαίνεται πολύ δύσκολο, μπορείτε να επέμβετε στην γραμμή 2805. Εκεί βρίσκεται το ύψος από το οποίο μια βολή βρίσκει τον στόχο της. Έτσι, το 79 αντιστοιχεί στο ύψος των 28 m που προαναφέραμε. Αλλάζοντας την γραμμή σε IF M<77 OR M>81 THEN RETURN, οι βολές θα είναι εύστοχες μεταξύ των υψών 26-30 m και γενικά το ύψος καθορίζεται από τα όρια της σύγκρισης που γράψαμε πιο πάνω.

Η πλοήγηση γίνεται με τα παρακάτω πλήκτρα:  
Q μέχρι P: αύξηση ρυθμού ανόδου  
A μέχρι ENTER: μείωση ρυθμού ανόδου  
SHIFT μέχρι V στροφή αριστερά

B μέχρι SPACE στροφή δεξιά  
1 μέχρι 0 laser

Κατά την πλοήγηση πρέπει να ακολουθηθεί η εξής πορεία:

- 1) Πληκτρολογείτε το κυριώς πρόγραμμα (10-9504) και "σώστε" το με την εντολή SAVE "BATTLE" LINE 10. Σημειώστε ότι όλοι οι χαρακτηριστές μέσα σε εισαγωγικά είναι οι αντίστοιχοι graphics χαρακτήρες. Εξαιρούνται οι γραμμές 2000 και 3010 όπου μόνον το "R" είναι graphic και φυσικά οι γραμμές που περιέχουν κείμενα ή λέξεις.
- 2) Πληκτρολογείτε και "τρέξτε" το πρόγραμμα του listing 2. Δώστε έναν-έναν τους αριθμούς που βρίσκονται ακριβώς από κάτω (όχι όμως και τους πενταψήφιους οι οποίοι είναι απλώς για να σας διευκολύνουν). Όταν φτάσετε στο τελευταίο 0 δώστε STOP και κάντε NEW.
- 3) Σειρά έχει το πρό-

γραμμα του listing 3 το οποίο καθορίζει τους graphic χαρακτήρες και πρέπει να το χειριστείτε όπως το προηγούμενο, με την διαφορά ότι αυτό θα σταματήσει μόνο του με το τέλος των δεδομένων.

4) Δώστε την εντολή SAVE "ON" CODE 32400, 190: SAVE "HOTH" CODE USR "A", 168 και "ώστε" έτσι τους κωδικούς αμέσως μετά το κυρίως πρόγραμμα.

5) Πληκτρολογήστε το πρόγραμμα που σχεδιάζει την οθόνη listing 4) και "τρέξτε" το.

Αφού σχεδιαστεί η οθόνη θα πάρετε το μήνυμα start tape then. οπότε πρέπει να "ώστε" την οθόνη αμέσως μετά τις προηγούμενες τρεις εγγραφές. Το ίδιο το πρόγραμμα που σχεδιάζει την οθόνη είναι άχρηστο αφού "τρέξει".

6) Γυρίστε την κασέτα στην αρχή του πρώτου προγράμματος. Αν όλα πάνε καλά, φορτώνοντάς το αυτό θα "τρέξει" αυτόματα και αφού φορτώσει μόνο του τις τρεις υπόλοιπες εγγραφές θα αρχίσει το παιχνίδι.

#### listing 1

```

10 CLEAR 32399: LOAD ""CODE :
LOAD ""CODE
30 PAPER 7: INK 7: CLS : BORDE
R 0: LOAD ""SCREEN$
40 IF INKEY$="" THEN GO TO 40
80 INPUT "DIFFICULTY (1 TO 5)
";X: INPUT "TIME (100↑) ":T
90 INK 0: OVER 0: PAPER 7: FOR
F=1 TO 15: PRINT AT F,0: "": NEX
T F
95 RESTORE
100 LET Z=0: LET S=0: LET M=100
: PAPER 7: LET A=-20: LET B=-40:
LET C=-60: LET D=-80: LET E=-10
0: LET F=X: LET G=F: LET H=F: LE
T I=F: LET J=F: LET M=79: LET N=
0
120 PLOT 0,128: FOR 0=0 TO 30:
READ R: DRAW 0,R: NEXT 0: DRAW 7
.5: DATA 2,3,4,3,1,-1,-3,-2,-4,-
2,-1,1,-1,-1,0,2,2,1,0,-1,-1,-2,-
-5,-5,0,0,1,1,2,2,-1
200 BEEP .5,0: BEEP .5,7: BEEP
.2,5.2: BEEP .2,3.2: BEEP .2,1.5
: BEEP .5,10.1: BEEP .3,5
210 BEEP .2,5.2: BEEP .2,3.2: B
EEP .2,1.5: BEEP .5,10.1: BEEP .
6,5
220 BEEP .2,5: BEEP .2,4: BEEP
.22,5: BEEP .7,1.5
2000 PRINT AT 17,26: OVER 0: PAP
ER 1: INK 7: BRIGHT 1: "OR": INK
3: "O": OVER 1
2010 LET L=0
2050 FOR K=0 TO 4
2055 LET O=K
2060 GO SUB 9350
2065 LET 0=0: GO SUB 2820
2070 GO SUB 9400: LET A=A+1: LET
B=B+1: LET C=C+1: LET D=D+1: LE
T E=E+1
2072 IF IN 63466<>255 OR IN 6143
8<>255 THEN GO SUB 2800
2075 GO SUB 9350
2080 RANDOMIZE 0#USR 32425: RAND
OMIZE 0#USR 32475
2090 GO SUB 9200
2100 NEXT K
2110 IF IN 32766<>255 THEN GO SU
B 2912: GO TO 3000
2150 GO TO 2050
2700 LET R=P*8+8: IF RND<.7 THEN
GO TO 2710
2701 INK 4: LET 0=INT (RND*8)-3:
PLOT R,74: DRAW 255-R,0: RANDOM
IZE USR 32500: INK 0: PLOT R,74:
DRAW 255-R,0: IF M>=77 THEN GO
TO 2710
2702 IF M<=69 THEN GO TO 2710
2703 GO SUB 2750
2710 LET 0=INT (RND*8+8): LET R=
INT (RND*10)+10: PRINT AT 0,R: P
APER 6: INK 2: BRIGHT 1: "P":AT 0
,R: INK 0: PAPER 7: BRIGHT 0: "P"
2720 RETURN

```

```

2750 RANDOMIZE 0#USR 32560: RAND
OMIZE 0#USR 32560: LET N=-5: RET
URN
2790 IF 0=0 THEN RETURN
2795 RANDOMIZE 0#USR 32560: LET
Z=Z+10: PRINT AT 20,26: PAPER 6:
INK 2: "HIT*": RETURN
2800 INK 2: PLOT 119,M: DRAW -11
9,0: RANDOMIZE 0#USR 32530: INK
0: PLOT 119,M: DRAW -119,0
2805 IF M<>79 THEN RETURN
2810 LET 0=1
2820 LET P=0
2822 IF A>=6 THEN IF A<=14 THEN
LET F=F-0: IF F>=1 THEN LET P=A:
GO SUB 2790
2823 IF F=0 THEN LET P=A: GO TO
2847
2825 IF B>=6 THEN IF B<=14 THEN
LET G=G-0: IF G>=1 THEN LET P=B:
GO SUB 2790
2826 IF G=0 THEN LET P=B: GO TO
2847
2830 IF C>=6 THEN IF C<=14 THEN
LET H=H-0: IF H>=1 THEN LET P=C:
GO SUB 2790
2831 IF H=0 THEN LET P=C: GO TO
2847
2835 IF D>=6 THEN IF D<=14 THEN
LET I=I-0: IF I>=1 THEN LET P=D:
GO SUB 2790
2836 IF I=0 THEN LET P=D: GO TO
2847
2840 IF E>=6 THEN IF E<=14 THEN
LET J=J-0: IF J>=1 THEN LET P=E:
GO SUB 2790
2841 IF J=0 THEN LET P=E: GO TO
2847
2845 IF 0=0 THEN GO TO 2700
2846 OVER 0: PRINT AT 20,26: PAP
ER 2: "": OVER 1: RETURN
2847 BEEP .01,0: OVER 0: LET P=P
-5: INK 2: PAPER 7: PRINT AT 11,
P: INK 0: "":AT 12,P: "HIJ":L
BRIGHT 1: PAPER 6: "0":AT 13,P: "L
": PAPER 6: "MN":AT 14,P: "CDCD":A
T 15,P: "ABBA"
2850 BEEP .01,-1: PRINT AT 12,P:
INK 0: "":AT 13,P: "H": PAP
ER 6: BRIGHT 1: "IO":AT 14,P: "L"
: PAPER 6: BRIGHT 1: "PPP":AT 15,
P: BRIGHT 1: PAPER 6: "OBPP"
2855 BEEP .01,-2: PRINT AT 13,P:
INK 0: "": PAPER 6:AT 14,P:
" H": BRIGHT 1: "OP":AT 15,P: "KQ"
: BRIGHT 1: "MP"
2860 BEEP .01,-3: PRINT AT 14,P:
INK 0: "": PAPER 6:AT 15,P:
"OP": BRIGHT 1: "5P"
2870 INK 0: PRINT AT 15,P: "HIJ"
2880 LET Z=Z+10: PRINT AT 20,8:Z
: IF F<=0 THEN IF G<=0 THEN IF H
<=0 THEN IF I<=0 THEN IF J<=0 TH
EN GO TO 9150
2890 GO TO 2822
2900 IF A=0 THEN IF F>=1 THEN GO
TO 2910
2903 IF B=0 THEN IF G>=1 THEN GO
TO 2910
2905 IF C=0 THEN IF H>=1 THEN GO
TO 2910
2907 IF D=0 THEN IF I>=1 THEN GO
TO 2910
2908 IF E=0 THEN IF J>=1 THEN GO
TO 2910
2909 RETURN
2910 PAPER 7: PRINT AT 12,0: "O":
RETURN
2912 IF A=35 THEN IF F>=1 THEN G
O TO 2922
2914 IF B=35 THEN IF G>=1 THEN G
O TO 2922
2916 IF C=35 THEN IF H>=1 THEN G
O TO 2922
2918 IF D=35 THEN IF I>=1 THEN G
O TO 2922
2920 IF E=35 THEN IF J>=1 THEN G
O TO 2922
2921 RETURN
2922 PAPER 7: PRINT AT 11,31: "G"
:AT 12,31: "K":AT 13,31: "E":AT 14
,31: "C":AT 15,31: "A"
2923 RETURN
3000 LET L=31
3010 PRINT OVER 0: PAPER 1: INK
3: BRIGHT 1:AT 17,26: "O": INK 7
: "RO"
3050 LET 0=4: FOR K=35 TO 31 STE
P -1
3060 GO SUB 9350
3065 LET 0=0: GO SUB 2820
3070 GO SUB 9400: LET 0=0-1: LET
A=A-1: LET B=B-1: LET C=C-1: LE

```



```

T D=D-1: LET E=E-1
3072 IF IN 61438(>255 OR IN 6348
6(>255 THEN INK 2: PLOT 136,M: D
RAW 119,0: RANDOMIZE 0#USR 32530
: INK 0: PLOT 135,M: DRAW 119,0
3075 GO SUB 9360
3080 RANDOMIZE 0#USR 32400: RAND
OMIZE 0#USR 32450
3090 GO SUB 9200
3100 NEXT K
3110 IF IN 65278(>255 THEN GO SU
B 2900: GO TO 2000
3150 GO TO 3050
9000 FLASH 1: OVER 0: INK 2: PRI
NT AT 15,15: PAPER 6: BRIGHT 1;"
QP": FOR F=0 TO 5: PLOT 128,50:
DRAW INT (RND*60)-30,INT (RND*20
)+8: NEXT F: FOR F=0 TO 4: RANDO
MIZE 0#USR 32560: NEXT F: PRINT
PAPER 2: INK 7: AT 17,8;"0": AT 2,
8: FLASH 1: BRIGHT 1: PAPER 1;"D
EFENCES BROKEN": GO TO 9100
9050 OVER 0: FLASH 1: PRINT AT 2
,0: PAPER 2: INK 6:">TIME OUT- G
ENERATORS DESTROYED("
9100 BEEP .5,0: BEEP .5,7: BEEP
.2,5.2: BEEP .2,3.2: BEEP .2,1.5
: BEEP .5,10.1: BEEP .75,5
9110 OVER 0: PRINT AT 4,3: PAPER
1: INK 7: BRIGHT 1;"* REBEL FOR
CES WIPED OUT *":
9120 OVER 0: PAUSE 100: INK 6: P
RINT AT 6,1: PAPER 4:"* Press an
y key to try again *": FLASH 0:
PAPER 2: PRINT AT 17,8;"": FO
R f=17 TO 20: PRINT AT f,17;"
": NEXT f: PRINT AT 21,17;"TTT"
9130 IF INKEY$="" THEN GO TO 913
0
9140 PRINT AT 20,8: PAPER 2;"
": GO TO 60
9150 OVER 0: FLASH 1: PRINT AT 2
,5: PAPER 7: INK 1:"IMPERIAL FOR
CES DOWNED": AT 4,0: BRIGHT 1: IN
K 6: PAPER 2:"*EVACUATION OF HOT
H PROCEEDING *": GO TO 9120
9200 LET T=T-1
9205 PAPER 2: INK 7: OVER 0: PRI
NT AT 17,8;"": AT 17,8;M-51: AT
20,8;2
9206 IF N(<=-1 THEN PRINT AT 17,2

```

```

6: PAPER 1: INK 7: BRIGHT 1;"5"
9208 IF N>=1 THEN PRINT AT 17,28
: PAPER 1: INK 7: BRIGHT 1;"R"
9210 OVER 1
9215 IF T=0 THEN GO TO 9050
9220 IF IN 65022(>255 OR IN 4915
6(>255 THEN LET N=N-1
9230 IF IN 64510(>255 OR IN 5734
2(>255 THEN LET N=N+1
9231 IF N(<=-9 THEN LET N=-8
9232 IF N>=9 THEN LET N=8
9235 LET M=M+N+(RND).2)-(RND).2)
9240 IF M(<=50 THEN GO TO 9000
9250 IF M>=164 THEN LET N=-2: LE
T M=164
9255 INK 6: PLOT 136,5+23: DRAW
23,0
9260 LET S=N#2
9270 PLOT 136,5+23: DRAW 23,2: I
NK 0
9300 RETURN
9350 PLOT 120,M: DRAW 12,0: DRAW
0,-3: DRAW -10,3: DRAW 4,2: DRA
W 4,0: RETURN
9360 PLOT 135,M: DRAW -12,0: DRA
W 0,-3: DRAW 10,3: DRAW -4,2: DR
AW -4,0: RETURN
9400 PAPER 7: IF A=K THEN IF F>=
1 THEN GO SUB 9500+0
9410 IF B=K THEN IF G>=1 THEN GO
SUB 9500+0
9420 IF C=K THEN IF H>=1 THEN GO
SUB 9500+0
9430 IF D=K THEN IF I>=1 THEN GO
SUB 9500+0
9440 IF E=K THEN IF J>=1 THEN GO
SUB 9500+0
9450 RETURN
9500 PRINT AT 12,L;"0": RETURN
9501 PRINT AT 11,L;"J": AT 12,L;"
N": AT 13,L;"F": AT 14,L;"D": AT 15
,L;"B": RETURN
9502 PRINT AT 11,L;"I": AT 12,L;"
M": AT 13,L;"E": AT 14,L;"C": AT 15
,L;"A": RETURN
9503 PRINT AT 11,L;"H": AT 12,L;"
L": AT 13,L;"F": AT 14,L;"D": AT 15
,L;"B": RETURN
9504 PRINT AT 11,L;"G": AT 12,L;"
K": AT 13,L;"E": AT 14,L;"C": AT 15
,L;"A": RETURN

```



ΒΕΒΑΙΑ, ΙΣΟΣ ΝΑ ΤΟ ΠΑΡΑΚΑΝΑΜΕ ΡΕΑΛΙΣΤΙΚΟ...

Ένας τέτοιος τίτλος ταιριάζει στο, εξτρεμιστικών διαθέσεων, πρόγραμμα που σας δίνουμε εδώ. Στο πρόγραμμα είσατε ο Guy Fawkes και αποστολή σας είναι να ανατινάξετε με μια βόμβα το κοινοβούλιο. Εδώ δεν θα εξετάσουμε κατά πόσο θα σας άρεσε ή όχι μια τέτοια ιδέα... αλλά θα σας δώσουμε τις απαραίτητες οδηγίες.

Ξεκινώντας από το κάτω αριστερό σημείο της οθόνης πρέπει να φθάσετε στη βόμβα που βρίσκεται πάνω-δεξιά στο λαβύρινθο. Στην αποστολή αυτή αντιμετωπίζετε δυο εχθρούς: Ο πρώτος είναι ένας φρουρός, που προσπαθεί να σας συλλάβει, και ο δεύτερος είναι ο χρόνος, μια και πρέπει να φτάσετε στη βόμβα σε

πολύ περιορισμένο χρονικό διάστημα. Αν τα καταφέρετε θα κερδίσετε επιπλέον χρόνο, μέσα στον οποίο πρέπει να τοποθετήσετε τη βόμβα στην πόρτα του κοινοβουλίου.

Τα πλήκτρα κίνησης του δρομέα χρησιμοποιούνται για την κίνηση σας μέσα στον λαβύρινθο. Αν δεν σας εξυπηρετούν μπορείτε να το αλλάξετε επεμβαίνοντας στις γραμμές 1610 και 1620. Αν πάλι το παιχνίδι σας φαίνεται δύσκολο μπορείτε να αυξήσετε το διαθέσιμο χρόνο αλλάζοντας την αρχική τιμή της μεταβλητής Time (γραμμές 18 και 3100, 3500).

Λόγω του αντικείμενου του παιχνιδιού, θα προτιμήσουμε να μη σας ευχηθούμε καλή τύχη...

## Η ΑΝΑΤΙΝΑΞΗ ΤΗΣ ... ΒΟΥΛΗΣ

```

9 RESTORE
10 GO SUB 9800
11 GO SUB 5825
15 RESTORE
16 LET Bomb=0: LET Ret=0
17 LET a=12: LET b=16
18 LET Time=99: LET x=20: LET
y=7
61 BEEP .05,0
990 BORDER 1: INK 1: PAPER 6: C
LS
1000 PRINT "
1010 PRINT "
1020 PRINT "
1030 PRINT "
1040 PRINT "
1050 PRINT "
1060 PRINT "
1070 PRINT "
1080 PRINT "
1090 PRINT "
1100 PRINT "
1110 PRINT "
1120 PRINT "
1130 PRINT "

```

```

1140 PRINT "
1150 PRINT "
1160 PRINT "
1170 PRINT "
1180 PRINT "
1190 PRINT "
1200 PRINT "
1210 PRINT "
1211 PRINT AT 0,0: INK 2: PAPER
7: "": AT 1,0: "": AT
2,0: "": AT 3,0: "": AT
T 4,0: "":
1220 PRINT AT x,y: INK 2: "I"
1229 PRINT AT a,b: INK 3: "E"
1230 PRINT AT 1,30: INK 0: "S"
1235 PRINT AT 21,7: PAPER 4: " "
1240 PRINT AT 0,23: INK 6: PAPER
1: "Time:": Time
1490 REM
1500 REM Main loop
1505 LET Time=Time-1: PRINT AT 0
,28: INK 7: PAPER 1: " ": PRINT
AT 0,28: INK 7: PAPER 1: Time: IF
Time=0 THEN GO TO 2000
1510 PRINT AT a,b: INK 1: "E"
1520 PRINT AT x,y: INK 2: "I"
1530 IF x=a AND y=b THEN GO TO 5
000
1501 IF x=3 AND y=3 AND Bomb=1 T
HEN GO SUB 3500
1502 LET xx=x: LET yy=y

```

```

1603 IF x=21 AND y=7 AND Ret=1 T
HEN GO SUB 4000
1610 LET x=x+(INKEY$="6" AND x<2
1)-(INKEY$="7" AND x>0)
1620 LET y=y+(INKEY$="8" AND y<3
1)-(INKEY$="5" AND y>0)
1623 IF y>21 AND x<1 THEN GO SUB
3000
1625 IF SCREEN$(x,y)=" " THEN GO
SUB 3000
1627 PRINT AT xx,yy;" "
1630 LET aa=a: LET bb=b
1635 IF AND>.5 THEN GO TO 1650
1640 IF a<x AND a<21 THEN LET a=
a+1
1645 IF a>x AND a>0 THEN LET a=a
-1
1647 IF AND>.5 THEN GO TO 1660
1650 IF b<y AND b<31 THEN LET b=
b+1
1655 IF b>y AND b>0 THEN LET b=b
-1
1660 IF SCREEN$(a,b)=" " THEN GO
SUB 1700
1670 PRINT AT aa,bb;" "
1675 GO TO 1500
1700 LET a=aa: LET b=bb: RETURN
2000 GO SUB 9990: BORDER 0: PAPE
R 0: INK 6: CLS
2010 PRINT AT 0,9;"Guy Fawkes";A
T 6,3;"You have run out of time
and have been blown up by
the bomb. Press "p" to
play again or "i" for
Instructions"
2020 BEEP .01,10: BEEP .01,20
2021 IF INKEY$="p" THEN GO TO 15
2022 IF INKEY$="i" THEN RUN
2023 GO TO 2020
3000 IF ATTR(x,y)=48 THEN GO TO
3100
3010 LET x=xx: LET y=yy: RETURN
3100 BEEP .33,10: LET Time=35: L
ET Bomb=1: RETURN
3500 BEEP .33,10: LET Time=80: L
ET Bomb=0: LET Ret=1: RETURN
4000 INK 2
4001 PRINT AT 0,0;" " ;AT 1,0
,0;" " ;AT 4,0;" " ;AT
3,0;" " ;AT 4,0;" " ;AT
4037 FOR a=15 TO -15 STEP -2
4039 FOR b=32 TO -22 STEP -5.5
4040 BEEP .03,-30: BEEP .03,-20
4041 PLOT 24,155: DRAW b,a
4043 NEXT b
4045 NEXT a
4046 FOR n=1 TO 500: NEXT n
4047 BORDER 0: PAPER 0: INK 6: C
LS: PRINT AT 2,9;"Guy Fawkes";A
T 7,3;"Congratulations you did i
t with ";Time;" units left"
: PRINT " Press "p" to play
again or "i" for Instruc
tions"
4048 BEEP .01,-10: BEEP .01,0: B
EEP .01,10
4049 IF INKEY$="p" THEN GO TO 15
4050 IF INKEY$="i" THEN RUN
4051 GO TO 4048
5000 GO SUB 9990: BORDER 0: PAPE
R 0: INK 6: CLS: PRINT AT 2,9;"
Guy Fawkes";AT 7,2;"The guard ca
ught you. You have been fo
und guilty of treason and
have been beheaded."
PRINT " Press "p" to play ag
ain or "i" for Instructio
ns"
5010 BEEP .01,0: BEEP .01,10
5011 IF INKEY$="p" THEN GO TO 15
5012 IF INKEY$="i" THEN RUN
5013 GO TO 5010
5025 GO SUB 6000
5030 PRINT " You are Guy Fawkes
and you have to blow-up the
house of parliament. You mus
t try to get to a bomb which is
in the right hand top corner of
the maze. You only have a lim

```

```

ited amount of time in which to
do this and there is also a
guard who you must avoid bein
g caught by."
5031 PRINT " Once you have the b
omb you will be given more
time. The next objective is t
o reach the house of parlia
ment."
5032 PRINT " AT 20,3; FLASH 1;"P
ress any key to continue"
5033 IF INKEY$="" THEN GO TO 503
3
5034 GO SUB 6000
5035 PRINT " Again you must avo
id running out of time and bei
ng caught by the guard. Once
you reach the house you must
move into the door. On doing
so you will be given more time
in which to return home."
5036 PRINT " INK 2;"#"; INK 6;"-
-----You"; PRINT INK 1
;"#"; INK 6;"-----Guard
"; PRINT "6";"-----Bomb"
5037 PRINT "5"-----Le
ft"; PRINT "8"-----Rig
ht"; PRINT "7"-----
Up"; PRINT "6"-----Do
wn"
5038 PRINT AT 21,6; FLASH 1;"Pre
ss any key to start"
5039 IF INKEY$="" THEN GO TO 503
9
5040 RETURN
6000 PAPER 0: INK 6: BORDER 0: C
LS
6010 PRINT AT 0,10;"Guy Fawkes";
AT 1,10;"-----";AT 4,2; INK
4;"Instructions"
6020 RETURN
9799 RETURN
9800 REM U.D.G.
9810 FOR n=0 TO 7
9815 READ udg: POKE USR "a"+n,ud
g: NEXT n
9900 FOR n=0 TO 7
9910 READ udg: POKE USR "b"+n,ud
g: NEXT n
9920 FOR n=0 TO 7
9930 READ udg: POKE USR "c"+n,ud
g: NEXT n
9932 FOR n=0 TO 7
9935 READ udg: POKE USR "d"+n,ud
g: NEXT n
9936 FOR n=0 TO 7
9937 READ udg: POKE USR "e"+n,ud
g: NEXT n
9938 FOR n=0 TO 7
9939 READ udg: POKE USR "f"+n,ud
g: NEXT n
9949 RETURN
9950 DATA 28,28,8,62,8,20,20,54,
92,92,72,126,72,84,20,54
9955 DATA 0,28,16,56,124,124,56,
0,230,131,194,129,126,129,128,16
7
9960 DATA 0,255,255,255,255,255,
255,255,238,187,238,187,238,187,
238,187
9990 BEEP .6,0: BEEP .4,0: BEEP
.2,0: BEEP .6,0: BEEP .4,3: BEEP
.2,2: BEEP .4,2: BEEP .2,0: BEE
P .4,0: BEEP .2,-1: BEEP .6,0: A
RETURN
9999 SAVE "Guy Fawkes" LINE 1
A : #
B : %
C : &
D : □
E : ■
F : ■

```

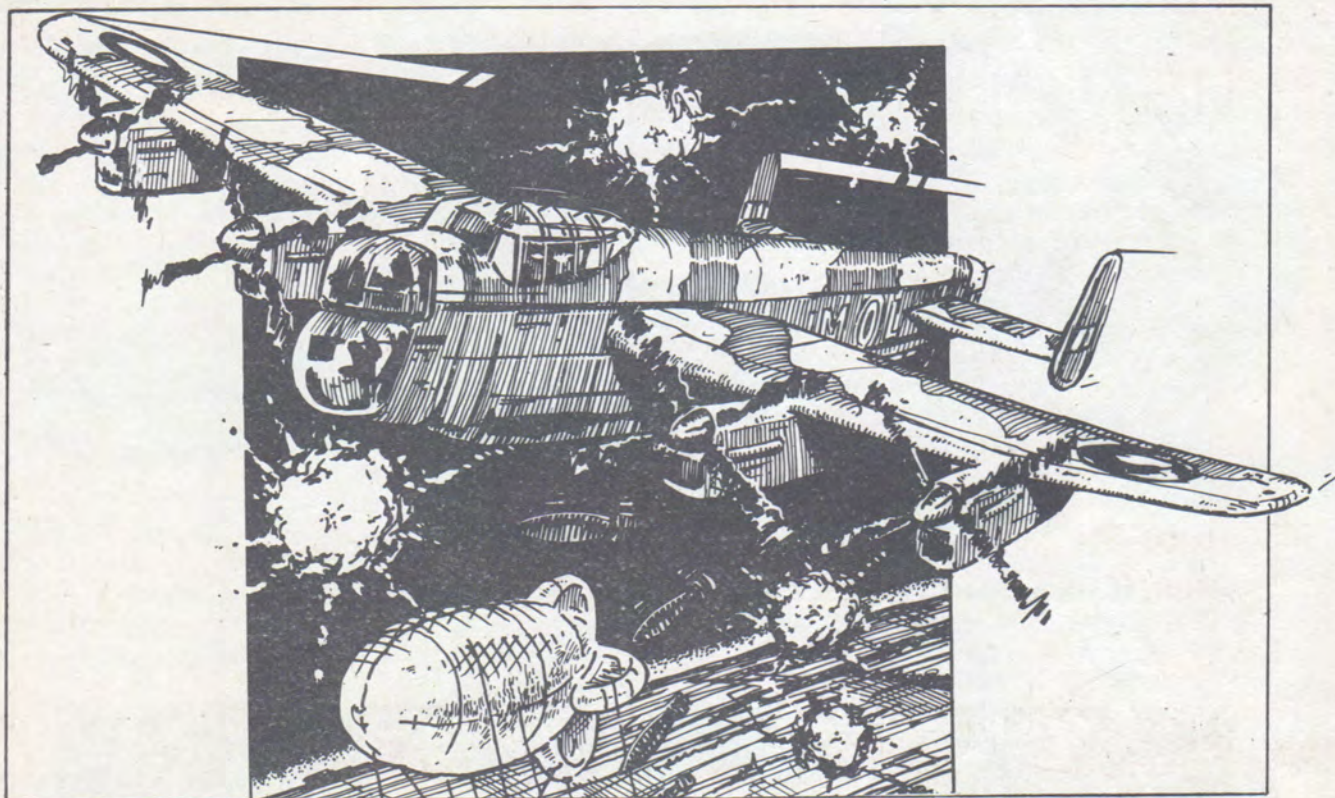
Προγράμματα για τόν:

ORIC

ΕΠΙΜΕΛΕΙΑ:

Δημήτρης Τσουροπλής





## ΕΠΙΚΙΝΔΥΝΗ ΠΡΟΣΓΕΙΩΣΗ

Το θέμα της καταστροφής μιας πόλης, ώστε να μπορέσει ένα αεροπλάνο να προσγειωθεί, φαίνεται να έχει πολύ προτίμηση από τους χρήστες-προγραμματιστές. Τα προγράμματα αυτού του τύπου έχουν το εξής "σενά-

ριο". Είστε πιλότος ενός βομβαρδιστικού που πετά πάνω από μια εγκαταλελειμμένη (ή όχι...) πόλη και χάνετε συνέχεια ύψος. Ο μόνος τρόπος να μπορέσει να προσγειωθεί είναι να ισοπεδώσει πρώτα την πόλη, καταστρέφοντας

με τις βόμβες τα διάφορα κτίρια. Στο συγκεκριμένο πρόγραμμα έχουμε σαν επιπλέον χαρακτηριστικό το ότι το σκάφος έχει περιορισμένα καύσιμα. Το παιχνίδι έχει διάφορα επίπεδα δυσκολίας και περιλαμβάνει και οδη-

γίες στα αγγλικά. Στην πληκτρολόγηση δεν πρέπει να υπάρξει πρόβλημα αλλά αν κάτι δεν πάει καλά ελέγξτε προσεκτικά αυτά που πληκτρολογήσατε γιατί κάτι, κάπου θα σας ξέφυγε.

```

90 PAPER4:INK3
99 PRINT CHR$(17)
100 TEXT:RELEASE:RESTORE
110 CLS:PLOT 5,10,"Do you want instructions?"
120 GET A$:IF A$="Y" THEN 300 ELSE 1000
130 REM
140 PLOT 11,9,12:PLOT 12,9,"PERFECT LANDING"
150 RESTORE
160 RETURN
170 REM
180 EXPLODE
190 PLOT 9,9,"Oh dear,you've crashed"
200 WAIT 500
210 RETURN
230 REM
240 PLOT 11,11,"Another try? (Y/N)"

```

```

245 GET A$
250 IF A$="Y" THEN RETURN
260 IF A$<>"N" THEN 245
270 PAPER0:INK7:CLS:PRINT CHR$(17):END
300 CLS:PRINT:PRINT SPC(15)"BOMB RUN"
310 PRINT:PRINT:PRINT "Your bomber is running out of fuel and you've got to find somewhere to land."
320 PRINT:PRINT:PRINT "To do this, the buildings below have got to be demolished."
340 PRINT:PRINT:PRINT "To increase height, press the 'cursor Up' key,but keep an eye on the fuel."
355 PRINT:PRINT:PRINT "Press 'cursor Down' key to release a bomb."

```

```

350 PRINT:PRINT:PRINT "Skill level (1=easy,5=hard)";
370 INPUT A:IF A<1 OR A>5 THEN 370 ELSE 1000
400 REM
410 CLS:GOSUB 700
450 FOR X=1 TO 35 STEP 2
460 FOR Y=INT(9+(RND(1)*15-D)) TO 26
470 PLOT X,Y,"[ ]"
480 NEXT Y:NEXT X:H=3
490 RETURN
500 REM
520 J=K+1:G=X:F=1:IF J>26 THEN RETURN
530 IF SCRN(G,J)=91 OR SCRN(G,J)=93 THEN W=W+1
540 PLOT G,J,"_"
550 RETURN
600 REM
610 PLOT G,J," "
620 IF W=2 THEN W=0:F=0:EXPLODE:RETURN
630 IF J=26 THEN W=0:F=0:EXPLODE:RETURN
640 J=J+1:B=B+1:GOTO 530
700 FOR CR=1 TO 6:READ C
705 FOR CH=#B400+8*C TO #B400+8*C+7:REAL R:POKE CH,R:NEXT NEXT
710 DATA 123,48,56,28,63,59,54,1,3
720 DATA 124,0,0,0,63,31,62,60,48
730 DATA 125,0,0,0,62,31,62,0,0
740 DATA 95,0,10,14,4,14,14,4
750 DATA 91,31,18,18,31,31,18,18,31
760 DATA 93,62,18,18,62,62,18,18,62

```

```

770 RETURN
8000 REM
1010 D=A:L=4:F=0:W=0:FL=30-INT((RND(1)*J*4+1)):GOSUB 400
1020 FOR X=1 TO 36
1030 K=H
1040 IF SCRN(X+2,H)<>#20 OR SCRN(X,H)<>#20 THEN L=L-1:PLOT X+3,H,"
":EXPLODE
1050 PLOT X,H,"( )":PLOT 1,0,2
1060 PLOT 25,0,"LIVES: ":PLOT 32,0,RIGHT$( " "+STR$(L),2)
1070 PLOT 3,0,"FUEL: ":PLOT 9,0,RIGHT$( " "+STR$(FL),2)
1080 IF L=0 THEN GOSUB 170:GOSUB 230:GOTO 100
1090 X$=KEY$
1100 IF X$=CHR$(10) AND F=0 THEN GOSUB 500 ELSE WAIT(10)
1110 IF X$=CHR$(11) AND FL>0 AND H>3 THEN H=H-1:FL=FL-1
1120 IF F=1 THEN GOSUB 600
1140 PLOT X,K,"
"
1150 IF H>25 AND X=35 THEN PLOT 35,26,"( )":GOSUB 130:A=D+1:GOSUB 230:GOTO 1000
1155 NEXT
1160 H=H+1
1170 GOTO 1020
140 ENTOLES

```

## ΚΑΛΕΙΔΟΣΚΟΠΙΟ

Τα δυο αυτά σύντομα προγράμματα για δημιουργία γραφικών παραστάσεων, μπορούν να ενσωματωθούν σε δικά σας προγράμματα ή να "τρέχουν" από μόνα τους. Και τα δυο εκμεταλλεύονται την υψηλή ανάλυση και τα χρώματα του ORIC με όμορφα αποτελέσματα ειδικά σε έγχρωμη TV. Το πρώτο πρόγραμμα δημιουργεί την εντύπωση της κίνησης μέσα στο διάστημα, και συνοδεύεται

από ταιριαστό θόρυβο. Το δεύτερο πρόγραμμα δημιουργεί ένα καλειδοσκοπικό εφέ με το να σχεδιάζει ένα σχέδιο στην οθόνη και μετά να του αλλάζει χρώματα (κάνοντας POKE κωδικούς attributes στο display file). Η εντολή PRINT CHR\$(17) εξαφανίζει τον δρομέα και για να τον επαναφέρετε μπορείτε να πατήσετε απευθείας CTRL Q αφού διακόψετε το πρόγραμμα.

```

5 HIMEM#2000
10 HIRES
20 INK 4:PAPER4
30 CURSET 0,0,0:DRAW 239,199,1
40 CURSET 239,0,0:DRAW -239,199,1
50 CURSET 120,0,0:DRAW 0,199,1
70 FOR Y=99 TO 0 STEP -3
90 CURSET 7,Y,0:FILL 2,1,7
85 CURSET 7,99+(99-Y),0:FILL 2,1,7
90 CURSET 7,Y,0:FILL 2,1,4
95 CURSET 7,99+(99-Y),0:FILL 2,1,4
100 PLAY 0,3,0,0:SOUND 4,Y,7
110 NEXT
120 GOTO 0

```

```

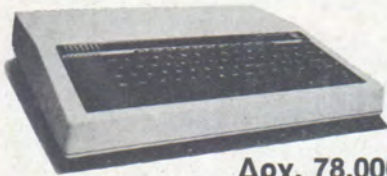
20 PAPER 0:CLS:HIRES:PRINT CHR$(17)
30 FOR A=239 TO 0 STEP -1:CURSET A,0,3:DRAW 239-A*2,199,2:NEXT
40 FOR A=0 TO 199:CURSET 0,A,3:DRAW 239,199-A*2,2:NEXT
50 FOR A=0 TO 7:PAPER A:FOR B=0 TO 7:FOR C=16 TO 22:IF B+16=C THEN C=C+1
60 FOR D=40960 TO 49000 STEP 80:POKE D,B
70 POKE D+1,C:NEXT D:WAIT 100:NEXT C,B,A
80 GOTO 50

```

# Personal Computers

## BBC MODEL B

- ★ 32K RAM
- ★ 16 ΧΡΩΜΑΤΑ - ΗΧΟΥΣ
- ★ A/D Converter
- ★ ΓΛΩΣΣΕΣ ΠΡΟΓΡΑΜΜΑΤΙΣΜΟΥ BASIC, PASCAL, FORTH
- ★ ΔΥΝΑΤΟΤΗΣ ΕΠΕΚΤΑΣΕΩΣ



Δρχ. 78.000

## SINCLAIR ZX SPECTRUM

- ★ 16 ή 48 K RAM
- ★ 8 ΧΡΩΜΑΤΑ - ΗΧΟ
- ★ ΜΕΓΑΛΗ ΠΟΙΚΙΛΙΑ ΣΕ ΠΡΟΓΡΑΜΜΑΤΑ
- ★ High res Graphics



Δρχ. 25.000

## SINCLAIR ZX 81

- ★ 1K expandable to 64 K RAM
- ★ Graphics
- ★ ΜΕΓΑΛΗ ΠΟΙΚΙΛΙΑ ΣΕ ΠΡΟΓΡΑΜΜΑΤΑ



Δρχ. 12.900

## ORIC - 1

- ★ 48 K RAM
- ★ High res Graphics
- ★ 8 ΧΡΩΜΑΤΑ - ΗΧΟΥΣ
- ★ Centronics Interface



Δρχ. 32.000

## COMMODORE 64

- ★ 64 K RAM + 20K ROM
- ★ 16 ΧΡΩΜΑΤΑ
- ★ Sound Synthesizer
- ★ Sprites
- ★ CP/M



Δρχ. 73.700

## NEWBRAIN

- ★ 32 K RAM - 2M RAM
- ★ High res 60 x 250
- ★ High res 640x250
- ★ CP/M
- ★ ΜΕ ΕΛΛΗΝΙΚΟΥΣ ΧΑΡΑΚΤΗΡΕΣ



Δρχ. 48.000

## IBM PC



## PRINTERS

SEIKOSHA GP 250  
SEIKOSHA GP 700  
EPSON



## MONITORS HANTAREX

- ★ 12" ΠΡΑΣΙΝΟ ή ΚΙΤΡΙΝΟ  
Δρχ. 22.000
- ★ 14" ΕΓΧΡΩΜΟ  
Δρχ. 49.000



- ★ TRS - 80 COLOUR COMPUTER extended Basic..... Δρχ. 39.000
- ★ Jupiter Ace language FORTH..... Δρχ. 18.000
- ★ ΔΙΣΚΕΤΤΕΣ 5 1/4" VERBATIM DATALIFE ..... Δρχ. 450
- ★ ΚΑΣΣΕΤΤΕΣ LOW NOISE ΜΙΚΡΗΣ ΔΙΑΡΚΕΙΑΣ..... Δρχ. 100
- ★ ΧΑΡΤΙ ΜΗΧΑΝΟΓΡΑΦΗΣΗΣ - ΜΕΛΑΝΟΤΑΙΝΙΕΣ
- ★ ΜΕΓΑΛΗ ΠΟΙΚΙΛΙΑ ΣΕ ΠΡΟΓΡΑΜΜΑΤΑ ΚΑΙ ΠΑΙΧΝΙΔΙΑ
- ★ ΠΛΗΡΗΣ ΣΕΙΡΑ ΠΕΡΙΦΕΡΕΙΑΚΩΝ ΓΙΑ ΟΛΟΥΣ ΤΟΥΣ ΥΠΟΛΟΓΙΣΤΕΣ

ΣΤΕΛΝΟΥΜΕ ΜΕ ΑΝΤΙΚΑΤΑΒΟΛΗ Σ' ΟΛΗ ΤΗΝ ΕΛΛΑΔΑ

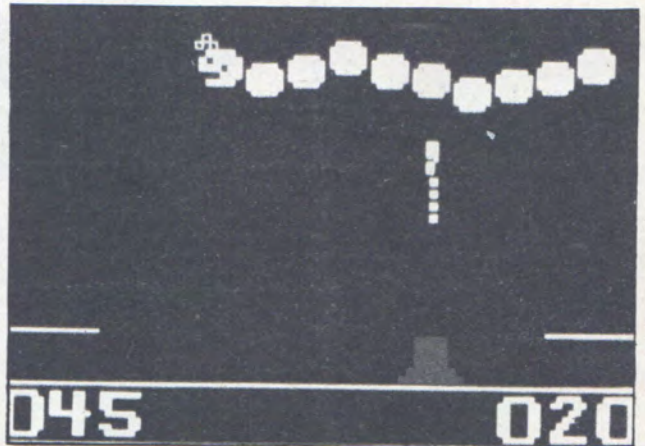


# Athens Computer Centre

Σολωμού 26 - Αθήνα 10 682 - τηλ.: 3609217 - TLX.: 214592 PRMN GR

Το παιχνίδι αυτό σε στυλ "arcade-game" εκμεταλλεύεται τα graphics και τον ήχο του ORIC-1 για να σας προσφέρει μια παραλλαγή της γνωστής "Σαρανταποδαρούσας". Οπλισμένος με έναν εκτοξευτή βλημάτων, βρισκόσαστε αντιμέτωπος μ'ένα σωρό περιεργούς εχθρούς, από μύγες μέχρι... μανιτάρια και πρέπει να τους εξοντώσετε όσο προλαβαίνετε. Ανάλογα με το τι σκοτώνετε παίρνετε από 25 μέχρι 1000 βαθμούς. Προσοχή στις πέτρες (μένουν στη θέση των μανιταριών όταν κτυπηθούν) γιατί κάθε βολή σ'αυτές σας αφαιρεί 1000 πόντους. Τα πλή-

κτρα κίνησης τα ορίζετε εσείς στην αρχή του παιχνιδιού όταν το πρόγραμμα σας το ζητήσει. Η σαρανταποδαρούσα που εμφανίζεται είναι ακίνδυνη μια και δεν κατεβαίνει μέχρι κάτω, αλλά δίνει πολλούς πόντους όταν την κτυπήσετε και ιδιαίτερα στο κεφάλι. Για τους υπόλοιπους κινδύνους σας αφήνουμε να τα βρείτε μόνοι σας γιατί αλλιώς δεν θάχε νόημα ο κήπος των εκπλήξεων... Η φράση τέλος "Once in every lifetime..." έχει ειρωνικό χαρακτήρα και είναι σαν σχόλιο όλων αυτών που συμβαίνουν στον κήπο!



## Ο ΚΗΠΟΣ ΤΩΝ ΕΚΠΛΗΞΕΩΝ

```
0 GOSUB 50000
1 CLS:GOSUB 60000
2 FOR X=1 TO 18:READ N,W
3 MUSIC 1,3,N,15
4 WAIT W*20:NEXT X
5 DATA 3,2,6,3,8,2,10,3,11,1,10,2,8,4,5,
6,1,3,3,1,5,2
6 DATA 6,4,3,2,3,3,2,1,3,2,5,4,2,2
7 PRINT CHR$(6):PRINT CHR$(17):EXPLODE
8 GOSUB 60000
9 INK 7:PAPER0
10 K$=KEY$
11 PLOT 1,1,"Once in every lifetime.
"
17 SC$=STR$(SC)
18 PLOT 2,0,SC$
19 PLOT 1,11,T$
20 IF K$=L$ THEN X=X-1
30 IF K$=R$ THEN X=X+1
32 IF X>32 THEN X=32
34 IF X<7 THEN X=7
40 IF K$=F$ THEN GOSUB 10000
50 IF K$=L$ THEN PLOT X+1,25," "
50 IF K$=R$ THEN PLOT X-1,25," "
```

```
70 PLOT X,25,"@"
75 IF SCRN(X,24)<>32 THEN GOTO 40000
77 GOSUB 10000
78 PLOT 1,26,"
"
80 GOTO 10
1000 FOR D=24 TO 13 STEP -1
1001 MUSIC 1,5,1,0:PLAY 7,0,1,50
1003 PLOT X,D,"!"
1006 IF SCRN(X,D-1)=125 THEN D=0
1008 IF SCRN(X,D-1)=125 THEN SC=SC+700
1009 IF SCRN(X,D-1)=94 THEN SC=SC-1000
1010 IF SCRN(X,D-1)=35 THEN SC=SC+50
1020 IF SCRN(X,D-1)=37 THEN SC=SC+100
1025 IF SCRN(X,D-1)=38 THEN SC=SC+25
1030 IF SCRN(X,D-1)=91 THEN SC=SC+300
1040 IF SCRN(X,D-1)=93 THEN SC=SC+1500
1050 IF SCRN(X,D-1)=123 THEN SC=SC+1000
1055 IF SCRN(X,D-1)<>32 THEN PLOT X,D-1
"@"
1060 IF SCRN(X,D-1)<>32 THEN 1080
1070 PLOT X,D," "
1074 PLOT 2,0,SC$
1077 NEXT D
```



```

1080 FOR T=24 TO D-1 STEP -1
1085 IF SCRNX(X,T-1)=37 THEN PLOT X,T-1,"
^"
1087 IF SCRNX(X,T)=37 THEN RETURN
1090 PLOT X,T,"
1100 NEXT T
1105 SHOOT
1107 WAIT 10
1110 RETURN
6000 PAPER 0:INK 7
6002 FOR SW=1 TO 12:PLOT 4,SW,17:NEXT SW
6010 FOR SH=13 TO 26:PLOT 4,SH,20:NEXT S
H
6020 PLOT 3,25,0
6025 FOR RT=1 TO 24
6028 GJ=INT(RND(1)*7)
6030 IF GJ=1 THEN GJ=0
6032 IF GJ=4 THEN GJ=7
6034 IF GJ=5 THEN GJ=2
6037 PLOT 3,RT,GJ
6038 NEXT RT
6050 RETURN
10000 K$=KEY$
10005 PLOT 1,1,"Once in every lifetime..
"
10010 PLOT 1,26,"
"
10020 IF K$=L$ THEN X=X-1
10030 IF K$=R$ THEN X=X+1
10040 IF K$=F$ THEN GOSUB 1000
10042 IF X<7 THEN X=7
10043 IF X>32 THEN X=32
10050 IF K$=R$ THEN PLOT X+1,25," "
10060 IF K$=R$ THEN PLOT X-1,25," "
10075 MUSIC 1,6,12,1
10080 IF SCRNX(X,24)<>32 THEN 40000.
10090 B=INT(RND(1)*30)+7
10092 C=INT(RND(1)*23)+1
10100 IF B=15 THEN GOSUB 30000
10105 IF B=17 THEN GOSUB 35000
10106 U=INT(RND(1)*27)+5
10107 IF B=18 THEN GOSUB 25000
10108 IF B=20 THEN GOSUB 15000
10110 PLOT B,C,"#"
10120 PLOT B,C-1,"x"
10125 PLOT X,25,"@"
10127 K=INT(RND(1)*20)+7
10130 RETURN
15000 FOR Q=0 TO 26 STEP INT(RND(1)*5)+1
15002 MUSIC 1,0,1,0:MUSIC 2,1,1,0:MUSIC
3,2,1,0:PLAY 7,0,5,100
15005 PLOT 1,1,"Once in every lifetime..
"
15010 K$=KEY$
15015 SE=INT(RND(1)*100):IF SE<10 AND SE
>5 THEN RETURN
15020 IF K$=L$ THEN X=X-1
15030 IF K$=R$ THEN X=X+1

```

```

15032 IF X<7 THEN X=7
15034 IF X>32 THEN X=32
15040 IF K$=L$ THEN PLOT X+1,25," "
15050 IF K$=R$ THEN PLOT X-1,25," "
15060 PLOT X,Q,")"
15070 IF K$=F$ THEN GOSUB 1000
15080 IF SCRNX(X,24)<>32 THEN 40000
15090 PLOT X,Q," "
15092 PLOT X,25,"@"
15093 PLOT 2,0,SC$
15095 NEXT Q
15097 PLOT 2,0,SC$
15100 RETURN
24050 U=INT(RND(1)*27)+5
25000 FOR D=4 TO 26 STEP INT(RND(1)*2)+1
"
25003 MUSIC1,5,(D/2)-1,0:MUSIC2,6,(D/2)-
1,0:MUSIC3,7,(D/2)-1,0:PLAY
7,2,3,100
25005 K$=KEY$
25007 PLOT 1,1,"Once in every lifetime..
...
"
25010 IF K$=L$ THEN X=X-1
25020 IF K$=R$ THEN X=X+1
25030 IF X<7 THEN X=7
25040 IF X>32 THEN X=32
25050 IF K$=L$ THEN PLOT X+1,25," "
25060 IF K$=R$ THEN PLOT X-1,25," "
25076 PLOT 2,0,SC$
25080 PLOT U,D,")"
25085 IF SCRNX(X,24)<>32 THEN 40000
25087 IF K$=F$ THEN GOSUB 1000
25090 PLOT U,D," "
25100 PLOT X,25,"@"
25200 NEXT D
25205 PLOT 1,26,"
"
25206 FOR UI=0 TO 26:PLOT U,UI," ":NEXT
UI
25210 RETURN
30000 FOR G=2 TO 23 STEP 2
30001 NJ=INT(RND(1)*27)+9:PLOT NJ,G+2,"*
:PLOT NJ+3,G+2,"*":PLOT NJ
-3,G+2,"*"
30002 PLOT INT(RND(1)*30)+7,INT(RND(1)*2
5)+1,"x"
30005 FOR H=10 TO 31
30006 PLOT INT(RND(1)*30)+7,INT(RND(1)*2
4)+1,"#"
30007 K$=KEY$
30008 SC$=STR$(SC)
30009 PLOT 2,0,SC$
30010 IF K$=L$ THEN X=X-1
30020 IF K$=R$ THEN X=X+1
30030 IF K$=F$ THEN GOSUB 1000
30040 IF X<7 THEN X=7
30050 IF X>32 THEN X=32
30060 IF SCRNX(X,24)<>32 THEN 40000

```

```

30070 IF K$=L$ THEN PLOT X+1,25," "
30075 IF K$=R$ THEN PLOT X-1,25," "
30076 MUSIC 1,1,G/2,0:MUSIC2,3,G/2,0:PLA
Y 7,0,1,400
30079 PLOT 1,1,"Once in every lifetime..

30085 PLOT X,25,"@"
30090 PLOT H,G,"J"
30100 PLOT H-1,G,"["
30105 PLOT H-1,G-2," "
30107 PLOT H+1,G-2,"."
30108 U=INT(RND(1)*20)+7:IF U=15 THEN GO
SUB 24050
30109 IF U=16 THEN GOSUB 15000
30110 NEXT H
30120 NEXT G
30125 PLOT 5,22,"

30130 RETURN
35000 PLOT 28,0,"Flee storm!"
35001 MUSIC 1,7,3,0:PLAY 7,0,1,2000
35002 FOR U=1 TO 26
35005 K=INT(RND(1)*20)+7
35010 K$=KEY$
35015 PLOT 1,1,"Once in every lifetime..

35020 IF K$=L$ THEN X=X-1
35030 IF K$=R$ THEN X=X+1
35040 IF K$=F$ THEN GOSUB 1000
35050 IF X<7 THEN X=7
35060 IF X>32 THEN X=32
35070 IF K$=R$ THEN PLOT X-1,25," "
35080 IF K$=L$ THEN PLOT X+1,25," "
35090 PLOT X,25,"@"
35100 PLOT K,U,"&"
35105 PLOT K,U-1," "
35110 SC$=STR$(SC)
35120 PLOT 2,0,SC$
35125 NEXT U
35127 PLOT 27,0,"
35130 RETURN
40000 PLOT X,25,"J"
40030 EXPLODE:PLAY 7,7,1,65000
40040 WAIT 700
40050 CLS:INK 0:PAPER 7
40060 PRINT :PRINT "Your score was "SC$
40080 PRINT:PRINT:PRINT "Would you like
to play again?"
40085 GET C$
40090 GET B$
40095 PRINT CHR$(6):PRINT CHR$(17)
40100 IF B$="Y" THEN RUN
40110 END
40480 RETURN
50000 CLS
50005 PING
50008 PRINT:PRINT"Which key would you li

```

```

ke to use?"
50010 PRINT:PRINT"a.To move left?":GET L
$
50020 PRINT:b.To move right?":GET R$
50030 PRINT"c.To fire?":GET F$
50040 PRINT:PRINT"THANK YOU"
50050 WAIT 200
50060 RETURN
60000 FOR Y1=46376 TO 46383
60010 READ Y2:POKE Y1,Y2
60020 DATA 0,30,53,45,53,53,43,33
60030 NEXT Y1
60040 FOR Y3=46360 TO 46367
60050 READ Y4:POKE Y3,Y4
60060 DATA 51,18,22,18,22,26,18,33
60070 NEXT Y3
60080 FOR Y5=46592 TO 46599
60090 READ Y6:POKE Y5,Y6
60100 DATA 12,12,30,63,63,63,30,30
60110 NEXT Y5
60120 FOR Y6=46808 TO 46815
60130 READ Y7:POKE Y6,Y7
60140 DATA 30,63,63,63,63,63,30
60150 NEXT Y6
60170 FOR Y8=46824 TO 46831
60180 READ Y9:POKE Y8,Y9
60190 DATA 1,61,54,60,54,61,1,0
60200 NEXT Y8
60210 FOR Z1=46832 TO 46839
60220 READ Z2:POKE Z1,Z2
60230 DATA 4,30,63,62,62,62,24,8
60250 NEXT Z1
60260 FOR Z3=46416 TO 46423
60270 READ Z4
60280 POKE Z3,Z4
60290 DATA 33,51,12,12,30,45,33,0
60300 NEXT Z3
60310 FOR Z5=46384 TO 46391
60320 READ Z6:POKE Z5,Z6
60330 DATA 8,42,62,62,62,62,42,0
60340 NEXT Z5
60360 FOR Z7=47064 TO 47071
60370 READ Z8:POKE Z7,Z8
60380 DATA 18,33,33,18,12,63,45,63
60390 NEXT Z7
60400 FOR Z9=47080 TO 47087
60410 READ A1:POKE Z9,A1
60420 DATA 0,0,12,30,63,63,30,12
60430 NEXT Z9
60440 FOR B6=47072 TO 47079
60450 READ B7:POKE B6,B7
60460 DATA 12,30,63,12,12,12,30,12
60470 NEXT B6
60480 FOR B8=46344 TO 46351
60490 READ B9:POKE B8,B9
60500 DATA 33,18,12,29,46,12,18,33
60510 NEXT B8
60520 RETURN

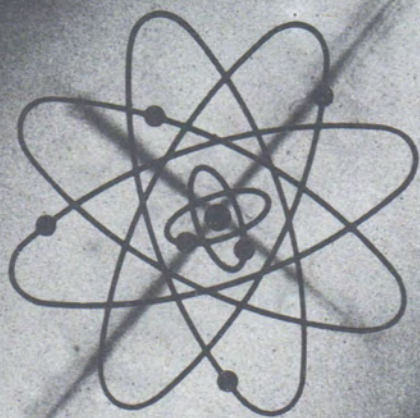
```

# INDEX-EL '83

3η ΕΚΘΕΣΗ

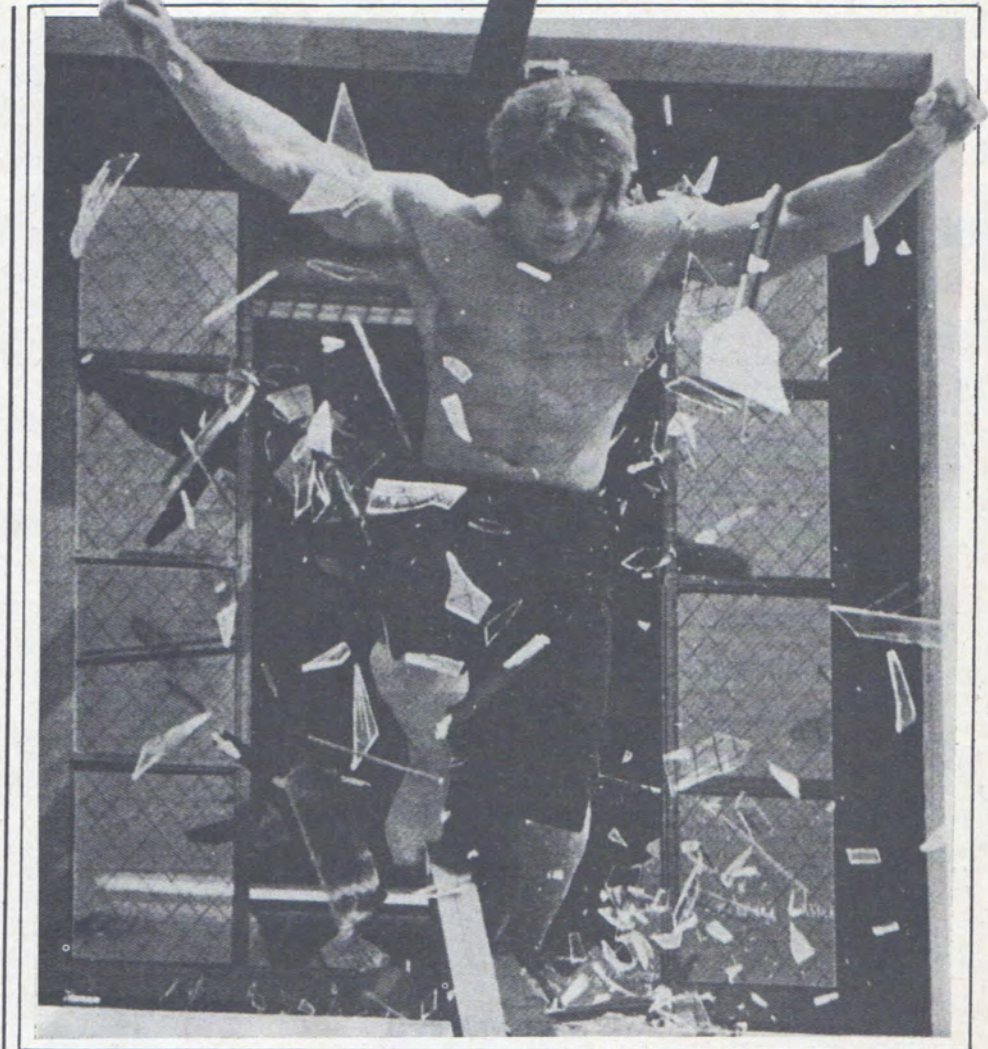
ΗΛΕΚΤΡΟΛΟΓΙΚΟΥ &  
ΗΛΕΚΤΡΟΝΙΚΟΥ

ΕΞΟΠΛΙΣΜΟΥ



18-23 ΝΟΕΜΒΡΙΟΥ 1983 ΠΕΙΡΑΙΑΣ

Το παιχνίδι αυτό με τον αγγλικό τίτλο "BREAKOUT" είναι από τα παλιότερα και γνωστότερα προγράμματα για προσωπικούς υπολογιστές. Πρέπει να γκρεμίσετε έναν τοίχο από τούβλα χρησιμοποιώντας ένα μπαλάκι το οποίο κινείται μέσα στην οθόνη όσο μπορείτε να το συγκρατείτε με τη ρακέτα σας. Η κίνηση της ρακέτας αριστερά-δεξιά γίνεται με τα αντίστοιχα πλήκτρα κίνησης του δρομέα, αλλά προσέξτε γιατί η ρακέτα διατηρεί κάθε φορά την τελευταία κίνηση που έκανε. Αν και όλο γραμμένο σε BASIC, το παιχνίδι είναι αρκετά γρήγορο ώστε να μην γίνεται βαρετό και σ' αυτό βοηθάει και ο τρόπος κίνησης της ρακέτας, που περιγράψαμε πιο πάνω. Από το πρόγραμμα μπορείτε επίσης να πάρετε χρήσιμες ιδέες για προγραμματισμό σχετικά με τους user defined χαρακτήρες, με τις PEEK, PLOT κ.ά. αν βέβαια δεν σας ενδιαφέρει μόνον το ίδιο το παιχνίδι!



## ΤΟ ΓΚΡΕΜΙΣΜΑ ΤΟΥ ΤΟΙΧΟΥ

```
5. HS=0:SC=0
8. IF SC>HS THEN HS=SC
10. RESTORE
20. FOR I=0 TO 7:READ A
30. POKE 46840+I,A
```

```
40. NEXT I
50. DATA 64,76,94,127,127,94,76,64
60. FOR I=0 TO 7
70. READ B
80. POKE 46376+I,B
90. NEXT
```

```

100 DATA 64,64,64,124,124,76,76,76
110 FOR I=0 TO 7
120 READ C
130 POKE 46384+I,C
140 NEXT I
150 DATA 64,64,64,79,79,76,76,76
160 FOR I=0 TO 7
170 READ D
180 POKE 47080+I,D
190 NEXT I
200 DATA 64,64,64,127,127,64,64,64
210 FOR I=0 TO 7
220 READ E
230 POKE 47064+I,E
240 NEXT I
250 DATA 76,76,76,76,76,76,76,76
260 FOR I=0 TO 7
270 READ F
280 POKE 46832+I,F
290 NEXT I
295 DATA 0,0,0,0,0,0,0,0:GOSUB 20000
300 IF M=0 THEN M=1:GOSUB 21000
301 LORES 0:GOSUB 12000
302 FOR I=26 TO 1 STEP -1
303 PLOT 1,1,"{"
304 NEXT I
310 PLOT 1,1,"&"
320 FOR I=2 TO 37
330 PLOT I,1,")"
340 NEXT I
350 PLOT 38,1,"x"
360 FOR I=2 TO 26
370 PLOT 38,I,"{" :NEXT I
371 GOSUB 6000:PLOT 0,25,CHR$(6):BN=3:SC
=0:B=0:GOSUB 13000:GOTO 101
7
372 M1=17:M2=M1+1:M3=M1+2
373 PLOT M1,25,")"
374 PLOT M2,25,")"
375 PLOT M3,25,")"
385 I=48620
387 D=RND(1)+.5:ON D GOTO 810
390 POKE I,95
395 GOSUB 10050
397 GOSUB 10050
399 I=I+41:IF PEEK(I)<33 THEN 440
400 GOSUB 5000
402 IF PEEK(I)=123 THEN I=I-41:GOTO 810
405 IF PEEK(I)=125 THEN I=I-41:D=1:GOTO
1050
410 IF PEEK(I)=91 THEN GOSUB 8000:I=I-41
:GOTO 700
420 IF PEEK(I)=93 THEN GOSUB 7000:I=I-41
:GOTO 700
440 IF I>#BF8 THEN 1000
450 POKE I-41,32
470 GOTO 390
600 POKE I,95
605 GOSUB 10050

```

```

610 GOSUB 10050
620 I=I-41:IF PEEK(I)<33 THEN 680
625 GOSUB 5000
630 IF PEEK(I)=125 THEN I=I+41:GOTO 810
640 IF PEEK(I)=123 THEN I=I+41:GOTO 700
650 IF PEEK(I)=38 THEN I=I+41:GOTO 390
660 IF PEEK(I)=91 THEN GOSUB 8000:I=I+41
:GOTO 810
670 IF PEEK(I)=93 THEN GOSUB 7000:I=I+41
:GOTO 810
680 POKE I+41,32
690 GOTO 600
700 POKE I,95
710 GOSUB 10050:GOSUB 10050
715 I=I-39:IF PEEK(I)<33 THEN 740
716 GOSUB 5000
720 IF PEEK(I)=125 THEN I=I+39:GOTO 390
730 IF PEEK(I)=123 THEN I=I+39:GOTO 600
732 IF PEEK(I)=37 THEN I=I+39:GOTO 810
733 IF PEEK(I)=91 THEN GOSUB 8000:I=I+39
:GOTO 390
734 IF PEEK(I)=93 THEN GOSUB 7000:I=I+39
:GOTO 390
740 POKE I+39,32
750 GOTO 700
810 POKE I,95
820 GOSUB 10050:GOSUB 10050
825 I=I+39:IF PEEK(I)<33 THEN 850
827 GOSUB 5000
830 IF PEEK(I)=123 THEN I=I-39:GOTO 390
840 IF PEEK(I)=125 THEN I=I-39:D=0:GOTO
1050
841 IF PEEK(I)=91 THEN GOSUB 8000:I=I-39
:GOTO 600
842 IF PEEK(I)=93 THEN GOSUB 7000:I=I-39
:GOTO 600
850 IF I>#8FB8 THEN 1000
870 POKE I-39,32
880 GOTO 810
1000 FOR I=2 TO 37:PLOT I,25," " :NEXT
1010 GOSUB 5000:WAIT 200
1015 BN=BN-1:IF BN=0 THEN Z$="GAME OVER"
:PLOT 16,15,Z$:GOTO 8
1017 Z$="balls remaining = "+STR$(BN)+"
score"
1018 PLOT 1,0,Z$
1020 GOTO 372
1050 N=I:IF B=108 THEN GOSUB 6000:B=0
1055 J=M1+#BF91-40:I=N
1056 IF I=J-1 THEN 600
1060 IF I=J THEN 600
1065 IF I=J+1 AND D=1 THEN 700
1070 IF I=J+1 AND D=0 THEN 600
1080 IF I=J+2 THEN 700
1090 GOTO 700
5000 SOUND 1,80,0
5010 PLAY 1,0,1,200
5020 RETURN
6000 REM

```

```

6005 FOR J=0 TO 5
6010 FOR I=2 TO 37 STEP 2
6030 PLOT 1,J+6,"["
6040 PLOT 1+1,J+6,"]"
6050 NEXT I
6052 Tz=J/2:ON Tz GOTO 6056,6057
6055 PLOT 0,J+6,CHR$(1):GOTO 6060
6056 PLOT 0,J+6,CHR$(2):GOTO 6060
6057 PLOT 0,J+6,CHR$(4)
6060 NEXT J:RETURN
7000 POKE 1,32:POKE 1-1,32
7050 Jx=(1-#BB81)/40:GOSUB 9000
7080 B=B+1
7100 RETURN
8000 POKE 1,32:POKE 1+1,32
8050 Jx=(1-#BB80)/40:GOSUB 9000
8080 B=B+1
8100 RETURN
9000 Tz=(14-Jx)/2:SC=SC+5*Tz
9010 PLOT 30,0,STR$(SC):RETURN
10050 A=PEEK(#2DF)AND 127
10060 IF A=8 THEN 10100
10070 IF A=9 THEN 10150
10080 RETURN
10100 IF M1=2 THEN RETURN
10110 M1=M1-1
10120 PLOT M1+3,25," " :PLOT M1,25,")"
10130 RETURN
10150 IF M1+2=37 THEN RETURN
10160 M1=M1+1:PLOT M1,25," "
10170 PLOT M1+2,25,")"
10180 RETURN
12000 Z$=CHR$(16)+CHR$(1)+CHR$(12)+
BREAKOUT"+CHR$(8)+CHR$(3)
12010 Z$=Z$+" high score "+STR$(HS)
12015 FOR I=0 TO 39:POKE #BB80+I,32:NEXT
12020 FOR I=1 TO LEN(Z$)
12030 POKE #BB7F+1,ASC(MID$(Z$,I,1))
12040 NEXT
12050 RETURN
13000 POKE #2DF,0
13010 Z$=CHR$(12)+"PRESS ANY KEY TO CONT
INUE"+CHR$(8)
13020 Q$="R"=resign"+CHR$(8)
13030 PLOT 6,16,Z$:PLOT 15,18,Q$
13040 GET A$:IF A$="R" THEN HIRES:TEXT:EN
D
13050 PLOT 6,16,"
" :PLOT 15,17,"

13055 PLOT 13,18,"
13060 WAIT 150:RETURN
20000 FOR I=0 TO 7
20010 READ H
20020 POKE 46808+I,H
20030 NEXT I
20040 DATA 64,255,255,255,255,255,255,25
5
20050 FOR I=0 TO 7
20060 READ N
20070 POKE 46824+I,N
20080 NEXT I

```

```

20030 DATA 64,251,254,254,254,254,254,25
1
20035 FOR I=1 TO 20:PLOT 0,1,5:NEXT
20100 RETURN
21000 TEXT:PAPER0:INK2:CLS
21010 PRINT
21020 PRINT CHR$(4);CHR$(27);"J";CHR$(27
);"A *** ";CHR$(27);"
N";
21025 PRINT CHR$(27);"E BREAKOUT";
21030 PRINT CHR$(27);"A";CHR$(27);"J ***
";CHR$(27);"G";CHR$(4)
21035 PRINT:PRINT:PRINT:PRINT
)))))))))
21055 PRINT:PRINT:PRINT:PRINT:PRINT:PRIN
T" POINTS":P
RINT
21060 PRINT " ";CHR$(27);"A [ ] [ ]....
... 15 points":PRINT
21070 PRINT " ";CHR$(27);"B [ ] [ ]....
... 10 points":PRINT
21080 PRINT " ";CHR$(27);"D [ ] [ ]....
... 5 points":PRINT
21085 PRINT:PRINT:PRINT:PRINT" )))
)))))))))
21090 PRINT:PRINT" PRESS ANY KEY TO C
ONTINUE":GET A$
21091 TEXT:PAPER0:INK2:CLS
21092 PRINT
21093 PRINT CHR$(4);CHR$(27);"I";CHR$(27
);"A *** ";CHR$(27);"
N";
21094 PRINT CHR$(27);"E BREAKOUT";
21095 PRINT CHR$(27);"A";CHR$(27);"J ***
";CHR$(27);"G";CHR$(4)
21096 PRINT:PRINT:PRINT:PRINT" )))
)))))))))
21100 PRINT:PRINT
22030 PRINT " ";CHR$(27);"B KEYBOA
RD CONTROLS"
22040 PRINT:PRINT" ( ).....
... left":PRINT
22050 PRINT " (SPACE BAR).....,s
top":PRINT
22055 PRINT " )).....,s
right":PRINT
22060 PRINT:PRINT" )))))))
))))))
" :PRINT
22065 PRINT:PRINT
22070 PRINT" s your cursor on ?":
GET A$
22080 IF A$=")" THEN PRINT CHR$(17):GOTO
22100
22090 IF A$("<"N" THEN GETA$:GOTO 22080
22100 PRINT:PRINT" s your key cli
ck on ?":GET A$
22110 IF A$="Y" THEN PRINT CHR$(6):CLS:R
ETURN
22120 IF A$("<"N" THEN GETA$:GOTO 22110
22130 CLS:RETURN

```

— Προγράμματα για τόν: —

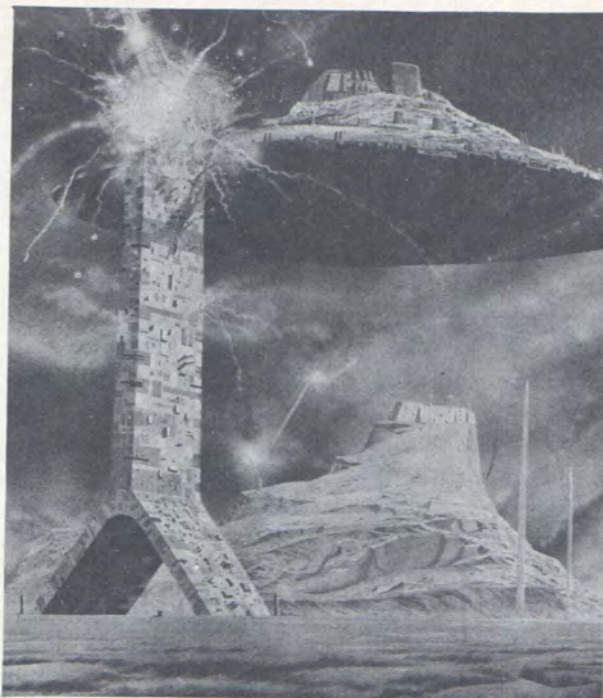
TI 99/4A



ΕΠΙΜΕΛΕΙΑ:

Δημήτρης Τσουροπλής

---



## ΠΡΟΣΕΔΑΦΙΣΗ ΣΤΟΝ ΤΙΤΑΝΑ

Μια παραλλαγή ενός γνωστού παιχνιδιού (lunar lander) είναι το πρόγραμμα αυτό για τον TI 99/4A με την EXTENDED BASIC. Αποστολή σας είναι να προσεδαφιστείτε με τη σωστή ταχύτητα στην εξέδρα προσγείωσης (η επίπεδη επιφάνεια), χειριζόμενος κατάλληλα τους πυραύλους κατεύθυνσης του διαστημοπλοίου σας. Ο χειρισμός γίνεται είτε από το πληκτρολόγιο, είτε από τα joysticks. Ανάλογα με το επίπεδο δυσκολίας που θα επιλέξετε στην αρχή, η εξέδρα προσγείωσης θα έχει μεγάλο ή μικρό μήκος, απαιτώντας χειρισμούς μεγαλύτερης ακριβείας όσο πιο μικρό είναι το μέγεθός της.

Τα πράγματα δυσκολεύουν λόγω της αδράνειας, η οποία κα-

θυστερεί την ανταπόκριση του σκάφους στους χειρισμούς της. Ακόμα κάθε πυροδότηση πυραύλων καταναλώνει πολύτιμο καύσιμα με κίνδυνο να πέσει το σκάφος στην επιφάνεια μόλις τελειώσουν. Το πρόγραμμα κάνει εκτεταμένη χρήση των SPRITES του TI με αποτέλεσμα την εξαιρετικά ομαλή κίνηση του διαστημόπλοιοι, που κάνει πιο ρεαλιστικό το παιχνίδι. Επιπλέον οδηγίες για το παιχνίδι περιλαμβάνονται μέσα στο ίδιο το πρόγραμμα το οποίο είναι και αρκετά καλό στην λειτουργία του.

Όσοι δεν συμπαθούν την πληκτρολόγηση μπορούν να προμηθευτούν ένα αντίγραφο του προγράμματος σε κασέτα, από την εδώ αντιπροσωπεία του TI 99/4A

```

100 REM *****
110 REM *LIFE LINE TO TITAN*
120 REM *****
130 REM
140 REM
150 ON ERROR 190
160 RAM$="N"
170 CALL INIT :: CALL LOAD(-3187
8,0)
180 RAM$="Y"
190 CALL CLEAR
200 RANDOMIZE
210 FOR SET=1 TO 14
220 CALL COLOR(SET,16,1)
230 NEXT SET
240 CALL SCREEN(9)
250 CALL CHAR(40,"80402010080402
010102040810204080FF"&RPT$( "0",2
8)&"FF")
260 CALL CHAR(96,"0010102828447C
8200081818286898040004081828C830
0800020C34C8281010")
270 CALL CHAR(100,"000002BC48502
0200000807E4C30204000806058465A6
0804020304C7E800000")
280 CALL CHAR(104,"20205048BC020
000101028C8340C02000830C828180B0
40004986B2818180800")
290 CALL CHAR(108,"827C442828101
0002019161418181000100C131418102
000080814132C304000")
300 CALL CHAR(112,"04040A123D400
00002040C327E01000001061A621A060
1000000017E320C0402")
310 CALL CHAR(116,"0000403D120A0
4040040302C131408080020101814130
C100010181814161920")
320 CALL CHAR(120,"0010181814161
94")
330 CALL CHAR(122,"0000040202000
000000000004040400000000000010080
8000000000000020202")
340 CALL CHAR(126,"0000000000206
03000000000080A060600000000000202
02000000000707010101")
350 DIM DIR(28,2),HGT(256),N$(3)
,RPLY(25,3)
360 N$(1)="CADET" :: N$(2)="OFFI
CER" :: N$(3)="VETERAN"
370 RESTORE 390 :: FOR CELL2=1 T
O 2 :: FOR CELL1=1 TO 24
380 READ CELL :: DIR(CELL1,CELL2
)=CELL :: NEXT CELL1 :: NEXT CEL
L2
390 DATA -3,-3,-3,-3,-2,-1,0,1,2
,3,3,3,3,3,3,2,1,0,-1,-2,-3,-3
,-3

```



```

400 DATA 0,1,2,3,3,3,3,3,3,2,1
,0,-1,-2,-3,-3,-3,-3,-3,-3,-2
,-1
410 CALL DELSPRITE(ALL)
420 IF RAM$="N" THEN 440
430 CALL INIT :: CALL LOAD(-3187
8,1)
440 GOSUB 790
450 GOSUB 1030
460 GOSUB 1480
470 REM MAIN CONTROL LOOP
480 CALL KEY(1,A,B):: CALL JOYST
(1,C,D)
490 CALL POSITION(#1,Y,X)
500 IF X+4>256 THEN 640
510 IF Y+8>HGT(X+4)+2 AND Y<192
THEN 1640
520 IF A=2 OR C=-4 THEN PNT=PNT-
1
530 IF A=3 OR C=4 THEN PNT=PNT+1
540 IF PNT=25 THEN PNT=1 ELSE IF
PNT=0 THEN PNT=24
550 CALL PATTERN(#1,95+PNT)
560 IF RDD=1 THEN 600
570 DISPLAY AT(23,7)SIZE(4):INT(
XV)
580 DISPLAY AT(23,21)SIZE(4):INT
(-YV)
590 DISPLAY AT(24,20):FUEL
600 CALL POSITION(#1,Y,X)
610 RPLY(PST,1)=Y :: RPLY(PST,2)
=X :: RPLY(PST,3)=PNT+95 :: IF P
ST=25 THEN PST=1 ELSE PST=PST+1
620 IF X+4>256 THEN 640
630 IF Y+8>HGT(X+4)+2 AND Y<192
THEN 1640
640 YV=YV+1+1.5*((1/(4-DIF))/2)
650 CALL KEY(1,A,B):: IF A=13 OR
A=18 THEN 660 ELSE 730
660 IF FUEL=0 THEN DISPLAY AT(22
,1):"FUEL SUPPLY IS GONE" :: GOT
O 730
670 XV=XV+DIR(PNT,2)
680 YV=YV+DIR(PNT,1)
690 FUEL=FUEL-1
700 IF FUEL=10 THEN DISPLAY AT(2
2,1):"FUEL RUNNING LOW" :: CALL
BEEP
710 IF FUEL=20 THEN DISPLAY AT(2
2,1):"WATCH YOUR FUEL SUPPLY" ::
CALL BEEP
720 CALL SOUND(-800,125,28,-7,15
)
730 IF ABS(XV)>254 OR ABS(YV)>25
4 THEN 760
740 CALL MOTION(#1,YV/(2-DIF*.25
),XV/(2-DIF*.25))
750 GOTO 480
760 XV=XV-DIR(PNT,2)
770 YV=YV-DIR(PNT,1)
780 GOTO 480
790 REM TITLE SCREEN
800 CALL CLEAR
810 DISPLAY AT(1,1):"LIFE LINE T
O TITAN"
820 DISPLAY AT(3,1):"SELECT YOUR
LEVEL"

```

```

830 DISPLAY AT(5,1):" 1 CADET":"
 2 OFFICER":" 3 VETERAN"
840 DISPLAY AT(9,1):"PRESS ""AID
"" FOR RULES"
850 CALL BEEP
860 CALL JOYST(1,A,B)
870 IF A=0 AND B=0 THEN 900
880 DIF=2-B*.25
890 GOTO 930
900 CALL KEY(0,C,D)
910 IF C=1 THEN 950
920 IF C>48 AND C<54 THEN DIF=C-
48 ELSE 860
930 CALL HCHAR(4+DIF,3,102)
940 RETURN
950 REM INSTRUCTIONS
960 DISPLAY AT(3,1):"YOU ARE PIL
OTING A CARGO":"LANDER.YOUR MISS
ION IS TO":"TOUCH DOWN ON THE FL
AT"
970 DISPLAY AT(6,1):"SURFACE USI
NG A MINIMUM OF FUEL."
980 DISPLAY AT(9,1):"YOU MAY USE
S & D TO ROTATE YOUR LANDER AND
V TO THRUST,OR JOYSTICK #1.DON'
T THRUST"
990 DISPLAY AT(12,1):"TOO OFTEN-
YOU'LL RUN OUT OF FUEL.TRY NOT T
O HIT ANY MOUNTAINS!"
1000 DISPLAY AT(17,1):"PRESS ANY
KEY TO GO BACK"
1010 CALL KEY(0,A,B)
1020 IF B<>0 THEN 800 ELSE 1010
1030 CALL HCHAR(2,1,32,736)
1040 DISPLAY AT(1,20):N$(DIF)
1050 DISPLAY AT(2,1):"PLEASE WAI
T."
1060 LNDSP=INT(25*RND+3)
1070 ROW=INT((10-DIF)*RND+10-DIF
)+(4-DIF)*1.5
1080 IF ROW>19 THEN 1070
1090 COL=0 :: DOT=1 :: X=1
1100 COL=COL+1
1110 IF COL=33 THEN RETURN
1120 IF COL=LNDSP THEN 1390
1130 OLS=SLP
1140 IF INT(5*RND+1)<DIF THEN 11
70
1150 SLP=INT(3*RND-1)
1160 IF SLP=0 THEN 1150
1170 IF SLP=1 THEN 1190 ELSE 129
0
1180 GOTO 1100
1190 IF ROW<6 THEN 1290
1200 IF OLS=1 THEN ROW=ROW-1
1210 SLP,OLS=1
1220 CALL HCHAR(ROW,COL,41)
1230 FOR L=1 TO 8
1240 DOT=ROW*8-L+1
1250 HGT(X)=DOT
1260 X=X+1
1270 NEXT L
1280 GOTO 1100
1290 IF ROW>18 THEN 1190
1300 IF OLS=-1 THEN ROW=ROW+1
1310 SLP,OLS=-1
1320 CALL HCHAR(ROW,COL,40)

```

```

1330 FOR L=1 TO 8
1340 DOT=ROW*8-8+L
1350 HGT(X)=DOT
1360 X=X+1
1370 NEXT L
1380 GOTO 1100
1390 IF SLP=-1 THEN CHA=43 ELSE
CHA=42
1400 CALL HCHAR(ROW,COL,CHA,4-DI
F)
1410 FOR L=1 TO 8*(4-DIF)
1420 IF CHA=42 THEN DOT=ROW*8-7
ELSE DOT=ROW*8
1430 HGT(X)=DOT
1440 X=X+1
1450 NEXT L
1460 COL=COL+3-DIF
1470 GOTO 1100
1480 CALL HCHAR(21,1,32,128)
1490 DISPLAY AT(2,1):"PRESS ANY
KEY TO START": "      N FOR NO RE
ADOUTS": " "
1500 CALL BEEP
1510 CALL KEY(0,A,B):: CALL JOYS
T(1,X,Y)
1520 IF B<>0 THEN 1530 ELSE IF X
=0 AND Y=0 THEN 1510
1530 IF A=78 THEN RDO=1 ELSE RDO
=0
1540 CALL HCHAR(2,1,32,64)
1550 IF RDO=0 THEN DISPLAY AT(23
,1):"X VEL      Y VEL
      FUEL"
1560 X=INT(180*RND+40)
1570 IF X>LNDSP*8-16 AND X<(LND
SPT+4-DIF)*8+16 THEN 1560
1580 CALL SPRITE(#1,96,16,8,X)
1590 PNT=1 :: FUEL=100-25*DIF ::
XV,YV=0 :: PST=1
1600 FOR L=1 TO 25
1610 RPLY(L,1),RPLY(L,2),RPLY(L,
3)=0
1620 NEXT L
1630 RETURN
1640 CALL MOTION(#1,0,0)
1650 IF X+4>LNDSP*8-11 AND X+6<
(LNDSP+4-DIF)*8 THEN 1840 ELSE
1670
1660 DISPLAY AT(21,1):"YOU'RE LA
NDING TOO FAST!"
1670 CALL PATTERN(#1,46)
1680 IF RAM$="N" THEN 1700
1690 CALL LOAD(-31878,5)
1700 CALL EXPLODE(YV,XV,Y,X)
1710 DISPLAY AT(2,1):"PRESS R FO
R A REPLAY      S FOR TH
E SAME GAME  ANY OTHER KEY FOR
A NEW GAME"
1720 CALL STOP
1730 CALL JOYST(1,A,B)
1740 IF A=0 AND B=0 THEN 1760
1750 IF B*.25=0 THEN 1800 ELSE I
F B*.25=1 THEN 1970 ELSE 410
1760 CALL KEY(0,C,D)
1770 IF D=0 THEN 1730
1780 CALL HCHAR(2,1,32,96)
1790 IF C=83 THEN 1800 ELSE IF C

```

```

=82 THEN 1970 ELSE 410
1800 CALL DELSPRITE(ALL)
1810 IF RAM$="N" THEN 1830
1820 CALL INIT :: CALL LOAD(-318
78,1)
1830 GOTO 460
1840 IF ABS(YV)>7-DIF OR ABS(XV)
>7-DIF THEN 1660
1850 IF Y>192 THEN 1670
1860 DISPLAY AT(21,1):"LANDER TO
BASE-      WE'RE SU
CCESFULLY DOWN!"
1870 CALL BEEP
1880 FOR L=1 TO 200 :: NEXT L
1890 IF RAM$="N" THEN 1910
1900 CALL INIT :: CALL LOAD(-318
78,3)
1910 FOR L=122 TO 127
1920 CALL SPRITE(#2,L,16,Y,X-5)
1930 FOR D=1 TO 75 :: NEXT D
1940 NEXT L
1950 CALL SPRITE(#2,128,16,Y,X-5
,#3,129,16,HGT(X-5)-5,X-13)
1960 GOTO 1710
1970 CALL HCHAR(2,1,32,96)
1980 DISPLAY AT(3,1):"REPLAY"
1990 CALL DELSPRITE(#2,#3,#4,#5,
#6)
2000 T=1 :: IF PST=25 THEN PST=1
ELSE PST=PST+1
2010 IF RPLY(PST,3)=0 THEN 2000
2020 CALL SPRITE(#1,RPLY(PST,3),
16,RPLY(PST,1),RPLY(PST,2))
2030 IF PST=25 THEN PST=1 ELSE P
ST=PST+1 :: IF T=23 THEN 1710 EL
SE T=T+1 :: IF RPLY(PST,3)=0 THE
N 1710
2040 FOR L=1 TO 150 :: NEXT L ::
GOTO 2020
2050 SUB BEEP
2060 VOL=28
2070 FOR N=3000 TO 2985 STEP -2,
5
2080 CALL SOUND(-200,N,VOL)
2090 VOL=VOL-4
2100 NEXT N
2110 SUBEND
2120 SUB EXPLODE(YV,XV,Y,X)
2130 CALL SCREEN(16):: CALL SCRE
EN(9)
2140 FOR L=2 TO 5
2150 CALL SPRITE(#L,46,16,Y,X,(I
NT(5*RND-4)*(ABS(YV)/5))-INT(2*R
ND),INT(7*RND-3)*(ABS(YV)/5))
2160 NEXT L
2170 P=1 :: N=160 :: V=0 :: T=6
2180 CALL SOUND(-250,N,V,110,V,-
T,V/2)
2190 N=N-4 :: V=V+1 :: P=P+1 ::
IF INT(P/8)=P/11 THEN T=T+1
2200 IF N>110 THEN 2180
2210 SUBEND
2220 SUB STOP
2230 FOR L=2 TO 5
2240 CALL MOTION(#L,0,0)
2250 NEXT L
2260 SUBEND

```

Το TEX THELLO είναι μια παραλλαγή για το TI 99/4A του αρκετά γνωστού παιχνιδιού "OTHELLO". Το OTHELLO λοιπόν είναι ένα παιχνίδι στρατηγικής με μεγάλο ενδιαφέρον. Το παιχνίδι παίζεται όπως το σκάκι σε μια σκακιέρα 8x8 μεταξύ δυο αντιπάλων. Το πιο εντυπωσιακό χαρακτηριστικό του OTHELLO για τον αρχάριο είναι η τρομακτική ευκολία με την οποία μπορεί να αλλάξει η κατάσταση του παιχνιδιού.

Λόγω της φύσης του παιχνιδιού η νίκη μπορεί να αμφισβητείται σχεδόν πάντα παρά την ενδεχόμενη υπεροχή ενός παίκτη. Έτσι μεγάλες περιοχές της σκακιέρας μπορούν να αλλάξουν στρατόπεδα με μια κίνηση και απαιτείται επιδέξιο παιχνίδι για την τελική νίκη. Υπάρχουν πολυάριθμες δυνατές κινήσεις και αντίστοιχες εξελίξεις. Αντίθετα από το σκάκι, το OTHELLO τε-

λειώνει υποχρεωτικά μέσα σε 60 κινήσεις, ενώ συνήθως ένα τυπικό παιχνίδι δεν υπερβαίνει τις 35 κινήσεις.

#### ΚΑΝΟΝΙΣΜΟΙ ΤΟΥ TEX THELLO

1. Οι αντίπαλοι παίζουν διαδοχικά πάνω στην σκακιέρα με τα 64 τετράγωνα.

2. Η κίνηση γίνεται με τοποθέτηση ενός νέου πεσσού πάνω στην σκακιέρα σ'ένα κενό τετράγωνο, δίπλα σε πεσό του αντιπάλου.

3. Κάθε κίνηση πρέπει να προκαλεί την κατάκτηση τουλάχιστον ενός πεσσού του αντίπαλου.

4. Οι κατακτήσεις γίνονται οριζόντια, κάθετα και διαγώνια και περιλαμβάνουν όλους τους πεσσούς που βρίσκονται ενδιάμεσα και σε ευθεία με τον πεσό που τοποθετείται και οποιονδήποτε του ίδιου χρώματος υπάρχει ήδη.

Αν έχετε απορίες ακόμα πάνω στους κανόνες, θα σας λυθούν εύκολα με λίγη εξάσκηση μια και ο κομπιούτερ ελέγχει την ορθότητα όλων των κινήσεων. Μερικές υποδείξεις που μπορεί να σας βοηθήσουν στο παιχνίδι είναι:

Τα τέσσερα τετράγωνα στις γωνίες, έχουν μεγάλη αξία καθώς δεν καταλαμβάνονται ενώ ελέγχουν κατά τρεις διευθύνσεις πολλά τετράγωνα.

Αν οι παραπάνω θέσεις είναι δικές σας, αρκετή αξία παρουσιάζουν θέσεις πάνω στις κεντρικές διαγωνίους.

Τα τετράγωνα στις πλευρές της σκακιέρας μπορούν να καταληφθούν εύκολα από τον αντίπαλο αν καταλάβει τα ακρογωνιαία τετράγωνα.

Στην αρχή του παιχνιδιού αποφεύγετε το να εξαπλώνεστε γρήγορα για να περιορίζετε τις πιθανές κινήσεις του αντίπαλου.

Το πρόγραμμα είναι αρκετά πολύπλοκο και ο υπολογιστής καταφέρνει να παίξει ένα αρκετά δυνατό παιχνίδι. Λόγω του μεγέθους χρειάζεται μεγάλη προσοχή για τυχόν λάθη κατά την πληκτρολόγηση.

Στην αρχή το πρόγραμμα σας ρωτά αν θέλετε να παίξετε με αντίπαλο τον κομπιούτερ ή απλώς να χρησιμοποιήσετε την σκακιέρα μ'έναν άλλο συμπαίκτη. Στη δεύτερη περίπτωση ο κομπιούτερ κάνει τις κινήσεις που του ζητάτε αφού ελέγξει πρώτα την νομιμότητά τους. Στην συνέχεια και εφόσον διαλέξατε τον κομπιούτερ για αντίπαλό σας ζητείται το επίπεδο δυσκολίας, το ποιος θα παίξει πρώτος και με τι χρώμα, και τέλος το όνομα του παίκτη. Από το σημείο αυτό και μετά θα πρέπει να δείξετε αν η δεξιοτεχνία σας είναι καλύτερη από αυτήν ενός κουτιού με εξαρτήματα ή έστω του προγραμματιστού.

## TEX THELLO

```

160 DIM EX(18),EY(18)
170 DATA 67,79,77,80,85,84,69,82
180 RESTORE
190 COMPLAY=0
200 CALL CLEAR
210 FOR I=1 TO 8
220 READ PLAY(1,I)
230 PLAY(2,I)=PLAY(1,I)
240 NEXT I
250 PRINT TAB(9):"TEX-THELLO":
::
260 PRINT "CHOOSE:"::"1 YOU AGA
INST COMPUTER"::"2 TWO PLAYERS":
:::
270 CALL KEY(0,K,S)
280 IF (K<49)+(K>50)=-1 THEN 270
290 IF K=50 THEN 410

```

```

300 CALL CLEAR
310 PRINT "CHOOSE:"::"1 EASY G
AME"::"2 INTERMEDIATE GAME"::"3
HARD GAME":
:::
320 CALL KEY(0,K,S)
330 IF (K<49)+(K>51)=-1 THEN 320
340 HARD=K-48
350 CALL CLEAR
360 PRINT "CHOOSE -- COMPUTER PL
AYS"::"1 FIRST--RED"::"2 SECO
ND--YELLOW":
:::
370 CALL KEY(0,K,S)
380 IF (K<49)+(K>50)=-1 THEN 370
390 COMPLAY=K-48
400 IF COMPLAY=1 THEN 470
410 PRINT ::"FIRST PLAYER NAME (<
RED):"

```

```

420 INPUT Z1$
430 FOR I=1 TO 10
440 CALL GCHAR(23,I+4,PLAY(1,I))
450 NEXT I
460 IF COMPLAY=2 THEN 520
470 PRINT "SECOND PLAYER NAME
(YELLOW)"
480 INPUT Z1$
490 FOR I=1 TO 10
500 CALL GCHAR(23,I+4,PLAY(2,I))
510 NEXT I
520 FOR I=1 TO 3
530 DIR(I)=I-2
540 NEXT I
550 FOR I=0 TO 9
560 FOR J=0 TO 9
570 A(I,J)=1
580 IF I*J*(J-9)*(I-9)<>0 THEN 6
00
590 A(I,J)=A(I,J)+1
600 NEXT J
610 NEXT I
620 CALL CLEAR
630 PRINT TAB(19);"X=  Y="
640 E$="FFFFFFFFFFFFFFF"
650 CALL CHAR(120,E$)
660 CALL COLOR(12,2,16)
670 FOR I=1 TO 8
680 Y=3+2*I
690 X=Y+4
700 CD=48+I
710 CALL VCHAR(Y,8,CD)
720 CALL VCHAR(4,X,CD)
730 NEXT I
740 A$="FF80808080808080"
750 B$="FF01010101010101"
760 C$="80808080808080FF"
770 D$="01010101010101FF"
780 CALL CHAR(96,A$)
790 CALL CHAR(97,B$)
800 CALL CHAR(98,C$)
810 CALL CHAR(99,D$)
820 CALL CHAR(104,A$)
830 CALL CHAR(105,B$)
840 CALL CHAR(106,C$)
850 CALL CHAR(107,D$)
860 CALL CHAR(112,A$)
870 CALL CHAR(113,B$)
880 CALL CHAR(114,C$)
890 CALL CHAR(115,D$)
900 CALL COLOR(9,2,16)
910 CALL COLOR(10,2,9)
920 CALL COLOR(11,2,11)
930 TYPE=1
940 FOR X=1 TO 8
950 FOR Y=1 TO 8
960 GOSUB 2270
970 NEXT Y
980 NEXT X
990 FOR X=4 TO 5
1000 FOR Y=4 TO 5
1010 IF X=Y THEN 1050
1020 TYPE=2
1030 A(X,Y)=3
1040 GOTO 1070
1050 TYPE=3
1060 A(X,Y)=4

```

```

1070 GOSUB 2330
1080 NEXT Y
1090 NEXT X
1100 FOR I=3 TO 6
1110 FOR J=3 TO 6
1120 IF A(I,J)<>1 THEN 1140
1130 A(I,J)=0
1140 NEXT J
1150 NEXT I
1160 PL=1
1170 TURN=5
1180 REM BEGIN LOOP
1190 FOR I=1 TO 10
1200 CALL HCHAR(23,I+7,PLAY(PL,I
))
1210 NEXT I
1220 CALL VCHAR(23,23,120)
1230 CALL VCHAR(23,28,120)
1240 IF COMPLAY=PL THEN 1340
1250 CALL KEY(0,RE,ST)
1260 IF ST<>1 THEN 1250
1270 X=RE-48
1280 CALL VCHAR(23,23,RE)
1290 CALL KEY(0,RE,ST)
1300 IF ST<>1 THEN 1290
1310 Y=RE-48
1320 CALL VCHAR(23,28,RE)
1330 GOTO 1370
1340 GOSUB 2960
1350 CALL VCHAR(23,23,48+X)
1360 CALL VCHAR(23,28,48+Y)
1370 IF X>8 THEN 1460
1380 IF X<1 THEN 1460
1390 IF Y>8 THEN 1460
1400 IF Y<1 THEN 1460
1410 IF A(X,Y)>1 THEN 1460
1420 GOSUB 2530
1430 IF EXTRA>0 THEN 1490
1440 GOSUB 2120
1450 IF SW=0 THEN 1640
1460 CALL SOUND(500,200,3)
1470 IF COMPLAY=PL THEN 1640
1480 GOTO 1220
1490 A(X,Y)=PL+2
1500 FOR I=X-1 TO X+1
1510 FOR J=Y-1 TO Y+1
1520 IF A(I,J)<>1 THEN 1540
1530 A(I,J)=0
1540 NEXT J
1550 NEXT I
1560 TURN=TURN+1
1570 TYPE=2
1580 IF PL=1 THEN 1600
1590 TYPE=3
1600 GOSUB 2270
1610 GOSUB 2530
1620 GOSUB 2840
1630 IF TURN=65 THEN 1810
1640 IF PL=1 THEN 1670
1650 A1=4
1660 GOTO 1680
1670 A1=3
1680 FOR I=1 TO 8
1690 FOR J=1 TO 8
1700 IF A(I,J)<=1 THEN 1720
1710 IF A(I,J)<>A1 THEN 1750
1720 NEXT J

```

```

1730 NEXT I
1740 GOTO 1810
1750 IF PL=1 THEN 1780
1760 PL=1
1770 GOTO 1190
1780 PL=2
1790 GOTO 1190
1800 REM END MAIN LOOP
1805 REM BEGIN TOTALS REGION

1810 TOT1=0
1820 TOT2=0
1830 FOR I=1 TO 8
1840 FOR J=1 TO 8
1850 IF A(I,J)=3 THEN 1890
1860 IF A(I,J) <> 4 THEN 1900
1870 TOT2=TOT2+1
1880 GOTO 1900
1890 TOT1=TOT1+1
1900 NEXT J
1910 NEXT I
1920 IF TOT2 <> TOT1 THEN 1950
1930 PRINT "DRAW 32 TO 32"
1940 GOTO 2050
1950 PRINT
1960 PRINT "AT A SCORE OF";TOT1;
" TO ";TOT2
1970 PRINT " WINS"
1980 IF TOT1>TOT2 THEN 2010
1990 PWIN=2
2000 GOTO 2020
2010 PWIN=1
2020 FOR I=1 TO 10
2030 CALL HCHAR(23,I+1,PLAY(PWIN
,I))
2040 NEXT I
2050 PRINT "PLAY AGAIN (Y OR N)"
;
2060 CALL KEY(O,K,S)
2070 IF K=89 THEN 180
2080 IF K <> 78 THEN 2060
2090 STOP
2100 REM
2110 REM IS THERE ALEGAL MOVE?

2120 SW=0
2130 ZZ=X
2140 ZY=Y
2150 FOR X=1 TO 8
2160 FOR Y=1 TO 8
2170 IF A(X,Y) <> 0 THEN 2210
2180 GOSUB 2530
2190 IF EXTRA=0 THEN 2210
2200 SW=1
2210 NEXT Y
2220 NEXT X
2230 X=ZZX
2240 Y=ZZY
2250 RETURN
2260 REM COLOR ONTO BOARD POUTI
NE
2270 IF TYPE <> 1 THEN 2330
2280 S1=96
2290 S2=97
2300 S3=98
2310 S4=99
2320 GOTO 2430

```

```

2330 IF TYPE <> 2 THEN 2390
2340 S1=104
2350 S2=105
2360 S3=106
2370 S4=107
2380 GOTO 2430
2390 S1=112
2400 S2=113
2410 S3=114
2420 S4=115
2430 X1=7+2*X
2440 X2=X1+1
2450 Y1=3+2*Y
2460 Y2=Y1+1
2470 CALL HCHAR(Y1,X1,S1)
2480 CALL HCHAR(Y1,X2,S2)
2490 CALL HCHAR(Y2,X1,S3)
2500 CALL HCHAR(Y2,X2,S4)
2510 RETURN
2520 REM EXTRA SQUARES
2530 EXTRA=1
2540 FOR I=1 TO 3
2550 FOR J=1 TO 3
2560 U=X+DIR(I)
2570 V=Y+DIR(J)
2580 IF U=X THEN 2590 ELSE 2600
2590 IF V=Y THEN 2790
2600 IF PL=1 THEN 2620
2610 IF A(U,V)=3 THEN 2630 ELSE
2790
2620 IF A(U,V)=4 THEN 2630 ELSE
2790
2630 U=U+DIR(I)
2640 V=V+DIR(J)
2650 IF A(U,V) <= 1 THEN 2790
2660 IF A(U,V)=2 THEN 2790
2670 IF PL=1 THEN 2690
2680 IF A(U,V)=4 THEN 2700 ELSE
2630
2690 IF A(U,V)=3 THEN 2700 ELSE
2630
2700 U=X+DIR(I)
2710 V=Y+DIR(J)
2720 IF A(U,V)=PL+2 THEN 2790
2730 EX(EXTRA)=U
2740 EY(EXTRA)=V
2750 EXTRA=EXTRA+1
2760 U=U+DIR(I)
2770 V=V+DIR(J)
2780 GOTO 2720
2790 NEXT J
2800 NEXT I
2810 EXTRA=EXTRA-1
2820 RETURN
2830 REM COLOR ADDITION SQUARES

2840 XX=X
2850 YY=Y
2860 FOR K=1 TO EXTRA
2870 X=EX(K)
2880 Y=EY(K)
2890 GOSUB 2270
2900 A(X,Y)=PL+2
2910 NEXT K
2920 X=XX
2930 Y=YY
2940 RETURN

```

```

2950 REM GET COMPUTER MOVE
2960 EXTRAP=-100
2970 FOR X=1 TO 8
2980 FOR Y=1 TO 8
2990 IF A(X,Y) <> 0 THEN 4110
3000 GOSUB 2530
3010 IF EXTRA=0 THEN 4110
3020 IF TURN<57 THEN 3040
3030 HARD=1
3040 IF HARD=1 THEN 4070
3050 IF X=1 THEN 3380
3060 IF X=8 THEN 3380
3070 IF Y=1 THEN 3260
3080 IF Y=8 THEN 3260
3090 IF HARD=2 THEN 4070
3100 IF X=2 THEN 3180
3110 IF X=7 THEN 3180
3120 IF Y=2 THEN 4070
3130 IF Y=7 THEN 4070
3140 IF X*X+Y*Y+36 <> 9*(X+Y) THEN
3160
3150 EXTRA=EXTRA+1
3160 EXTRA=EXTRA+4
3170 GOTO 4070
3180 IF Y=2 THEN 3210
3190 IF Y=7 THEN 3210
3200 GOTO 4070
3210 U=(7*X-9)/5
3220 V=(7*Y-9)/5
3230 IF A(U,V) > 1 THEN 4070
3240 EXTRA=EXTRA-12
3250 GOTO 4070
3260 IF HARD=2 THEN 3360
3270 FOR I=1 TO 8
3280 EDGE(I)=A(I,Y)
3290 NEXT I
3300 FOR I=1 TO EXTRA
3310 IF EY(I) <> Y THEN 3330
3320 EDGE(EX(I))=PL+2
3330 NEXT I
3340 Z=X
3350 GOTO 3570
3360 EXTRA=EXTRA+3
3370 GOTO 4070
3380 IF Y=1 THEN 3520
3390 IF Y=8 THEN 3520
3400 IF HARD=2 THEN 3500
3410 FOR I=1 TO 8
3420 EDGE(I)=A(X,I)
3430 NEXT I
3440 FOR I=1 TO EXTRA
3450 IF EX(I) <> X THEN 3470
3460 EDGE(EY(I))=PL+2
3470 NEXT I
3480 Z=Y
3490 GOTO 3570
3500 EXTRA=EXTRA+3
3510 GOTO 4070
3520 IF HARD=2 THEN 3550
3530 EXTRA=EXTRA+14
3540 GOTO 4070
3550 EXTRA=EXTRA+6
3560 GOTO 4070
3570 CORN=0
3580 FOR I=1 TO EXTRA
3590 IF 2*(EX(I)-2)*(EX(I)-7)+(E
Y(I)-2)*(EY(I)-7)=0 THEN 3610

```

```

3600 GOTO 3660
3610 U=(7*EX(I)-9)/5
3620 V=(7*EY(I)-9)/5
3630 IF A(U,V) <= 1 THEN 3650
3640 GOTO 3660
3650 CORN=1
3660 NEXT I
3670 TYPE=0
3680 IF EDGE(1)=PL+2 THEN 3730
3690 LT=EDGE(1)
3700 IF LT <> 0 THEN 3740
3710 LT=1
3720 GOTO 3740
3730 LT=2
3740 FOR I=2 TO Z-1
3750 IF EDGE(I)=PL+2 THEN 3800
3760 IF EDGE(I) <= 1 THEN 3790
3770 LT=EDGE(I)
3780 GOTO 3800
3790 LT=1
3800 NEXT I
3810 IF LT=2 THEN 3970
3820 IF EDGE(8)=PL+2 THEN 3870
3830 HT=EDGE(8)
3840 IF HT <> 0 THEN 3880
3850 HT=1
3860 GOTO 3880
3870 HT=2
3880 FOR I=7 TO Z+1 STEP -1
3890 IF EDGE(I)=PL+2 THEN 3940
3900 IF EDGE(I) <= 1 THEN 3930
3910 HT=EDGE(I)
3920 GOTO 3940
3930 HT=1
3940 NEXT I
3950 IF HT=2 THEN 3970
3960 IF HT <> LT THEN 4030
3970 TYPE=1
3980 IF CORN=0 THEN 4010
3990 EXTRA=EXTRA-8
4000 GOTO 4070
4010 EXTRA=EXTRA+8
4020 GOTO 4070
4030 IF CORN=0 THEN 4060
4040 EXTRA=EXTRA-12
4050 GOTO 4070
4060 EXTRA=EXTRA-4
4070 IF EXTRA > EXTRAP THEN 4080 E
LSE 4110
4080 EXTRAP=EXTRA
4090 ZX=X
4100 ZY=Y
4110 NEXT Y
4120 NEXT X
4130 IF EXTRAP=-100 THEN 4140 EL
SE 4210
4140 FOR X=1 TO 8
4150 FOR Y=1 TO 8
4160 IF A(X,Y) > 1 THEN 4190
4170 ZX=X
4180 ZY=Y
4190 NEXT Y
4200 NEXT X
4210 X=ZX
4220 Y=ZY
4230 RETURN
4240 END

```

# Η ΓΗ ΣΕ ΚΙΝΔΥΝΟ



Ο πλανήτης μας αντιμετωπίζει κίνδυνο αφανισμού από τους εξωγήινους εισβολείς που προσπαθούν να διασπάσουν τις αμυντικές δυνάμεις μας. Αποστολή σας να αποτρέψετε την εισβολή, καταστρέφοντας τους εξωγήινους με τα τρία πυροβόλα που διαθέτετε. Κάθε φορά που ένας εξωγήινος διασπάσει την άμυνα, χάνετε και ένα πυροβόλο και όταν όλα εξαντληθούν τίποτα δεν μπορεί να σταματήσει τους εισβολείς... Χρησιμοποιώντας τα πλήκτρα "A" και "Z" μετακινείτε το πυροβόλο πάνω και κάτω, αντίστοιχα, και με το πλήκτρο "." πυροβολείτε. Προσοχή όμως γιατί οι εξωγήινοι είναι κι αυτοί οπλισμένοι και κινδυνεύετε να πέσετε πρώτοι εσείς στο πεδίο της μάχης... Κάθε εξωγήινος που καταστρέφεται αυξάνει το σκορ σας κατά 100 πόντους. Για να φτάσετε τους 10.000 πόντους που απαιτούνται για να νικήθούν οι επιδρομείς θα χρειαστεί πολλή δεξιοτεχνία και γρήγορα

αντανακλαστικά.

Σαν πρόγραμμα, το παιχνίδι είναι αρκετά απλό και σύντομο. Οι γραμμές 870-950 δείχνουν έναν έξυπνο τρόπο για να απεικονίζεται το σκορ πάνω στην οθόνη και να ανανεώνεται συνεχώς. Γι' αυτούς που δεν συμπαθούν πολύ τους εξωγήινους, η εξολόθρευση μπορεί να διευκολυνθεί με δυο τρόπους: Ο ένας είναι να αυξήσετε τον αριθμό των πυροβόλων σας που βρίσκεται στην γραμμή 100 (SPA). Ο δεύτερος είναι να μειώσετε την συχνότητα με την οποία πυροβολεί κάθε εξωγήινος επεμβαίνοντας στην γραμμή 560. Αυξάνοντας το 0,9 μειώνετε την συχνότητα ενώ ελαττώνοντας το την αυξάνετε. Η τιμή που θα δώσετε αντί του 0,9 δεν πρέπει να είναι μικρότερη του 0 ή μεγαλύτερη της μονάδας.

Μη βιαστείτε πάντως να κάνετε πιο εύκολο το παιχνίδι γιατί σκεφτείτε ότι σε μια πραγματική εισβολή δεν θάχετε τέτοια δυνατότητα!

```

100 SPA=3
110 RANDOMIZE
120 CALL CLEAR
130 CALL SCREEN(5)
140 CALL CHAR(33,"818142306AFF7E
24")
150 CALL CHAR(40,"32418183C56FF5
A")
160 CALL CHAR(34,"83442180022124
49")
170 CALL CHAR(64,"000000FFFF")
180 CALL CHAR(72,"FF8080808080
80")
190 CALL CHAR(73,"808080808080
FF")
200 REM
210 CALL HCHAR(1,25,48,4)
220 CALL COLOR(3,16,1)
    
```

```

230 CALL COLOR(4,16,1)
240 REM
250 FOR F=2 TO 20 STEP 3
260 CALL HCHAR(F,1,72,20)
270 CALL HCHAR(F+2,1,73,20)
280 CALL SOUND(100,200,1)
290 NEXT F
300 CALL COLOR(6,4,1)
310 CALL SOUND(100,-3,1)
320 REM
330 CALL CHAR(74,"80808080808080")
340 CALL VCHAR(2,30,74,21)
350 CALL SOUND(100,500,1)
360 CALL HCHAR(24,25,32,5)
370 CALL HCHAR(24,25,40,SPA)
380 X=INT(6*RND)*3+3
390 Y=1
400 H=12
410 CALL HCHAR(H,22,40)
420 CALL HCHAR(X,Y,33)
430 CALL COLOR(1,12,1)
440 CALL COLOR(2,12,1)
450 CALL KEY(D,K,S)
460 IF S=0 THEN 500
470 IF K=65 THEN 580
480 IF K=90 THEN 630
490 IF K=46 THEN 680
500 CALL GCHAR(X,Y+1,C)
510 IF C=40 THEN 960
520 CALL HCHAR(X,Y+1,33)
530 CALL HCHAR(X,Y,32)
540 Y=Y+1
550 IF Y=30 THEN 960
560 IF RND>.9 THEN 1180
570 GOTO 450
580 IF H=3 THEN 500
590 CALL HCHAR(H-3,22,40)
600 CALL HCHAR(H,22,32)
610 H=H-3
620 GOTO 500
630 IF H=21 THEN 500
640 CALL HCHAR(H+3,22,40)
650 CALL HCHAR(H,22,32)
660 H=H+3
670 GOTO 500
680 IF H<>X THEN 810
690 CALL HCHAR(H,Y+1,64,21-Y)
700 CALL SOUND(10,900,1)
710 CALL COLOR(5,16,1)
720 CALL COLOR(5,1,1)
730 CALL HCHAR(X,Y,34)
740 CALL HCHAR(X,Y,32)
750 CALL SOUND(-500,-5,1)
760 GOSUB 870
770 CALL HCHAR(X,1,32,21)
780 X=INT(6*RND)*3+3
790 Y=1
800 GOTO 410
810 CALL HCHAR(H,1,64,21)
820 CALL SOUND(10,900,1)

```

```

830 CALL COLOR(5,16,1)
840 CALL COLOR(5,1,1)
850 CALL HCHAR(H,1,32,21)
860 GOTO 500
870 CALL GCHAR(1,26,C)
880 IF C=57 THEN 910
890 CALL HCHAR(1,26,C+1)
900 RETURN
910 CALL HCHAR(1,26,48)
920 CALL GCHAR(1,25,C)
930 IF C=57 THEN 1320
940 CALL HCHAR(1,25,C+1)
950 RETURN
960 SPA=SPA-1
970 CALL HCHAR(X,Y,32)
980 CALL HCHAR(H,22,32)
990 IF SPA>0 THEN 360
1000 CALL SOUND(10,-2,1)
1010 CALL SOUND(500,-4,1)
1020 CALL HCHAR(H,22,34)
1030 CALL VCHAR(5,5,33,5)
1040 CALL HCHAR(5,5,33,4)
1050 CALL HCHAR(9,5,33,4)
1060 CALL HCHAR(9,5,33,4)
1070 CALL VCHAR(5,10,33,5)
1080 CALL VCHAR(5,13,33,5)
1090 CALL HCHAR(5,10,33,3)
1100 CALL VCHAR(5,15,33,5)
1110 CALL VCHAR(6,18,33,3)
1120 CALL HCHAR(5,15,33,3)
1130 CALL HCHAR(9,15,33,3)
1140 FOR DEL=0 TO 400
1150 NEXT DEL
1160 CALL CLEAR
1170 GOTO 1350
1180 IF H<>X THEN 1260
1190 CALL HCHAR(X,Y+1,64,21-Y)
1200 CALL SOUND(10,-3,1)
1210 CALL COLOR(5,9,1)
1220 CALL COLOR(5,1,1)
1230 CALL HCHAR(X,Y+1,32,21-Y)
1240 CALL SOUND(1000,-7,1)
1250 GOTO 960
1260 CALL HCHAR(X,Y+1,64,30-Y)
1270 CALL SOUND(10,-3,1)
1280 CALL COLOR(5,9,1)
1290 CALL COLOR(5,1,1)
1300 CALL HCHAR(X,Y+1,32,30-Y)
1310 GOTO 450
1320 CALL CLEAR
1330 CALL SCREEN(7)
1340 PRINT "well done attack wav
e has been defeated,you have sav
ed earth."
1350 PRINT "press y for new game
or n to stop"
1360 CALL KEY(D,K,S)
1370 IF S=0 THEN 1360
1380 IF K=89 THEN 100
1390 CALL CLEAR
1400 END

```

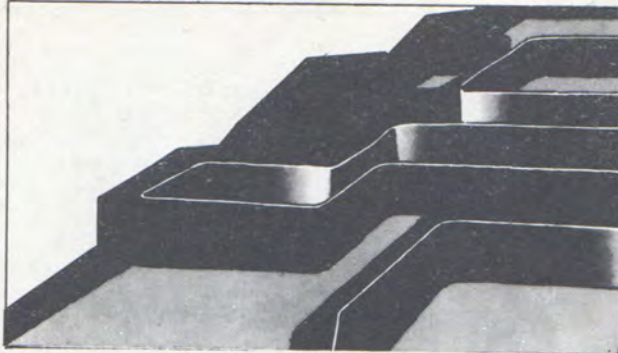


Προγράμματα για τόν:

**TRS-80**



ΕΠΙΜΕΛΕΙΑ:  
'Ανδρέας Μπιρμπίλης



# ΤΡΕΛΛΟΣ ΛΑΒΥΡΙΝΘΟΣ

Αν και περίεργος ο τίτλος, είναι απόλυτα δικαιολογημένος. Ο λαβύρινθος που φτιάχνει αυτό το πρόγραμμα δεν είναι συνηθισμένος, αλλά μεταβάλλεται συνεχώς κατά τη διάρκεια του παιχνιδιού με ένα ρυθμό μια φορά κάθε δυο με πέντε δευτερόλεπτα. Έτσι, εκεί

που υπάρχει κάποια έξοδος, την επόμενη στιγμή μπορεί να υπάρχει ένα αδιέξοδο, πράγμα που δυσκολεύει πολύ το παιχνίδι. Σκοπός σας είναι να βγείτε από το λαβύρινθο στο συντομότερο δυνατό χρόνο. Αρχίζετε από την επάνω αριστερή γωνία της οθόνης και η

έξοδος βρίσκεται πάντα στην κάτω δεξιά γωνία. Για να κινηθείτε, χρησιμοποιείτε τα τέσσερα βελάκια του πληκτρολογίου. Στην αρχή του παιχνιδιού τοποθετούνται στη τύχη διακόσια περίπου εμπόδια μέσα στο λαβύρινθο, τα οποία δυσκολεύουν ακόμα περισσότερο το παιχνίδι. Ο παίχτης έχει ακόμα μια δυνατότητα. Πατώντας το πλήκτρο του διαστήματος (spacebar), όταν βρεθεί σε αδιέξοδο, μπορεί να ανοίξει δρόμο. Αυτή του η ενέργεια όμως επιβαρύνει το συνολικό χρόνο που θα

κάνει για να βγει από τον λαβύρινθο. Την πρώτη φορά του "κοστίζει" δυο δευτερόλεπτα, τη δεύτερη τέσσερα, τη τρίτη οκτώ, τη τέταρτη δέκα δευτερόλεπτα κλπ.

Στο παιχνίδι αυτό μπορούν να παίξουν διαδοχικά πολλοί παίχτες. Ο κομπιούτερ συγκρατεί τους δέκα καλύτερους χρόνους και τους εμφανίζει στο τέλος κάθε παιχνιδιού. Για να "τρέξει" το πρόγραμμα χρειάζεται Extended Basic και 16K μνήμης, ενώ τα joysticks δεν είναι απαραίτητα.

```

5  * *** CRAZY MAZES ***
10 CLEAR500
20 X=RND(TIMER):Y=0
30 GOSUB570
40 GOSUB1050
50 B=CHR$(94)+CHR$(9)+CHR$(10)+CHR$(8)
60 FOR X=1 TO 10:Y=0(X)="LOW SCORE":S(X)=100:NEXT
70 CX=1:CY=1
80 D=2
90 CLS
100 TIMER=0
110 FOR X=0 TO 99:SET(X,0,5):SET(X,30,5):NEXT:FOR X=0 TO 30:SET(0,X,5):SET(60,X,5):NEXT
120 FOR Y=0 TO 29 STEP3
130 FOR X=0 TO 59 STEP 6
140 ON RND(2) GOTO 150,160
150 FOR A=X TO X+5:SET(A,Y+2,5):NEXT:GOTO170
160 FOR A=Y TO Y+2:SET(X+5,A,5):SET(X+4,A,5):NEXT
170 NEXT:NEXT
180 FOR X=1 TO125:SET(RND(80),RND(25)+4,5):NEXT
190 RESET(60,20):RESET(60,27):RESET(59,27):RESET(59,20):RESET(59,27):RESET(59,20)
200 SET(CX,CY,5)
210 PL=2
220 SOUND 220,5
230 SET(63,29,8):SET(63,29,8):TIMER=0
240 AA=INKEY$:IF AA="" THEN 270
250 IF AA=" " THEN1010
260 FOR U=1 TO 4:IF AA=MID$(80,U,1) THEN D=U ELSE NEXT
270 SET(CX,CY,5)
280 POKE1502,128:POKE1501,128
290 IF CX=61 THEN PLAY"7L3AP0L180P8L4FL3EP8L4C":PRINT"CONGRATULATIONS"
300 S="PRINT"YOU MADE IT THROUGH THE MAZE!!":PRINT"TIME "":SC=INT(TIMER+0.1)*100/100
310 PRINTSC" SECONDS":FOR UP=1 TO 1000:NEXT:QT=0:GOTO470
320 ON D GOTO220,340,360,380
330 CY=CY-1:GOTO410
340 IF POINT(CX,CY-1) THEN30ELSE400
350 CX=CX+1:GOTO410
360 IF POINT(CX+1,CY) THEN30ELSE400
370 CY=CY+1:GOTO410
380 IF POINT(CX,CY+1) THEN30ELSE400
390 CX=CX-1:GOTO410
400 IF POINT(CX-1,CY)=0 THEN 400
410 GOTO240
420 B=RND(3):A=RND(3):B
430 IF POINT(A+5,B) THEN FOR A1=A TO A+5:RESET(A,B):RESET(A1,B+1):SET(A1,B+2,5)
NEXT:RETURN
440 FOR A1=B TO B+2:RESET(A,A1):RESET(A+1,A1):RESET(A+2,A1):RESET(A+3,A1):SET(A+4,A1,5):SET(A+5,A1,5):NEXT:RETURN
450 AA=INKEY$:IF AA="" THEN440
460 IF AA="Y" THEN RUN ELSECLS:END
470 FOR X=1 TO 10:IF SC(S(X)) THEN 430
480 NEXT:GOTO500
490 CLS:PRINTSTRING$(32,255):"CONGRATULATIONS !!":STRING$(32,255)
500 PRINT"YOUR TIME IS IN THE TOP TEN":PRINT:PRINT
510 FOR Y=10 TO X+1 STEP-1:SY=S(CY-1)+N*(Y)-N*(Y-1):NEXT
520 INPUT"ENTER YOUR NAME":N$(X):S(X)=SC
530 CLS:FOR X=1 TO 10:PRINTUSING"### *
*****.##":X;N$(X):S(X):NEXT
Y
540 PRINT"Hit ENTER TO CONTINUE"
550 IF INKEY$=CHR$(13) THEN 550
560 BL=0:GOTO70
570 PHODEC,1:PLCLS:COLOR2,3:SCREEN1,0:PLCLS
580 Y=44:FOR X=20 TO54STEP-2:GOSUB1000:NEXT
590 X=54:FOR Y=44 TO70:GOSUB1000:NEXT
600 Y=70:FOR X=54 TO70STEP2:GOSUB1000:NEXT
610 X=82:FOR Y=44 TO70:GOSUB1000:NEXT
620 Y=44:FOR X=84 TO100STEP2:GOSUB1000:NEXT
630 Y=58:FOR X=84 TO100STEP2:GOSUB1000:NEXT
640 X=100:FOR Y=44 TO70:GOSUB1000:NEXT

```

```

650 Y=80:FOR X=90 TO100:GOSUB1000:Y=Y+1:NEXT
660 X=100:FOR Y=44 TO70:GOSUB1000:NEXT
670 Y=44:FOR X=110 TO126STEP2:GOSUB1000:NEXT
680 X=58:FOR X=110 TO126STEP2:GOSUB1000:NEXT
690 X=126:FOR Y=44 TO70:GOSUB1000:NEXT
700 Y=44:FOR X=134 TO156STEP2:GOSUB1000:NEXT
710 X=156:FOR Y=40 TO68:GOSUB1000:Y=X-1:NEXT
720 Y=70:FOR X=134 TO156STEP2:GOSUB1000:NEXT
730 X=188:FOR Y=44 TO56:GOSUB1000:Y=X-1:NEXT
740 X=184:FOR Y=44 TO56:GOSUB1000:Y=X+1:NEXT
750 X=178:FOR Y=30 TO70:GOSUB1000:NEXT
760 X=54:FOR Y=80 TO114:GOSUB1000:NEXT
770 X=58:FOR Y=80 TO98:GOSUB1000:Y=X+1:NEXT
780 Y=98:FOR X=80 TO70:GOSUB1000:Y=Y-1:NEXT
790 X=78:FOR Y=80 TO114:GOSUB1000:NEXT
800 X=84:FOR Y=80 TO114:GOSUB1000:NEXT
810 Y=88:FOR X=80 TO102STEP2:GOSUB1000:NEXT
820 Y=102:FOR X=80 TO102STEP2:GOSUB1000:NEXT
830 X=102:FOR Y=80 TO114:GOSUB1000:NEXT
840 Y=88:FOR X=110 TO122STEP2:GOSUB1000:NEXT
850 X=132:FOR Y=90 TO112:GOSUB1000:Y=X-1:NEXT
860 Y=114:FOR X=110 TO122STEP2:GOSUB1000:NEXT
870 X=140:FOR Y=80 TO114:GOSUB1000:NEXT
880 Y=88:FOR X=142 TO156STEP2:GOSUB1000:NEXT
890 Y=102:FOR X=142 TO152STEP2:GOSUB1000:NEXT
900 Y=114:FOR X=142 TO158STEP2:GOSUB1000:NEXT
910 Y=88:FOR X=184 TO186STEP-2:GOSUB1000:NEXT
920 X=186:FOR Y=80 TO82:GOSUB1000:NEXT
930 Y=102:FOR X=180 TO184STEP2:GOSUB1000:NEXT
940 X=184:FOR Y=102 TO114:GOSUB1000:NEXT
950 Y=114:FOR X=184 TO160STEP-2:GOSUB1000:NEXT
960 FOR X=10 TO40:NEXT
970 P=1
980 FOR X=1 TO25
990 SCREEN1,P:P=P-1:P:PLAY"255L255ABC":NEXTX:CLS:SCREEN=0:RETURN
1000 PSET(X,Y,2):RETURN
1010 SOUND 2,1:IFCX-1<1 THEN1000ELSERESET(CX-1,CY)
1020 IFCX-2<1 THENRESET(CX-2,CY)
1030 IFCX+1<60 THENRESET(CX+1,CY)
1040 IFCX-2<0 THENRESET(CX-2,CY)
1050 IFCY-1<1 THENRESET(CX,CY-1)
1060 IFCY+1<20 THENRESET(CX,CY+1)
1070 QT=QT+PL:BL=PL+2
1080 GOTO270
1090 CLS:PRINT"INPUT" DO YOU WANT INSTRUCTIONS?;N10:IF LEFT$(N10,1)="" THEN
EN RETURN ELSE CLS:PRINT0,"CRAZY MAZES":PRINT
1100 PRINT"CRAZY MAZES PLACES YOU (A DOT) IN A MAZE. YOU TRY TO GET OUT IN THE
SHORTEST TIME POSSIBLE."PRINT
1110 Q=CHR$(34)
1120 PRINT"HOEVER, THIS IS NO ORDINARY MAZE. THIS IS A "00"CRAZY MAZE"00".
WHAT IS A CRAZY MAZE? IT IS A MAZE WHICH CHANGES IF YOU STAND STILL. ONE SECOND
THERE IS AN OPEN PATH LEADING OUT, AND THE NEXT THING YOU KNOW THERE IS A
DEAD END."
1130 PRINT
1140 PRINT"PRESS ANY KEY TO CONTINUE."
1150 IF INKEY="" THEN1150
1160 CLS:PRINT"YOUR DOT STARTS OUT IN THE UPPERLEFT CORNER OF THE MAZE. US
ING THE ARROW KEYS TO MOVE, TRY TO MANUEVER YOURSELF TO THE ORANGE DOT IN THE
LOWER-RIGHT CORNER OF THE SCREEN."
1170 PRINT"BEFORE YOU APPEAR ON THE SCREEN 200 RANDOMLY PLACED DOTS FLOOD
THE MAZE. THESE SERVE TO BLOCK YOUR PATH AND CONFUSE YOU."
1180 PRINT"PRINT"PRESS ANY KEY TO CONTINUE"
1190 IF INKEY="" THEN1190 ELSE CLS
1200 PRINT"PRINT"YOU HAVE ONE SPECIAL FEATURE. BY PRESSING -SPACEBAR- YOU CAN
BLAST A HOLE IN ALL SURROUNDING WALLS THROUGH WHICH YOU CAN GO."
1210 PRINT"PRINT"UNFORTUNATELY, THIS ADDS TIME TOYOUR SCORE. THE FIRST TIME IT
ADDS 2 SECONDS TO YOUR TIME, THEN 4,6,8,10 AND SO ON."
1220 PRINT"PRINT"0-0-0 L-U-C-K !!"
1230 PRINT
1240 PRINT"PRESS ANY KEY TO CONTINUE"
1250 IF INKEY="" THEN1250
1260 RETURN

```

# GRAPHICS

Το πρόγραμμα αυτό κατασκευάζει πολύ όμορφες γραφικές παραστάσεις. Στην πραγματικότητα είναι πολλά προγράμματα μαζί, τα οποία μπορούν να πληκτρολογηθούν και να "τρέξουν" χωριστά το καθένα, ή να ενσωματωθούν σε άλλα προγράμματα για πιο όμορφη παρουσίαση τίτλων, επιταγών κλπ. Τα προγράμματα αυτά αρχίζουν στις γραμμές 20, 200, 290, 340, 420, 590, 660, 760, 1090, 1170, 1270. Ολόκληρο το πρό-

γραμμα διαρκεί συνολικά πεντέμισι λεπτά. Οι γραφικές παραστάσεις είναι όλες μαυρόασπρες γιατί χρησιμοποιείται η υψηλότερη ανάλυση του κομπιούτερ, 256X192 (PMODE 4). Για να τρέξει το πρόγραμμα, είναι απαραίτητη η Extended Basic και 16K μνήμης. Αν την πρώτη φορά που "τρέξετε" το πρόγραμμα διαπιστώσετε κάποια μεγάλη παραμόρφωση και σχήματα ή πάρετε ένα ? SN ERROR, πατήστε το πλήκτρο "BREAK" και "ξανατρέξτε" το απ' την αρχή.

```

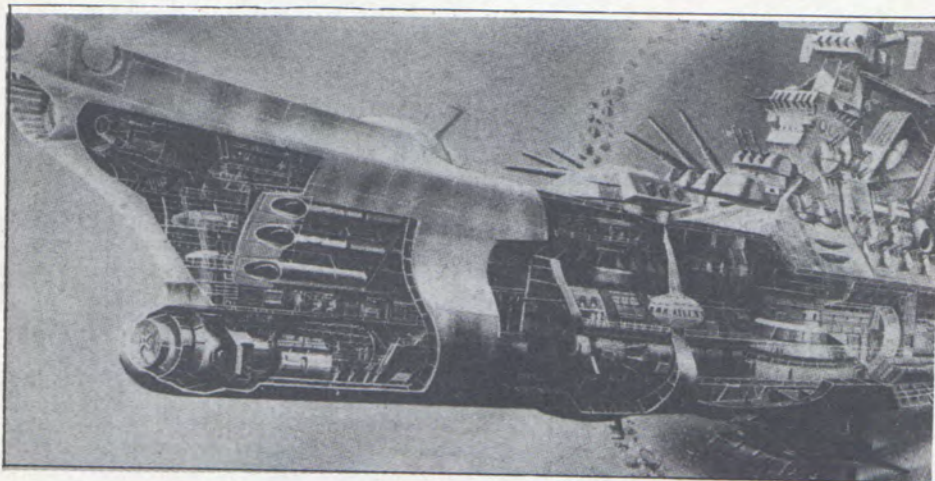
5 * *** GRAPHICS PROGRAM ***
10 OSUB 1170
20 PMODE 4,1:PCLS:SCREEN1,1
30 FOR X=0 TO 95 STEP 2:LINE(126,96),@X,NEXT
40 FOR X=0 TO 25:LINE(126,45),@X,NEXT
50 FOR X=0 TO 25 STEP 2:LINE(126,45),@X,NEXT
60 FOR X=0 TO 25:LINE(192,96),@X,NEXT
70 FOR X=0 TO 25 STEP 2:LINE(192,96),@X,NEXT
80 FOR X=0 TO 25:LINE(126,147),@X,NEXT
90 FOR X=0 TO 25 STEP 2:LINE(126,147),@X,NEXT
100 FOR X=0 TO 25:LINE(64,96),@X,NEXT
110 FOR X=0 TO 25 STEP 2:LINE(64,96),@X,NEXT
120 FOR X=0 TO 50:LINE(126,96),@X,NEXT
130 FOR X=0 TO 50 STEP 2:LINE(126,96),@X,NEXT
140 FOR X=0 TO 95 STEP 4:LINE(0,96)-(X,0),PSET:NEXT
150 FOR X=0 TO 95 STEP 4:LINE(0,96)-(X,192),PSET:NEXT
160 FOR X=250 TO 161 STEP 4:LINE(256,96)-(X,0),PSET:NEXT
170 FOR X=250 TO 161 STEP 4:LINE(256,96)-(X,192),PSET:NEXT
180 FOR X=70 TO 122 STEP 4:LINE(33,96)-(0,X),PSET:NEXT
190 FOR X=70 TO 122 STEP 4:LINE(223,96)-(256,X),PSET:NEXT
200 FOR X=102500:NEXT:PCLS
210 FOR X=0 TO 96 STEP 2:LINE(0,0)-(126,X),PSET:NEXT
220 FOR X=126 TO 0 STEP 2:LINE(0,0)-(X,96),PSET:NEXT
230 FOR X=0 TO 96 STEP 2:LINE(256,0)-(126,X),PSET:NEXT
240 FOR X=126 TO 256 STEP 2:LINE(256,0)-(X,96),PSET:NEXT
250 FOR X=0 TO 126 STEP 2:LINE(0,192)-(X,96),PSET:NEXT
260 FOR X=96 TO 192 STEP 2:LINE(0,192)-(126,X),PSET:NEXT
270 FOR X=256 TO 126 STEP 2:LINE(256,192)-(X,96),PSET:NEXT
280 FOR X=96 TO 192 STEP 2:LINE(256,192)-(126,X),PSET:NEXT
290 FOR X=102500:NEXT:PCLS
300 FOR X=0 TO 192 STEP 2:LINE(0,0)-(126,X),PSET:NEXT
310 FOR X=0 TO 192 STEP 2:LINE(256,0)-(126,X),PSET:NEXT
320 FOR X=126 TO 0 STEP 2:LINE(0,0)-(X,192),PSET:NEXT
330 FOR X=126 TO 256 STEP 2:LINE(256,0)-(X,192),PSET:NEXT
340 FOR X=102000:NEXT:PCLS:SCREEN1,1
350 FOR X=0 TO 145 STEP 2:LINE(126,96),@X,2:NEXT
360 FOR X=0 TO 95 STEP 2:LINE(126,48),@X,2:NEXT
370 FOR X=0 TO 95 STEP 2:LINE(126,48),@X,2:NEXT
380 FOR X=0 TO 95 STEP 2:LINE(126,144),@X,2:NEXT
390 FOR X=0 TO 95 STEP 2:LINE(126,144),@X,2:NEXT
400 FOR X=0 TO 40:LINE(126,96),@X,NEXT
410 FOR X=0 TO 40 STEP 2:LINE(126,96),@X,NEXT
420 FOR X=102000:NEXT:PCLS
430 FOR X=0 TO 126 STEP 2:LINE(0,48)-(X,0),PSET:NEXT
440 FOR X=0 TO 96 STEP 2:LINE(0,48)-(X,96),PSET:NEXT
450 FOR X=126 TO 0 STEP 2:LINE(0,48)-(X,96),PSET:NEXT
460 FOR X=96 TO 0 STEP 2:LINE(0,48)-(0,X),PSET:NEXT
470 FOR X=126 TO 256 STEP 2:LINE(192,48)-(X,0),PSET:NEXT
480 FOR X=0 TO 96 STEP 2:LINE(192,48)-(X,96),PSET:NEXT
490 FOR X=256 TO 126 STEP 2:LINE(192,48)-(126,X),PSET:NEXT
500 FOR X=96 TO 0 STEP 2:LINE(192,48)-(126,X),PSET:NEXT
510 FOR X=126 TO 256 STEP 2:LINE(192,144)-(X,96),PSET:NEXT
520 FOR X=96 TO 192 STEP 2:LINE(192,144)-(256,X),PSET:NEXT
530 FOR X=256 TO 126 STEP 2:LINE(192,144)-(X,192),PSET:NEXT
540 FOR X=192 TO 96 STEP 2:LINE(192,144)-(126,X),PSET:NEXT
550 FOR X=0 TO 126 STEP 2:LINE(0,144)-(X,96),PSET:NEXT
560 FOR X=96 TO 192 STEP 2:LINE(0,144)-(126,X),PSET:NEXT
570 FOR X=126 TO 0 STEP 2:LINE(0,144)-(126,X),PSET:NEXT
580 FOR X=192 TO 96 STEP 2:LINE(0,144)-(0,X),PSET:NEXT
590 FOR X=101000:NEXT
600 FOR X=0 TO 96:LINE(0,48)-(126,X),PSET:NEXT
610 FOR X=0 TO 96:LINE(192,48)-(126,X),PSET:NEXT
620 FOR X=96 TO 192:LINE(0,144)-(126,X),PSET:NEXT
630 FOR X=96 TO 192:LINE(192,144)-(126,X),PSET:NEXT
640 FOR X=0 TO 70 STEP 2:LINE(126,48),@X,NEXT
650 FOR X=0 TO 70 STEP 2:LINE(126,144),@X,NEXT
660 FOR X=102500:NEXT:PCLS
670 FOR X=0 TO 256 STEP 2:LINE(126,96)-(X,0),PSET:NEXT
680 FOR X=0 TO 192 STEP 2:LINE(126,96)-(X,192),PSET:NEXT
690 FOR X=256 TO 0 STEP 2:LINE(126,96)-(X,192),PSET:NEXT
700 FOR X=192 TO 0 STEP 2:LINE(126,96)-(0,X),PSET:NEXT:FOR X=10750:NEXT
710 FOR X=0 TO 192:LINE(126,96)-(X,48),PSET:NEXT
720 FOR X=48 TO 144:LINE(126,96)-(192,X),PSET:NEXT

```

```

730 FOR X=192 TO 64 STEP 2:LINE(126,96)-(X,144),PSET:NEXT
740 FOR X=144 TO 48 STEP 2:LINE(126,96)-(0,X),PSET:NEXT
750 FOR X=0 TO 96 STEP 2:LINE(126,96),@X,NEXT
760 FOR X=0 TO 2500:NEXT:PCLS
770 FOR X=0 TO 48 STEP 2:LINE(0,0)-(X,48),PSET:NEXT
780 FOR X=48 TO 0 STEP 2:LINE(0,0)-(X,48),PSET:NEXT
790 FOR X=0 TO 48 STEP 2:LINE(126,0)-(0,X),PSET:NEXT
800 FOR X=0 TO 126 STEP 2:LINE(126,0)-(X,48),PSET:NEXT
810 FOR X=126 TO 64 STEP 2:LINE(126,0)-(X,48),PSET:NEXT
820 FOR X=64 TO 96 STEP 2:LINE(126,0)-(0,X),PSET:NEXT
830 FOR X=0 TO 64 STEP 2:LINE(0,0)-(X,48),PSET:NEXT
840 FOR X=64 TO 96 STEP 2:LINE(0,0)-(0,X),PSET:NEXT
850 FOR X=0 TO 48 STEP 2:LINE(126,0)-(192,X),PSET:NEXT
860 FOR X=192 TO 126 STEP 2:LINE(126,0)-(X,48),PSET:NEXT
870 FOR X=0 TO 48 STEP 2:LINE(256,0)-(192,X),PSET:NEXT
880 FOR X=192 TO 256 STEP 2:LINE(256,0)-(X,48),PSET:NEXT
890 FOR X=256 TO 96 STEP 2:LINE(256,0)-(192,X),PSET:NEXT
900 FOR X=96 TO 192 STEP 2:LINE(256,0)-(192,X),PSET:NEXT
910 FOR X=126 TO 192 STEP 2:LINE(126,96)-(X,48),PSET:NEXT
920 FOR X=48 TO 96 STEP 2:LINE(126,96)-(192,X),PSET:NEXT
930 FOR X=96 TO 144 STEP 2:LINE(126,96)-(192,X),PSET:NEXT
940 FOR X=192 TO 126 STEP 2:LINE(126,96)-(X,144),PSET:NEXT
950 FOR X=96 TO 144 STEP 2:LINE(256,96)-(192,X),PSET:NEXT
960 FOR X=192 TO 256 STEP 2:LINE(256,96)-(X,144),PSET:NEXT
970 FOR X=256 TO 192 STEP 2:LINE(256,192)-(X,144),PSET:NEXT
980 FOR X=144 TO 192 STEP 2:LINE(256,192)-(192,X),PSET:NEXT
990 FOR X=126 TO 192 STEP 2:LINE(126,192)-(X,144),PSET:NEXT
1000 FOR X=144 TO 192 STEP 2:LINE(126,192)-(192,X),PSET:NEXT
1010 FOR X=96 TO 144 STEP 2:LINE(0,96)-(0,X),PSET:NEXT
1020 FOR X=0 TO 0 STEP 2:LINE(0,96)-(0,X),PSET:NEXT
1030 FOR X=0 TO 144 STEP 2:LINE(126,96)-(0,X),PSET:NEXT
1040 FOR X=0 TO 126 STEP 2:LINE(126,96)-(X,144),PSET:NEXT
1050 FOR X=126 TO 64 STEP 2:LINE(126,192)-(X,144),PSET:NEXT
1060 FOR X=144 TO 192 STEP 2:LINE(126,192)-(0,X),PSET:NEXT
1070 FOR X=0 TO 64 STEP 2:LINE(0,192)-(X,144),PSET:NEXT
1080 FOR X=144 TO 192 STEP 2:LINE(0,192)-(0,X),PSET:NEXT
1090 FOR X=102500:NEXT:PCLS
1100 PCLS:PMODE 4,1:SCREEN1,1:FOR X=0 TO 126 STEP 2:LINE(0,96)-(X,0),PSET:NEXT
1110 FOR X=0 TO 192 STEP 2:LINE(0,96)-(126,X),PSET:NEXT
1120 FOR X=126 TO 0 STEP 2:LINE(0,96)-(X,192),PSET:NEXT
1130 FOR X=256 TO 126 STEP 2:LINE(256,96)-(X,0),PSET:NEXT
1140 FOR X=0 TO 192 STEP 2:LINE(256,96)-(126,X),PSET:NEXT
1150 FOR X=126 TO 256 STEP 2:LINE(256,96)-(X,192),PSET:NEXT
1160 FOR X=102500:NEXT:GOTO 20
1170 PCLS:SCREEN1,1:PMODE 4,1
1180 DRAW"BP10,20:115R2D7L5R500R13R2L7U15BR10D15BR7U15L3R2B7R4B4D15BR10R7L7U15
BR10R2L7D7R2D8R2R13U15R5F2D5L7R8F2D5D2L7R17U15R5B8L4D15BR15U5R7D15L7R9L5U1
5R2L7D8R2D8R16R1BR10R1BR10R1BR10R1BR10R1BR10R1"
1190 FOR X=36 TO 24 STEP 4:LINE(126,96)-(0,X),PSET:NEXT
1200 FOR X=0 TO 256 STEP 4:LINE(126,96)-(X,24),PSET:NEXT
1210 FOR X=24 TO 96 STEP 4:LINE(126,96)-(256,X),PSET:NEXT
1220 DRAW"BP34,100:R13L13D20R15U10L10BR20U10R15D10L10U10R7F10BR8U20R15D10L15R
15D10BR10U20R15D10L15R15BR10U10D20U10R15U10D20BR10U20BR10R15L15D20R15BR10R15U10L1
5U10R15"
1230 FOR X=124 TO 192 STEP 4:LINE(126,124)-(0,X),PSET:NEXT
1240 FOR X=0 TO 256 STEP 4:LINE(126,124)-(X,192),PSET:NEXT
1250 FOR X=192 TO 124 STEP 4:LINE(126,124)-(256,X),PSET:NEXT
1260 FOR X=104000:NEXT:PCLS
1270 PMODE 4,1:PCLS:SCREEN1,1
1280 DRAW"BP34,24:R13L13D20R15U10L10BR20U10R15D10L15U10R7F10BR8U20R15D10L15R1
5D10BR10U20R15D10L15R15BR10U10D20U10R15U10D20BR10R15L15D20R15BR10R15U10L1
5U10R15"
1290 DRAW"BP80,55:R10D10L10U10BR17D10U10F10U10BR20R10L50BR10U10D5R10U5D10BR7R1
0L10UR10L10UR10"
1300 DRAW"BP32,80:R26L13D30BR10U30R23D120L23R12F13BR3R23E3U3H3L23H3U3E3R23BR3
0D15R15BR12H3U3E3R20F3D30L30D30F3R20E3U3H3BR12U12E3R20F3D24G3L20H3U12"
1310 DRAW"BP67,120:R15L15D20R15BR10R15L15U20R15D20BR10U20C20R15BR10R15L15U20R15
D20BR10U20R15D10L15R5F10"
1320 DRAW"BP32,150:R15L15D20R15BR10R15L15U20R15D20BR10U20R20D0R0R0C20BR10U20R15D
15BR15BR10U20C20R15U20BR10R15L15D20BR17R15L15U10R15L15U10R15BR10C20U20R15D10L15
8F10"
1330 FOR X=1 TO 3500:NEXT:PCLS
1340 RETURN

```



φορές, έχετε χάσει το παιχνίδι.

Το παιχνίδι έχει επίσης ωραία ηχητικά εφέ. Υπάρχει ακόμα η δυνατότητα να χρησιμοποιήσετε την γρήγορη ταχύτητα του κομπιούτερ. Όμως δεν μπορούν όλοι οι Color Computers να λειτουργήσουν στη γρήγορη ταχύτητα. Γι'αυτό το πρόγραμμα σας ρωτάει στην αρχή του προγράμματος αν ο δικός σας κομπιούτερ μπορεί να εργασθεί με την γρήγορη ταχύτητα, και αν του απαντήσετε "YES", τότε στη γραμμή 500 του προγράμματος εκτελείται η εντολή POKE 65495,0. Έτσι αυξάνεται κατά πολύ η ταχύτητα του παιχνιδιού. Για να επαναφέρετε τον κομπιούτερ σας στην κανονική του ταχύτητα, πατήστε το κουμπί "RESET" στο πίσω μέρος. Το πρόγραμμα για να τρέξει χρειάζεται την Extended Basic και 16K RAM.

Το αντικείμενο του παιχνιδιού αυτού είναι να βυθίσετε τα υποβρύχια του εχθρού ρίχνοντας βόμβες βυθού από το πλοίο σας. Οι βόμβες βυθού παριστάνονται σαν άσπρες κουκίδες και μπορείτε να τις ρίξετε πατώντας το πλήκτρο "↑" ή το πλή-

κτρο "Q". Για να κινήσετε το πλοίο σας δεξιά ή αριστερά στην οθόνη χρησιμοποιήστε τα αντίστοιχα βελάκια στο πληκτρολόγιο. Ανάλογα με το χρώμα του υποβρυχίου που θα χτυπήσετε συγκεντρώνετε από 10 ως 40 βαθμούς. Προσέξτε όμως! Τα υπο-

βρύχια εκτός του ότι κινούνται γρήγορα, έχουν και πολυβόλα με τα οποία πυροβολούν συνέχεια εναντίον του πλοίου σας. Οι βολές των υποβρυχίων παριστάνονται με μπλε κουκίδες. Αν καταστραφεί το πλοίο σας περισσότερο από τρεις

## ΠΟΛΕΜΟΣ ΣΤΗ ΘΑΛΛΑΣΑ

```

5 * *** QUICK THINK ***
10 CLS(3):PRINT(30,"QUICK THINK":PRINT(223,"DO YOU NEED INSTRUCTIONS");PRINT(
200,"");INPUT I#
20 IF MID$(I#,1,1)="" THEN 30 ELSE 100
30 CLS(3):PRINT(10,"QUICK THINK";PRINT(004,"QUICK THINK IS A TWO PLAYER
E WHICH IS PLAYED ON A 3X7 GRID. PLAYERS TAKE TURNS BY ENTERING A NUMBER F
ROM 1 TO 7. THE DEALER WILL THEN SHOVE OUT A CHIP ON THE NUMBER RACK."
40 PRINT:PRINT(10,"THE OBJECT OF THE GAME IS TO TRY TO LINE-UP FOUR OF YOUR
CHIPS IN A ROW LIKE YOU WOULD IN TIC-TAC-TOE."
50 PLAY(70):FOR X=1 TO 5
60 PLAY(01):FOR Y=1 TO 4
70 PLAY(CDEF0AB)+0
80 NEXT Y:NEXT X
90 PRINT(454,"HIT ANY KEY TO CONT.":T#=INKEY$:IF T#="" THEN 30
100 CLS(3):PRINT(022,"THE CHALLENGING PART ABOUT QUICK THINK IS THE TIMER
FUNCTION: THIS FUNCTION WILL ALLOW YOU AS MUCH AS 59 SECONDS OR
AS LITTLE AS 1 SECOND TO MAKE YOUR MOVE OTHERWISE THE COMPUTER WILL
L"
110 PRINT(100,"MAKE THE MOVE FOR YOU.":PRINT:PRINT(100,"THE PLAYER WHO WANTS TO GO
FIRST IS BLUE, THE PLAYER WHO GOES SECOND IS RED"
120 PLAY(70):FOR X=1 TO 5
130 PLAY(05):FOR Y=1 TO 4
140 PLAY(BAQFDCD-T)+NEXT Y:NEXT X
150 PRINT(454,"HIT ANY KEY TO CONT.":T#=INKEY$:IF T#="" THEN 150
160 CLS(3):PRINT(100,"DO YOU WANT THE TIMER FUNCTION (Y/N)");INPUT T#
170 IF MID$(T#,1,1)="" THEN T#="Y":PRINT(132,"HOW MUCH TIME (1-59)");INPUT TI:
IF TI<1 OR TI>59 THEN 170
180 TI=INT(TI)+.88:TI
190 DIM A(196)
200 P=1
210 P#MODE1
220 PCLS
230 SCREEN 1,1
240 DRAW"C4;R#0,20;R105R47R100D154L255UT34"
250 FOR X=30 TO 210 STEP 30:LINE(X,174)-(X,70),PSET:LINE(X-1,174)-(X-1,70),PSET:
LINE(X+1,174)-(X+1,70),PSET:NEXT X
260 DRAW"BR#3,178;D12BR20R06UR06UL06R30R06DL06L06L06R30R06UL06UR06R06R06DL06
R30R06UL06UL12R06R22R06D12"
270 DRAW"BR#120,5;R10F3D110L10H3U11E3BL3BR06D02R3U2B0L3" :HEAD
280 GOSUB 290:GOSUB 300:GOSUB 310:GOTO 330
290 DRAW"C4;R#17,18;L100D010F0R06H0Z1BR06D10" :RETURN:"LEFT ARM
300 DRAW"C4;R#139,19;R10F2D1006L6E0U7H1L248D10" :RETURN:"RIGHT ARM
310 IF P=1 THEN DRAW"BR#0,0;D10R5BU1D10R4D10F10R3D10F10R3D10L700L45" ELSE DRAW"R
#106,8;D10L3BU1D10L4D10L10L3D10;D10L3D10R20B025"
320 DRAW"BR#0,0;R#4ND4E4BR06DL06URR12D0F2R4E2U0BR4ND06R0D4L0R3F4BL33R010UL4R06

```

```

R4D0F2R4E2U0BR4ND06R0D4L0R4F4BR4UL06R08" :RETURN:"YOUR TURN
330 GOSUB 300:DRAW"BR#120,30"
340 GOSUB 1020
350 DRAW"C5;R#234,50;RBL0D12R4U12R4D12L0U12R4D12UBL4" :DRAW"C3;R#10,0" :TIMER=0:
D=0:BB=TI:CC=0:FD=0:CN=0:T=0
360 I#="" :I#=INKEY$
370 IF I#="" THEN GOTO 700
380 IF I#="" THEN 300
390 I#VAL(I#):IF PPOINT(I#+(I#*3),80)<5 THEN 1300
400 IF I#<1 OR I#>7 THEN 300
410 DRAW"BR#120,45" :GOSUB 1020
420 IF I#<4 THEN DRAW"C5;R#149,18;F2D1006L6E0U7H1" :DRAW"C4;R#149,18;R2F1006L6E0H1
06F100L190D5" :GOSUB 1020 :FOR PA=1 TO 450 :NEXT PA:GOSUB 1020
430 IF I#<4 THEN DRAW"C5;R#149,18;R2F1006L6E0H106F100L190D5" :GOSUB 300 :GOSUB 1020
:GOSUB 1020
440 IF I#>4 THEN DRAW"C5;R#107,18;G2D10F0R06H0Z1" :DRAW"C4;R#107,18;L201006R06U0E1
06L06D15" :GOSUB 1020 :FOR PA=1 TO 450 :NEXT PA:GOSUB 1020
450 IF I#>4 THEN DRAW"C5;R#107,18;L201006R06U0E106D20" :GOSUB 290 :GOSUB 1020 :GOSUB
1020
460 ON I GOTO 470,480,490,500,510,520,530
470 FOR Z=1 TO 10:DRAW"BR#-3,+3" :GOSUB 1020:IF Z=10 THEN 540 ELSE GOSUB 1020:NEXT
Z
480 FOR Z=1 TO 10:DRAW"BR#-0,+3" :GOSUB 1020:IF Z=10 THEN 540 ELSE GOSUB 1020:NEXT
Z
490 FOR Z=1 TO 10:DRAW"BR#-3,+3" :GOSUB 1020:IF Z=10 THEN 540 ELSE GOSUB 1020:NEXT
Z
500 FOR Z=1 TO 10:DRAW"BR#0,+3" :GOSUB 1020:IF Z=10 THEN 540 ELSE GOSUB 1020:NEXT
Z
510 FOR Z=1 TO 10:DRAW"BR#0,+3" :GOSUB 1020:IF Z=10 THEN 540 ELSE GOSUB 1020:NEXT
Z
520 FOR Z=1 TO 10:DRAW"BR#0,+3" :GOSUB 1020:IF Z=10 THEN 540 ELSE GOSUB 1020:NEXT
Z
530 FOR Z=1 TO 10:DRAW"BR#3,+3" :GOSUB 1020:IF Z=10 THEN 540 ELSE GOSUB 1020:NEXT
Z
540 FOR PA=1 TO 150 :NEXT PA:GOSUB 1020
550 X=30+((I-1)*30)
560 FOR Y=20 TO 150 STEP 10
570 IF PPOINT(X,Y,10)<5 THEN 590
580 GOSUB 1020:GOSUB 1020:DRAW"C4;R#10,0" :LINE(X,70)-(X,Y+8),PSET:LINE(X-1,70)
-(X-1,Y+8),PSET:LINE(X+1,70)-(X+1,Y+8),PSET:DRAW"BR#-3,+3" :NEXT Y
590 GOSUB 1020
600 DRAW"C5" :GOSUB 310
610 B=20+((170-Y)/10)*3:R#B+1:IF P=1 THEN A(B)=7 ELSE A(B)=8
620 IF P=1 THEN S=7 ELSE S=8
630 IF A(B-1)<5 AND A(B-2)<5 AND A(B-3)<5 THEN R1=0:R2=-13:R3=-26:R4=-33:GOTO
1100:"UP AND DOWN

```

```

040 IF (A(B-1)=S AND A(B-2)=S AND A(B-3)=S) THEN R1=-3:R2=-2:R3=-1:R4=0:GOTO 1100
'LEFT AND RIGHT
050 IF (A(B-1)=S AND A(B-2)=S AND A(B-1)=S) THEN R1=-2:R2=-1:R3=0:R4=1:GOTO 1100
060 IF (A(B-1)=S AND A(B-1)=S AND A(B-2)=S) THEN R1=-1:R2=0:R3=1:R4=2:GOTO 1100
070 IF (A(B-1)=S AND A(B-2)=S AND A(B-3)=S) THEN R1=0:R2=1:R3=2:R4=3:GOTO 1100
080 IF (A(B-1)=S AND A(B-2)=S AND A(B-4)=S) THEN R1=-4:R2=-3:R3=-2:R4=0:GOTO
1100 'UPPER LEFT TO LOWER RIGHT
090 IF (A(B-14)=S AND A(B-28)=S AND A(B-42)=S) THEN R1=-28:R2=-14:R3=0:R4=14:GOTO
1100
100 IF (A(B-14)=S AND A(B-14)=S AND A(B-28)=S) THEN R1=-14:R2=0:R3=14:R4=28:GOTO 1
100
110 IF (A(B-14)=S AND A(B-28)=S AND A(B-42)=S) THEN R1=0:R2=14:R3=28:R4=42:GOTO 11
00
120 IF (A(B-12)=S AND A(B-24)=S AND A(B-36)=S) THEN R1=-36:R2=-24:R3=-12:R4=0:GOTO
1100
130 IF (A(B-12)=S AND A(B-24)=S AND A(B-12)=S) THEN R1=-24:R2=-12:R3=0:R4=12:GOTO
1100
140 IF (A(B-12)=S AND A(B-12)=S AND A(B-24)=S) THEN R1=-12:R2=0:R3=12:R4=24:GOTO 1
100
150 IF (A(B-12)=S AND A(B-24)=S AND A(B-36)=S) THEN R1=0:R2=12:R3=24:R4=36:GOTO 11
00
700 GOTO 390
720 IF T=0 THEN 390
780 CC=INT(T/(TIMER/60))
790 IF BB=CC THEN 380
800 IF CC=3 THEN 310
810 IF CC=0 THEN 380
820 GOSUB 380
830 CN=CC
840 DRAW"C5":BM220,50:IT=CN:GOSUB 1250:DRAM"C3":BM220,50:IT=CN:GOSUB 1250:IC=CC
850 IF CC=4 THEN PLAY"01T10L100FF"
860 IF CC=5 THEN PLAY"01T10L100FF"
870 IF CC=2 THEN PLAY"01T10L100EE"
880 IF CC=1 THEN PLAY"01T10L100DD"
890 IF CC=0 THEN PLAY"01U25T4L4CCU15":R=AND(7):I=0:GOTO 410
900 GOTO 380
910 IF BB=CC THEN 370
920 IF CC=3 AND CC=20 THEN FD=1:CO=BB-18:CN=CC-18:IF CC=19 OR TIMER<100 THEN GOS
UB 380:DRAM"BM220,50:RBD0L80LB"
930 IF CC=19 AND CC=30 THEN FD=2:CO=BB-20:CN=CC-20:IF CC=29 OR TIMER<100 THEN GOS
UB 380:DRAM"BM220,50:RBD0L80LB"
940 IF CC=29 AND CC=40 THEN FD=3:CO=BB-30:CN=CC-30:IF CC=39 OR TIMER<100 THEN GOS
UB 380:DRAM"BM220,50:RBD0L80LB"
950 IF CC=39 AND CC=50 THEN FD=4:CO=BB-40:CN=CC-40:IF CC=49 OR TIMER<100 THEN GOS
UB 380:DRAM"BM220,50:RBD0L80LB"
960 IF CC=49 THEN FD=5:CO=BB-50:CN=CC-50:IF CC=59 OR TIMER<100 THEN GOSUB 380:OR
AM"BM220,50:RBD0L80LB"
970 BB=CC:GOTO 840
980 DRAW"C5":BM220,50:ID12R4NU12R4UNL80UB" :DRAM"C4":BM10,0" :RETURN
990 IF P=1 THEN P=2 ELSE P=1
1000 DRAW"C4":GOSUB 310
1010 GOTO 330
1020 'MAKE CHIP APPEAR

```

```

1030 IF P=1 THEN DRAW"C7":BM10,0" ELSE DRAW"C8":BM10,0"
1040 DRAW"BM10,0:ND0R160D2L160D2R160D2NU0L16" :PLAY"L200:1200:U30:01:A:U15" :FOR
PA=1 TO 80:NEXT PA
1050 RETURN
1060 'MAKE CHIP DISAPPEAR
1070 DRAW"C5":BM10,0" :ND0R160D2L160D2R160D2NU0L160UB"
1080 IF P=1 THEN DRAW"C7":BM10,0" ELSE DRAW"C8":BM10,0"
1090 RETURN
1100 RET
1110 DRAW"C5":GOSUB 310
1120 FOR U=1 TO 5:DRAM"C4":BM25,41" :GOSUB 1230:DRAM"C4":BM170,41" :GOSUB 1230:FOR P
A=1 TO 200:NEXT PA:DRAM"C5":BM25,41" :GOSUB 1230:DRAM"C5":BM170,41" :GOSUB 1230:FOR
PA=1 TO 200:NEXT PA
1130 NEXT U
1140 FOR U=1 TO 4
1150 B1=R1+B:GOSUB 1020:B1=R2+B:GOSUB 1220:GOSUB 1020:B1=R3+B:GOSUB 1
220:GOSUB 1020:B1=R4+B:GOSUB 1220:GOSUB 1020
1160 B1=R1+B:GOSUB 1220:DRAM"BM10,0" :GOSUB 1070:B1=R2+B:GOSUB 1220:DRAM"BM10,0"
:GOSUB 1070:B1=R3+B:GOSUB 1220:DRAM"BM10,0" :GOSUB 1070:B1=R4+B:GOSUB 1220:DRAM
"BM10,0" :GOSUB 1070
1170 FOR PA=1 TO 150:NEXT PA:NEXTU
1180 FOR PA=1 TO 1000:NEXT PA
1190 PLAY"05" :FOR X=1 TO 12
1200 PLAY"05" :FOR L=1 TO 4:PLAY "BAOFEDCT+0-" :NEXT L:NEXT X
1210 CLS:PRINT@200,"PLAY AGAIN!" :END
1220 Q1=INT(B1/13):Q0=B1-(Q1*13):Q3=(Q1*13)+3:Q2=(Q1*13)+0:Q4=INT(Q3/13):Q5=(Q3-13)/13
:Q6=L:LINE(X1-Q1,Y1-Q2)-(X1-Q2,Y1-Q2):PRESET:RETURN
1230 DRAW"BM10,0:ID12R4NU12R4UNL80UB" :A WINNER!240 RETURN
280DL8R5F0" :PLAY"03:1250:1270,1280,1290,1300,1310,1320,1330,1340,1350
1250 T=T+1:ON T GOTO 1260,1270,1280,1290,1300,1310,1320,1330,1340,1350
1260 DRAW"BM-4,0:RBD0L80R80UB" :RETURN:120
1270 DRAW"BM-4,0:RBD0L80R80UB" :RETURN:1
1280 DRAW"BM-4,0:RBD0L80R80UB" :RETURN:2
1290 DRAW"BM-4,0:RBD0L80R80UB" :RETURN:3
1300 DRAW"BM-4,0:RBD0L80R80UB" :RETURN:4
1310 DRAW"BM-4,0:RBD0L80R80UB" :RETURN:5
1320 DRAW"BM-4,0:RBD0L80R80UB" :RETURN:6
1330 DRAW"BM-4,0:RBD0L80R80UB" :RETURN:7
1340 DRAW"BM-4,0:RBD0L80R80UB" :RETURN:8
1350 DRAW"BM-4,0:RBD0L80R80UB" :RETURN:9
1360 IF PPOINT(41,80)<5 AND PPOINT(71,80)<5 AND PPOINT(101,80)<5 AND PPOINT(131,80)<5 THEN
31,80)<5 AND PPOINT(161,80)<5 AND PPOINT(191,80)<5 AND PPOINT(221,80)<5 THEN
1380
1370 GOTO 360
1380 'TIE
1390 PLAY"120"
1400 FOR X=1 TO 20
1410 PLAY"05" :FOR Y=1 TO 4
1420 PLAY"BAOFEDC"
1430 PLAY"1+0-"
1440 NEXT Y:NEXTX
1450 CLS:PRINT@230,"TIE GAME":END

```

Αυτό το παιχνίδι, που παίζεται από δυο παίχτες, θα σας ακονίσει αρκετά το μυαλό. Μόλις "τρέξετε" το πρόγραμμα, εμφανίζεται στην οθόνη ένα ανθρωπάκι, ο "dealer" και επτά αριθμημένες κολωνίτσες, καθεμία απ'τις οποίες χωράει εννέα κρίκους. Οι παίχτες με τη σειρά τους διαλέγουν ένα αριθμό από το 1 ως το 7 και ο dealer ρίχνει ένα κρίκο στο κολωνάκι με το ίδιο νούμερο. Σκοπός του παιχνιδιού είναι να ρίξετε συνεχόμενους κρίκους σε ένα από τα κολωνάκια, οπότε κερδίζετε και το παιχνίδι. Πρώτος παίζει ο παίχτης με τους μωβ κρίκους και ακολουθεί ο παίχτης με τους πορτοκαλί. Υπάρχει η προ-

αιρετική δυνατότητα χρησιμοποίησης χρονόμετρου, οπότε το παιχνίδι γίνεται αρκετά πιο ενδιαφέρον και δύσκολο. Το χρονόμετρο μπορεί να ρυθμισθεί για χρόνους από 1 ως 59 δευτερόλεπτα. Μέσα σ' αυτόν το χρόνο, ο παίχτης που έχει σειρά πρέπει να εκτελέσει την κίνησή του, αλλιώς ο κομπιούτερ εκτελεί τυχαία μια κίνηση αντί γι' αυτόν.

Μπορείτε βέβαια να βάλετε τον κομπιούτερ να παίξει εξ'ολοκλήρου το παιχνίδι και για τους δυο παίχτες και εσείς να είσαστε απλοί θεατές. Για να "τρέξει" το πρόγραμμα χρειάζεται απαραίτητα η Extended Basic και 16K μνήμης RAM.

# 5 ΚΡΙΚΟΙ-ΕΝΑ ΤΑΛΗΡΟ

```

5 ' *** DESTROY ***
10 PMODE 1
20 CLS:PRINT @ 234,"(ONE MOMENT)"
30 COLOR 1,3
40 PCLS
50 LINE (0,77)-(256,77),PSET
60 PAINT (0,0),1,1
70 DRAW "BM75,76:R104E28L60U26L48D26L60F28"
80 PAINT (76,74),2,2
90 DRAW "BM00,47:J4:J6L80R100LB"
100 PAINT (62,46),3,4:PAINT (59,46),3,4
110 DRAW "BM196,47:J4:J6L80R100LB"
120 PAINT (196,46),3,4:PAINT (193,46),3,4
130 DRAW "BM76,176:J4:J6B0U30L48UBR0UBL16D16L48"
140 CIRCLE (76,101),16,4,1,25,.75
150 CIRCLE (176,101),16,4,1,25,.75
160 PAINT (76,174),4,4
170 DRAW "C2:58"
180 DRAW "EM58,84:ND2R6F2D4Q2L6"
190 DRAW "EM78,84:NR8D4NR8D4RB"
200 DRAW "EM96,84:NR8D4R8D4LB"
210 DRAW "EM118,84:R4NDR4"
220 DRAW "BM138,84:R8D4L4NF4L4D4UB"

```



# Commodore 64

## Τό καλύτερο οικιακό κομπιούτερ στον κόσμο.

«Τό Commodore 64 είναι τό πιο ξεχωριστό προϊόν στή βιομηχανία κομπιούτερ από τή μέρα τής γέννησής τους».

Αυτή ήταν ή γνώμη του κριτικού / αναλυτή κομπιούτερ στήν παγκόσμια γνωστή Shearson/American Express όταν παρουσίασαν αυτό τό καταπληκτικό προσωπικό κομπιούτερ.

Οί λόγοι γιά τούς όποιους έκαναν μία τέτοια δήλωση γίνονται εμφανείς αν διαβάσετε τί προσφέρει τό Commodore 64:

- Ενσωματωμένη μνήμη 64 K
- 16 χρώματα από τό πληκτρολόγιο

- Γραφικές παραστάσεις ύψηλης διακριτικότητας (320x200)

- Τρισδιάστατα έφφέ
- Επαγγελματικό συνθεσάιζερ μουσικής
- Μεγάλη γκάμα από περιφερειακά και interfaces

Επίσης καταπληκτική είναι και ή τιμή, πού είναι ή μισή από τήν αντίστοιχη τιμή του πιο κοντινού ανταγωνιστή.

# 64



**MEMOX**

**MEMOX A.E.**

Βασ. Σοφίας 82, Αθήνα  
Τηλ. 7778680, 7781912,  
7788711

**Μεγάλη ποικιλία  
έτοιμων προγραμμάτων**

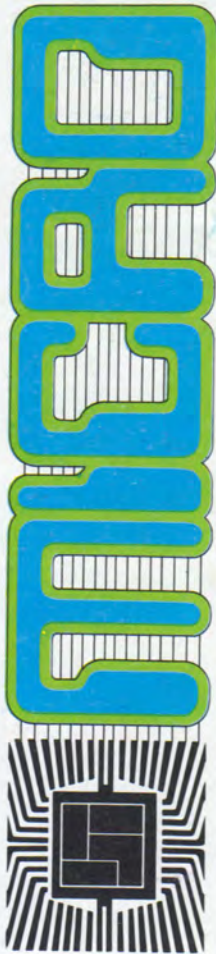
- Γιά επιχειρήσεις
- Στο σπίτι
- Γιά εκπαίδευση



**commodore**  
COMPUTER

00ΩΝΟΣ 99  
ΚΗΦΙΣΙΑ  
ΤΗΛ. 8085-587

...for personal computers



```
RETURN
160 FORZ=1T03:A$(D,E,Z)=A$(F,G,Z):NEXTZ:
RETURN
170 FORZ=1T03:A$(D,E,Z)=A$(F,G,Z):NEXTZ:
RETURN
180 FORZ=1T03:A$(D,E,Z)=A$(F,Z,G):NEXTZ:
RETURN
190 D=0:E=0:F=4:G=H:GOSUB150
200 D=4:E=H:F=1:GOSUB150
210 D=1:F=3:G=4-G:GOSUB150
220 D=3:E=G:F=6:GOSUB150
230 D=6:F=0:G=0:GOSUB150
240 IFH=1THENH1=5:GOSUB400ELSEIFH=3THENH
1=2:GOSUB400
250 RETURN
260 D=0:E=0:F=1:G=H:GOSUB160
270 D=1:E=G:F=2:GOSUB160
280 D=2:F=6:G=4-G:GOSUB160
290 D=6:E=G:F=5:GOSUB160
300 D=5:F=0:G=0:GOSUB160
310 IFH=1THENH1=4:GOSUB400ELSEIFH=3THENH
1=3:GOSUB400
320 RETURN
330 D=0:E=0:F=3:G=H:GOSUB160
340 D=3:E=G:F=2:GOSUB160
350 D=2:F=4:G=4-G:GOSUB170
360 D=4:E=G:F=5:GOSUB160
370 D=5:F=0:G=0:GOSUB170
380 IFH=1THENH1=1:GOSUB400ELSEIFH=3THENH
1=6:GOSUB400
390 RETURN
400 FORX=1T02:A$(0,0,X)=A$(H1,1,X):A$(H1
,1,X)=A$(H1,4-X,1)
410 A$(H1,4-X,1)=A$(H1,3,4-X):A$(H1,3,4-
X)=A$(H1,X,3)
420 A$(H1,X,3)=A$(0,0,X):NEXTX:RETURN
430 J=0:I=0:M=0
440 IFQA THENPRINT@448," SOLVED. HIT X T
O START AGAIN ";:ELSEPRINT@," ";
450 PRINTSTRING$(32,32);:PRINT@," ";:GOS
UB2200
460 IFQ$=CHR$(31)THEN430
470 IFQ$="X"THENI=1:PRINT"*";:GOTO450
480 IFQ$="R"THENJ=2:H=3-I
490 IFQ$="L"THENJ=2:H=1+I:M=1
500 IFQ$="B"THENJ=1:H=1+I
510 IFQ$="F"THENJ=1:H=3-I:M=1
520 IFQ$="T"THENJ=3:H=1+I:M=1
530 IFQ$="U"THENJ=3:H=3-I
540 IFQ$="X"THEN730
550 IFJ=0THEN450
560 PRINTQ$;
570 GOSUB2200
580 IFQ$=CHR$(31)THEN430
590 K=VAL(Q$):IFK<10RK>3THEN570
600 PRINTK;
610 IFM=1THENK=4-K
620 FORL=1TOK:ONJ GOSUB190,260,330:NEXTL
630 GOSUB140:GOTO430
640 PRINT"SHUFFLING..." :FORL=1T020:J=RND
(3):H=RND(3):ONJ GOSUB190,260,330
650 NEXTL
660 PRINT"COMPUTER SOLVE THE CUBE (Y/N)?
":GOSUB2200:PRINTQ$:IFQ$="Y"THENQA=1:GO
10900ELSE100
670 PRINT"B/W OR COLOR TV (B OR C)?":GO
SUB2200:PRINTQ$:PRINT"ONE MOMENT...":IFQ
$="B"THENQ$="O Y B G W R "ELSEQ$="MOTOR
MOTOR^^RESTORESTORE!!SCREENSCREEN"
680 FORI=1T06:AA$(I)=MID$(Q$,1*2-1,2):NE
XT
690 FORX=1T06:FORY=1T03:FORZ=1T03:A$(X,Y
,Z)=AA$(X):NEXTZ,Y,X
700 PRINT"SHUFFLE THE CUBE (Y/N)?":GOSU
B2200:PRINTQ$:IFQ$="Y"THEN640
710 PRINT"SET IT UP A CERTAIN WAY (Y/N)?
":GOSUB2200:PRINTQ$:IFQ$="Y"THEN740
720 QA=0:GOTO100
730 CLS(6):PRINT"START COMPLETELY OVER (
Y/N)?":GOSUB2200:PRINTQ$:IFQ$="Y"THENRU
NELSE660
740 GOSUB60
750 FORM1=1T06:FORM2=1T03:FORM3=1T03
760 PRINT@0,"FACE, COLUMN, ROW: ";M1;"",
M2;"",M3;
770 A$(M1,M2,M3)="?":GOSUB1240
780 PRINT@480,"ENTER COLOR (O,Y,B,G,W,R)
":GOSUB2200:X=INSTR("OYBGWR",Q$):IFX=0
THEN780ELSEPRINTQ$:A$(M1,M2,M3)=AA$(X):
GOSUB1240
790 NEXTM3,M2,M1
800 FORX=1T06:A2$(X)="123":NEXTX
810 M4=0:FORM1=1T06:FORM2=1T03:FORM3=1T0
3
820 FORM5=1T0M4:IFA$(M1,M2,M3)=A2$(M5)TH
ENM(M5)=M(M5)+1:IFM(M5)>9THEN850ELSE840
830 NEXTM5:M4=M4+1:IFM4>6THEN850ELSEM(M4
)=1:A2$(M4)=A$(M1,M2,M3)
840 NEXTM3,M2,M1:CLS(6):GOTO660
850 GOSUB60:PRINT@0,"ERROR! PLEASE MAKE
CORRECTIONS. ";:GOSUB140
860 PRINT@448,"ENTER FACE, COLUMN, ROW TO
CHANGE (TYPE 0,0,0 TO END) ";
870 GOSUB2200:M1=VAL(Q$):PRINTQ$;";":GO
SUB2200:M2=VAL(Q$):PRINTQ$;";":GOSUB220
0:M3=VAL(Q$):PRINTQ$;:IFM1*M2*M3=0THEN80
0
880 A$(M1,M2,M3)="?":GOSUB1240
890 PRINT@448,STRING$(32,32);"CORRECT CO
LOR (O,Y,B,G,W,R)?":GOSUB2200:X=INSTR("
OYBGWR",Q$):IFX=0THEN890ELSEPRINT@480,ST
RING$(28,32);:A$(M1,M2,M3)=AA$(X):GOSUB1
240:GOTO860
900 FORX=1T06:AA$(X)=A$(X,2,2):NEXTX:PRI
NT:PRINT"THE COMPUTER IS THINKING. ";
910 FORX=1T06:FORY=1T03:FORZ=1T03
920 FORQ=1T06:IFAA$(Q)=A$(X,Y,Z)THENQQ(X
,Y,Z)=Q:GOTO940
930 NEXTQ:GOTO850
940 NEXTZ,Y,X
950 Q$="11153341313143321113321331111333
1531633411513613231431611313233631511333
":FORB=0T07:FORC=0T02:X=(B*3+C)*3+1:Q3(0
,B+1,C+1)=QQ(VAL(MID$(Q$,X,1)),VAL(MID$(
Q$,X+2,1)),VAL(MID$(Q$,X+1,1))):NEXTC,B
960 FORX=1T08:B=Q3(0,X,1)+Q3(0,X,2)+Q3(0
,X,3)
970 Q3(0,X,0)=INSTR("2/.17436",CHR$(B+40
)):IFQ3(0,X,0)=0THEN850
980 NEXTX
990 FORX=1T08:FORY=1T03:IFQ3(0,X,Y)=10RQ
3(0,X,Y)=6THENQ3(0,X,1)=Y:GOTO1010
1000 NEXTY:GOTO850
1010 NEXTX
1020 Q$="1124321232211323121215235324212
1242323232151232363241262122361233262352
1":FORB=0T011:FORC=0T01:X=(B*2+C)*3+1:Q3
(1,B+1,C+1)=QQ(VAL(MID$(Q$,X,1)),VAL(MID
$(Q$,X+1,1)),VAL(MID$(Q$,X+2,1))):NEXTC,
B
1030 FORX=1T012:B=Q3(1,X,1)+Q3(1,X,2):C=
```

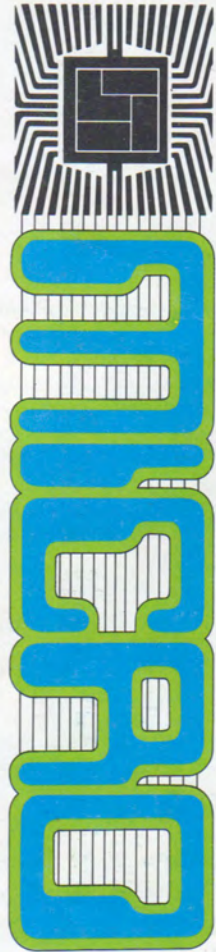




## Ο ΚΥΒΟΣ ΤΟΥ RUBIC

```
5 / *** RUBIC CUBE ***
20 GOTO2210
30 CLS(6):DIMA$(6,3,3),OO(6,3,3),O3(1,12
.3),O1(40)
40 F=1:X=1:BP=1:B=1:B1=1:B2=1:B3=1:RW=2:
CL=32:CD=40:GP=PEEK(188)*256:FOR1=1TO6:R
EADNA(I):NEXT:POKE279,PEEK(275):DATA175,
303,310,168,55,289
50 GOTO670
60 GLS(6):PRINT@66,"RUBIK'S CUBE";:PRINT
@99,"SIMULATION";:PRINT@322,"CLOCKWISE";
:PRINT@354,"ROTATION";
70 PRINT@120,"U L T R";:PRINT@92,"B";:PR
```

```
INT@156,"F";
80 PRINT@376,"6 4 1 3";:PRINT@348,"5";:P
RINT@412,"2";
90 RETURN
100 GOSUB60
110 PRINT@480,"ENTER YOUR MOVE X TO E
XIT";
120 QA=0
130 GOSUB140:GOTO430
140 FORM1=1TO6:FORM2=1TO3:FORM3=1TO3:GOS
UB1240:NEXTM3,M2,M1:RETURN
150 FORZ=1TO3:A$(D,Z,E)=A$(F,Z,G):NEXTZ:
```



...for personal computers

ΟΘΩΝΟΣ 99  
ΚΗΦΙΣΙΑ  
ΤΗΛ. 8085-587

## Ο ΚΥΒΟΣ ΤΟΥ RUBIC

```

03(1,X,1):IFQ3(1,X,2)<C THENC=Q3(1,X,2)
1040 IFC=1THENIFB=5THENQ3(1,X,0)=1:GOTO1
070ELSEIFB=6THENQ3(1,X,0)=4:GOTO1070
1050 IFC=4ANDB=9THENQ3(1,X,0)=5:GOTO1070
ELSEIFC=3ANDB=8THENQ3(1,X,0)=8:GOTO1070
1060 Q3(1,X,0)=INSTR(" +, - 2013",CHR$(
B+40)):IFQ3(1,X,0)=0THENB50
1070 NEXTX
1080 FORX=1TO12:FORX=1TO2:IFQ3(1,X,Y)=10
RQ3(1,X,Y)=6THENQ3(1,X,1)=Y:GOTO1110
1090 NEXTY:FORX=1TO2:IFQ3(1,X,Y)=5ORQ3(1
,X,Y)=2THENQ3(1,X,1)=Y:GOTO1110
1100 NEXTY:GOTO850
1110 NEXTX
1120 B=:FORX=1TO8:Q1(B)=Q3(0,X,0)*2-1:Q
1(B+1)=Q3(0,X,1):B=B+2:NEXTX
1130 FORX=1TO12:Q1(B)=Q3(1,X,0)*2+15:Q1(
B+1)=Q3(1,X,1):B=B+2:NEXTX:PRINT"*":GOT
O1310
1140 Q$="( -57/(.68Q(=EMG(>FNI(" :Q2$="000
000121200000000000":GOTO1200
1150 Q$="( )13+(X24,(9AIC(:BJJC":Q2$="000
000121200000000000":GOTO1200
1160 Q$="( )/71(X082(?G0A@HPB(" :Q2$="000
00021210000001111":GOTO1200
1170 Q$="( +35-(,46.( ;CKE(<DLF(" :Q2$="000
00012120000001111":GOTO1200
1180 Q$="( )+-(X,.0(9;=?(:<>@(" :Q2$="000
000000000000000000":GOTO1200
1190 Q$="(1753(2864(PNLJ(QMKI(" :Q2$="000
000000000000000000"
1200 FORX=1TO20:Q1(ASC(MID$(Q$,X,1))-CD)
=Q1(ASC(MID$(Q$,X+1,1))-CD)+VAL(MID$(Q2$,
,X,1)):NEXTX:FORX=2TO16STEP2:IFQ1(X)>3THE
NQ1(X)=Q1(X)-3

```

```

1210 NEXTX
1220 FORX=20TO40STEP2:IFQ1(X)=3THENQ1(X)
=1
1230 NEXTX:RETURN
1240 ONM1 GOTO1250,1250,1270,1250,1260,1
270
1250 PRINT@NA(M1)+RW*M2+CL*M3,A$(M1,M2,M
3):RETURN
1260 PRINT@NA(M1)-RW*M2+CL*M3,A$(M1,M2,M
3):RETURN
1270 PRINT@NA(M1)+RW*M2-CL*M3,A$(M1,M2,M
3):RETURN
1280 BP=BP+1:POKEGP+BP,B:B1=INT(B/3):B2=
B-B1*3+1
1290 FORB3=1TOB2:ONB1+1GOSUB1140,1150,11
60,1170,1180,1190,1300:NEXTB3
1300 RETURN
1310 BP=0
1320 C=19:GOSUB1450:C=21:GOSUB1450:C=23:
GOSUB1450:C=17:GOSUB1450:PRINT"*";
1330 GOSUB1610:F=3:GOSUB1740:PRINT"*";:G
OSUB1770:PRINT"*";:GOSUB1940:GOSUB1960:P
RINT"*";:GOSUB2000:GOSUB2060
1340 IFQ1(9)>9THENB=15:GOSUB1280:GOTO13
40
1350 PRINT"*";:GOSUB1390
1360 FORX=1TO39STEP2:IFQ1(X)<>X ANDQ1(X+
1)<>1THENPRINT"ERROR";:ELSENEXTX
1370 PRINT:INPUT"COMPUTER IS READY, PUSH
ENTER";A1
1380 GOSUB1510:GOTO430
1390 FORX=2TOBP:BL=PEEK(GP+X):BB=PEEK(GP
+X-1):B=INT(BL/3)*3:C=BL-B:D=INT(BB/3)*3
:E=BB-D:IFB=D THENBB=18:BL=BL+E+1:GOTO14
10

```

Ένα πολύ χρήσιμο παιχνίδι, για όποιον προσπάθησε κάποτε να λύσει τον μαγικό αυτό κύβο χωρίς επιτυχία. Το πρόγραμμα μπορεί να λύσει οποιοδήποτε πρόβλημα του δώσετε σε λιγότερο από πέντε λεπτά. Έπειτα θα σας δείξει μια προς μια τις κινήσεις που πρέπει να κάνετε. Επίσης μπορεί ο ίδιος ο κομπιούτερ να "μπερδέψει" τον κύβο και να ζητήσει από σας να τον λύσετε, δίνοντάς σας και την λύση αν δεν τα καταφέρετε.

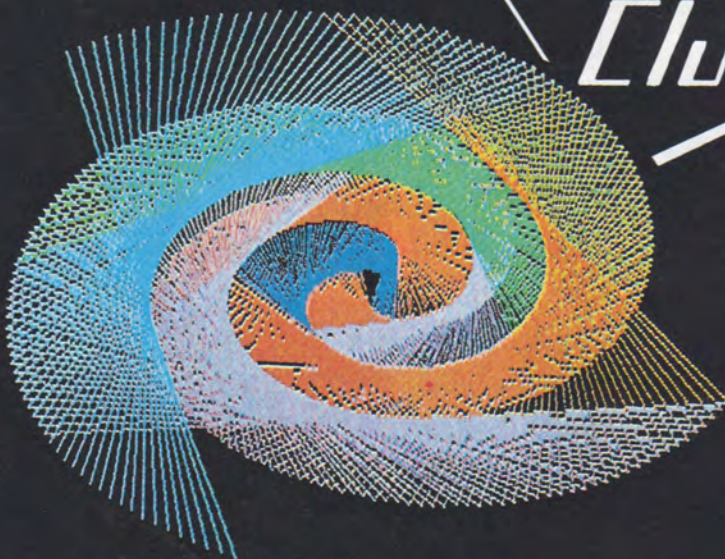
ΕΜ. ΜΠΕΝΑΚΗ & ΚΩΛΕΤΤΗ 15

# Computer Club

ΤΗΛ.: 3637442

για όλους...

ΠΩΛΗΣΕΙΣ ΜΕ ΕΥΚΟΛΙΕΣ  
ΔΑΝΕΙΣΤΙΚΗ ΒΙΒΛΙΟΘΗΚΗ  
ΕΝΟΙΚΙΑΣΕΙΣ  
ΔΩΡΕΑΝ ΜΑΘΗΜΑΤΑ  
ΠΡΟΓΡΑΜΜΑΤΑ  
ΕΥΧΑΡΙΣΤΟΣ ΦΙΛΙΚΟΣ ΧΩΡΟΣ



**EPSON** HX20  
PORTABLE COMPUTER

**ZX SPECTRUM**

# LASER

COLOR COMPUTER 200

## το φθηνότερο έγχρωμο micro



### CPU :

Z80A running at 3.58MHz

### MEMORY :

ROM — 16K Bytes Microsoft\* BASIC Interpreter  
RAM — 4K Bytes Expandable to 64K Bytes

### VIDEO DISPLAY :

Text mode — 32 columns x 16 rows  
Graphics — 128 x 64 dots (8 colors)  
— 64 x 32 dots (9 colors)  
Mix text & graphics — 32 columns x 16 rows  
— 64 x 32 dots (9 colors)  
9 colors — Black, Green, Yellow, Blue, Red,  
Buff, Cyan, Magenta & Orange.  
8 colors — Green, Yellow, Blue, Red, Buff,  
Cyan, Magenta & Orange.

### KEYBOARD :

45 key full size moving rubber keyboard  
Single key entry. - Automatic repeat key  
'BEEP' for key entry

### EDITING :

Full on-screen editing, Insert & rubout etc.

**SOUND :** Single channel sound output

### CASSETTE INTERFACE :

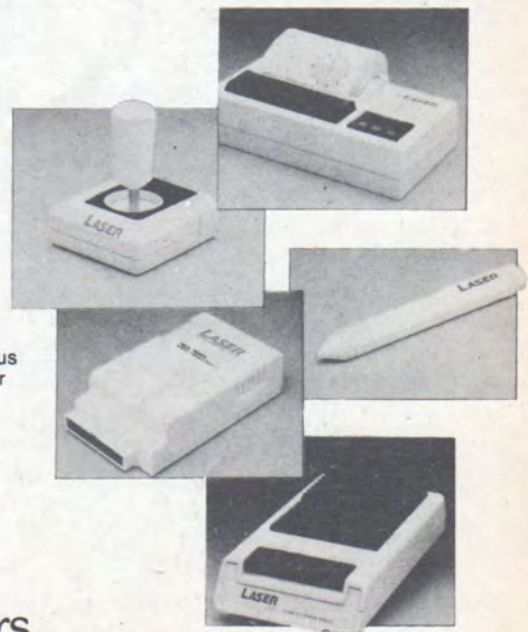
Baud rate at 600 bps  
Connect to ordinary audio cassette recorder or  
LASER DR10 Data cassette recorder  
Interface cord included

### VIDEO OUTPUT :

RF output to TV antenna cord included  
Video output to monitor

### OPTIONAL EXPANSIONS & PERIPHERALS :

16 K Bytes Memory Expansion Module  
64K Bytes Memory Expansion Module  
Printer Interface — connect to Centronics® Bus  
printer or LASER PP40 4 color printer/plotter  
Joysticks — for playing games  
Light-Pen—drawing on screen  
Expansion Module—facilitate more peripherals  
to plug in  
Floppy disk drive—mass storage  
Modem—communication over telephone line.



# micropoint computers

Nikos Loukidis • 207 Grevenon st., Nikea, Piraeus 18453 • Tel. 4902473 • Telex 221021 ALFA GR • 223193 EUGE GR

# Ο ΚΥΒΟΣ ΤΟΥ RUBIC

```

1400 POKEGP+X,BL:POKEGP+X-1,BB:NEXTX:RET
URN
1410 IFC+E=2THENBL=18
1420 IFC+E>2THENBL=BL-4
1430 IFBL=19THENBL=18
1440 GOTO1400
1450 D=18:FORJ=1TO8:IFQ1(ASC(MID$( "E9;=?
CGA",1))-40)=C THENB=ASC(MID$( "1,2)/30."
,1))-40:D=ASC(MID$( "::::.0",1))-40:GOT
O1470
1460 NEXT:GOTO1480
1470 GOSUB1280
1480 IFQ1(35)=C THEN1490ELSEB=15:GOSUB12
80:GOTO1480
1490 B=D:GOSUB1280:IFQ1(36)=1THENQ1$="24
":GOTO2190
1500 Q1$="14-":GOTO2190
1510 B1=1:J4=0:GOSUB60
1520 FORJ1=1TOBP:J2=PEEK(GP+J1):J3=INT(J
2/3):J2=J2-J3*3+1:IFJ3=6THEN1600
1530 PRINTQ0,"MOVE ";J2;" NUMBER OF MO
UES";B1;:B1=B1+1
1540 J=VAL(MID$( "221133",J3+1,1)):H=VAL(
MID$( "311313",J3+1,1))
1550 IFJ3=10RJ3=30RJ3=4THENJ2=4-J2
1560 PRINTQ5,MID$( "RLBFTU",J3+1,1):FORL
=1TOJ2:ONJ GOSUB190,260,330:NEXTL
1570 GOSUB140
1580 IFQ$="C"THENIFINKEY$=""THENPRINTQ44
9,STRING$(33,32);"HIT ANY KEY TO SINGLE
STEP ";:GOTO1600
1590 PRINTQ448," HIT ANY KEY TO SEE NEXT
MOVE HIT C FOR CONTINUOUS DISPLAY "
:POKE1535,143:GOSUB2200:IFQ$="C"THEN158
0
1600 NEXTJ1:RETURN
1610 C=5:FORX=1TO4:C1(X)=0:NEXTX:FORX=1T
O7STEP2:IFQ1(X)=X ANDQ1(X+1)=1THENC1((X+
1)/2)=1
1620 NEXTX
1630 IFC1(1)=1ANDC1(2)=1ANDC1(3)=1ANDC1(
4)=1THENRETURN
1640 FORF=1TO4
1650 IFQ1(14)=3AND(Q1(13)=10RQ1(13)=30RQ
1(13)=50RQ1(13)=7)THENGOSUB1710:Q1$="173
":GOSUB2190:GOTO1630
1660 IFQ1(14)=2AND(Q1(13)=10RQ1(13)=30RQ
1(13)=50RQ1(13)=7)THENGOSUB1710:Q1$="*9(
":GOSUB2190:GOTO1630
1670 IFQ1(14)=1AND(Q1(13)=10RQ1(13)=30RQ
1(13)=50RQ1(13)=7)THENGOSUB1710:Q1$="193
*8(" :GOSUB2190:GOTO1630
1680 B=15:GOSUB1280
1690 NEXTF:FORF=1TO4:IFC1(F)=0THENGOSUB1
740:Q1$="*7(" :GOSUB2190:GOTO1640
1700 NEXTF
1710 IFC=Q1(13)THENC1((C+1)/2)=1:RETURN
1720 B=12:GOSUB1280:C=C+2:IFC=9THENC=1
1730 GOTO1710
1740 IFC=F*2-1THENRETURN
1750 B=12:GOSUB1280:C=C+2:IFC=9THENC=1
1760 GOTO1740
1770 FORF=1TO4:C1(F)=0:NEXTF
1780 FORF=1TO4:IFQ1(F*2+23)=F*2+23ANDQ1(
F*2+24)=1THENC1(F)=1
1790 NEXTF
1800 FORF=1TO4:IFC1(F)=0THEN1810ELSENEXT
F:RETURN
1810 FORF=1TO4:IFQ1(37)=27ANDQ1(38)=2THE
NQ1$="+9-9371" :GOSUB2190:C1(2)=1:GOTO180
0

```

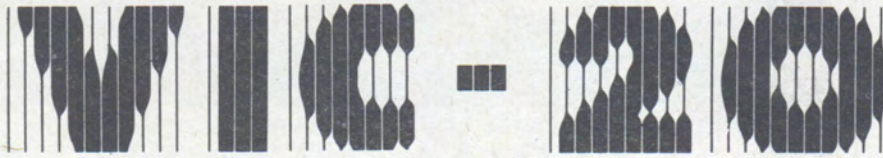
```

1820 IFQ1(39)=27ANDQ1(40)=1THENGOSUB2130
:C1(2)=1:GOTO1800
1830 IFQ1(39)=29ANDQ1(40)=1THENQ1$="1939
*7(" :GOSUB2190:C1(3)=1:GOTO1800
1840 IFQ1(33)=29ANDQ1(34)=2THENGOSUB2140
:C1(3)=1:GOTO1800
1850 IFQ1(33)=31ANDQ1(34)=2THENQ1$="(9*9
07." :GOSUB2190:C1(4)=1:GOTO1800
1860 IFQ1(35)=31ANDQ1(36)=1THENGOSUB2150
:C1(4)=1:GOTO1800
1870 IFQ1(35)=25ANDQ1(36)=1THENQ1$=" .909
-7+" :GOSUB2190:C1(1)=1:GOTO1800
1880 IFQ1(37)=25ANDQ1(38)=2THENGOSUB2120
:C1(1)=1:GOTO1800
1890 B=15:GOSUB1280:NEXTF
1900 FORF=1TO4:IFC1(F)=0THEN1920
1910 NEXTF
1920 ONFGOSUB2120,2130,2140,2150
1930 GOTO1810
1940 FORF=1TO4:IFQ1(13)-2=Q1(15)ORQ1(13)
+6=Q1(15)THENQ1$="-3+0-1+." :GOSUB2190
1950 B=15:GOSUB1280:NEXTF:RETURN
1960 FORF=10TO14STEP2:IFQ1(F)<>1THEN1970
:ELSENEXTF:RETURN
1970 FORF=2103:IF(Q1(10)=F)+(Q1(12)=F)+
(Q1(14)=F)+(Q1(16)=F))=-3THEN2030:ELSENE
XTF
1980 FORF=1104:IF((Q1(10)=2)+(Q1(12)=2)+
(Q1(14)=1)+(Q1(16)=2))=-1THENQ1$="+7-7+8
-":GOSUB2190:GOTO2030
1990 B=15:GOSUB1280:NEXTF
2000 IFQ1(40)=2ANDQ1(36)=2ANDQ1(38)=1THE
NGOTO2180
2010 IFQ1(40)=1ANDQ1(36)=1ANDQ1(38)=2THE
NB=15:GOSUB1280:GOTO2180
2020 RETURN
2030 FORF=1TO4:IFQ1(12)=2ANDQ1(14)=1THEN
Q1$="+7-7+8-":GOTO2190
2040 IFQ1(12)=3ANDQ1(10)=1THENQ1$="+8-9+
9-":GOTO2190
2050 B=15:GOSUB1280:NEXTF:STOP
2060 FORF=1TO4:IFQ1(34)=2ANDQ1(40)=21HEN
2160:ELSEB=15:GOSUB1280:NEXTF
2070 IFQ1(35)-Q1(15)=24ANDQ1(39)-Q1(11)=
24THENQ1$=")",7/25/27," :GOSUB2190:NEXT:R
ETURN
2080 FORF=1TO2:IFQ1(37)-Q1(15)=24ANDQ1(3
9)-Q1(13)=24THENQ1$="/24)/2,/26/2":GOSUB
2190:NEXT:RETURN
2090 B=15:GOSUB1280:NEXTF
2100 FORF=1TO4:IF(Q1(33)-Q1(15)=24)AND(Q
1(37)-Q1(13)<>24)THEN2180:ELSEB=15:GOSUB
1280:NEXTF
2110 RETURN
2120 Q1$="-7+7,90":GOTO2190
2130 Q1$="3717+9-":GOTO2190
2140 Q1$="*7(7193":GOTO2190
2150 Q1$="07.7(9*":GOTO2190
2160 IFQ1(39)-Q1(9)=24ORQ1(33)-Q1(11)=24
OR(Q1(35)-Q1(15)=24ANDQ1(36)=1)THENQ1$="
3(108.3(10":GOSUB2190:GOTO2060
2170 FORI1=1TO11:B=ASC(MID$( "(-3*+8(-3*+
",11))-40:GOSUB1280:NEXT:GOTO2060
2180 Q1$="),6*+2(-6),":IFQ1(37)-Q1(11)=2
4THENQ1$="),4*+2(-4),":
2190 FORI1=1TOLEN(Q1$):B=ASC(MID$(Q1$,11
))-CD:GOSUB1280:NEXT:RETURN
2200 Q$=INKEY$:IFQ$=""ORQ$<"0"THEN2200EL
SERETURN
2210 PMODE0:PCLEAR1:GOTO30

```

Το πρόγραμμα αυτό μπορεί να δουλέψει το ίδιο καλά σε μαυρό-ασπρη ή έγχρωμη τηλεόραση. Στην αρχή του προγράμματος, ο κομπιούτερ σας ρωτάει τι τηλεόραση έχετε. Αν του απαντήσετε ότι έχετε μαυρόασπρη τότε τα χρώματα των πλευρών του κύβου συμβολίζονται με γράμματα: W για το άσπρο, G για το πράσινο, R για το κόκκινο, B για το μπλέ, Y για το κίτρινο και O για το πορτοκαλί. Επίσης αν κάνετε κάποιο λάθος όταν δίνετε στον κομπιούτερ ένα πρόβλημα, αυτός θα το εντοπίσει και θα σας ζητήσει να το διορθώσετε. Το πρόγραμμα απαιτεί Extended Basic και τουλάχιστον 16K μνήμης RAM. Οι κάτοχοι κομπιούτερ 16K RAM πρέπει πριν "φορτώσουν" το πρόγραμμα να πληκτρολογήσουν τις εξής εντολές: PMODE0:PCLEAR1 (enter) για να έχουν αρκετή μνήμη στη διάθεσή τους.

# Προγράμματα για τόν:



ΕΠΙΜΕΛΕΙΑ:  
Μανώλης Παπαθεοφάνους

Έχετε την τελευταία βάση στη γη και προσπαθείτε να καταστρέψετε τον εισβολέα. Μπορείτε, μετακινώντας το πυροβόλο, να πυροβολήσετε προς τρεις διαφορετικές κατευθύνσεις. Το κοντρόλ σας είναι:

Βάση: "Z" (αριστερά), "X" (δεξιά), Πυροβόλο, "," (αριστερά), "." (δεξιά)

Μπορείτε να πυροβολήσετε με τη μπάρα του διαστήματος. Το παιχνίδι τρέχει στον unexpanded Vic.



## ALPHA ATTACK

```

5 A$(0)=" " A$(1)=" " A$(2)=" " A$(3)=" " A(0)=23:A(1)=22:A(2)=21:Y=5:P
=0
4 GOTO1000
5 R$="          " S$="          "
6 C(0)=77:C(1)=93:C(2)=78:PRINT" "
9 FORA=1TO50:POKE7680+(RND(TI)*396),46:NEXT
10 FORD=1TO10
20 PRINTS$:"          "
40 N=RND(TI)*20+7760:GOTO170
50 POKEN,32:POKEN+1,32:POKEN+2,32:N=N+RND(TI)*2-1+(INT(RND(TI)*2)*22)
60 POKEN,60:POKEN+1,22:POKEN+2,62
70 IFPEEK(N+22) <> 32 AND PEEK(N+22) <> 46 THEN 280
30 T=TI
30 POKE197,255:GETA$:IFA$="" AND T+V>T I THEN 90
100 IFA$="" THEN 50
110 IFA$="X" AND Y<18 THEN Y=Y+1:P=0:GOTO170
120 IFA$="Z" AND Y>5 THEN Y=Y-1:P=2:GOTO170
125 GETA$
130 IFA$="," AND P>0 THEN P=P-1:GOTO170
140 IFA$="." AND P<2 THEN P=P+1:GOTO170
150 IFA$<>" " THEN 50
160 GOTO180
170 PRINTS$;SPC(Y);R$+A$(P)
175 IFPEEK(8115+Y) <> 32 OR PEEK(8123+Y) <> 32 OR PEEK(8093+Y) <> 32 OR PEEK(8101+Y) <> 32 THE
N320
177 GOTO50
180 O=8096+Y+P
185 X=X-1:IFX<0 THEN 170
190 POKE36878,10:FORF=8096+Y+PTO7720STEP-A(P):POKEF,C(P):POKEO,32:O=F
195 POKE36876,(F-7680)/2
200 Z=PEEK(F-A(P)):IFZ <> 32 AND Z <> 46 THEN POKEO,32:POKE36878,0:GOTO240
210 IF(F-7680)/22 <> INT((F-7680)/22) THEN NEXT
220 POKEO,32
225 POKE36878,0
230 GOTO170
240 POKEN,32:POKEN+1,32:POKEN+2,32:POKE36877,220:FORL=15TO0STEP-1:POKE36878,L
250 FORM=1TO100:NEXT
260 NEXT:POKE36877,0:POKE36878,0
265 I=I+1
270 NEXTI
275 GOTO2000
280 U=0:R=PEEK(N)
285 IFR=119ORR=77ORR=93ORR=78 THEN 320
287 IFU=0 THEN U=1:R=PEEK(N+22):GOTO285
290 POKE36878,15:FORL=30TO35:FORM=220-LTO160-LSTEP-4:POKE36876,M:NEXTM
300 FORM=160-LTO220-LSTEP4:POKE36876,M:NEXTM,L:POKE36878,0:POKE36876,0
310 GOTO270

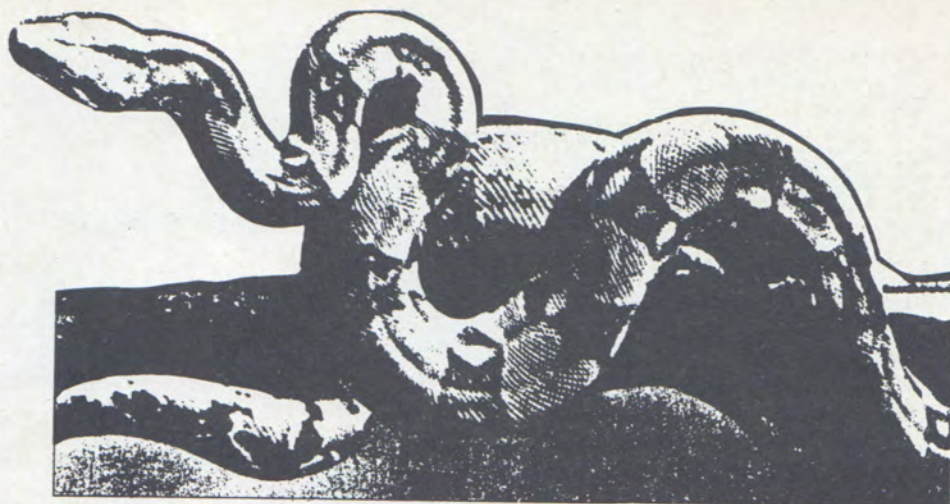
```



# ΤΟ ΦΙΔΙ

Ένα πολύ ωραίο παιχνίδι, που χρησιμοποιεί πολύ έξυπνα τις δυνατότητες του Vic. Αν και γραμμένο σε BASIC, είναι πολύ γρήγορο. Το πρόγραμμα είναι για τον unexpanded Vic και χρησιμοποιεί σχεδόν όλη τη μνήμη του.

Σκοπός του παιχνιδιού είναι να καθοδηγήσετε ένα φίδι έτσι ώστε να τρώει τις μύγες και τα σκαθάρια που υπάρχουν στην οθόνη και να αποφεύγει τα δηλητηριώδημανιτάρια. Και οι μύγες και τα σκαθάρια μεγαλώνουν το μέγεθος του φιδιού, αλλά τα σκαθάρια περισσότερο.



Όσο πιο πολλά έντομα τρώτε, τόσο δυσκολότερο γίνεται το παιχνίδι. Αν το φίδι ακουμπήσει τον εαυτό του, πεθαίνει. Αν προσπαθήσει να χτυπήσει σ'ένα τοίχο, θα σταμα-

τήσει και το σκορ θ' αρχίσει να κατεβαίνει. Όταν το σκορ κατέβει κάτω από το μηδέν, το φίδι πεθαίνει.

Αν τελειώνοντας έχετε σκορ ανάμεσα

στους πέντε πρώτους, το πρόγραμμα σας δίνει την ευκαιρία να τυπώσετε τα αρχικά σας. Τα κοντρόλ του παιχνιδιού είναι:

S·(επάνω), X·(κάτω),  
(δεξιά), (αριστερά).

```
10 REM FOR "■" READ INVERSE POUND SIGN (CNTRL+KEY3)
20 DIMB$(5):FORI=1TO5:B$(I)="000000 ???":NEXT
40 DEFFNR(X)=INT(RND(1)*X)
50 POKE56,29:POKE52,29
60 S=42:M=39:B=40:F=41:W=171
70 TC=7680:CO=30720:S1=36874:S2=S1+2:S3=S2+1:V=S2+2:SR=V+1
80 GOSUB3000
100 NM=20:GOSUB1500:POKEV,47
120 HP=TC+10+22*2:L=5:TP=HP+L:D=22:H=46
140 POKEHP,H:POKEHP+CO,1:FORI=HP+1TOTP:POKEI,S:POKEI+CO,13:NEXT
150 SC=0:GOSUB900
160 PRINT"■■■■"TAB(16)LEFT$(B$(1),6);
170 T=TI
180 GETA$:IFA$=""ANDTI-T<300THEN180
190 GOTO210
200 IFSC<0THEN2000
205 GETA$
210 Z=22*(A$="S")-22*(A$="X")+(A$=",")-(A$="."):D=-D*(Z=0)+Z
250 H=-43*(D=1)-44*(D=-1)-45*(D=-22)-46*(D=22)
270 POKEHP,H:NP=HP+D:P=PEEK(NP)
280 IFP=MORP=STHEN2000
290 IFP=WORPEEK(NP+1)=SORPEEK(NP-1)=SORPEEK(NP+22)=SORPEEK(NP-22)=STHENGOSUB800
GOTO200
300 IFP=BTHENGOSUB700
310 IFP=FTHENGOSUB750
320 IFNB=0ANDNF=0THENNM=10:GOSUB1560:GOTO170
330 GOSUB500:GOTO205
500 POKES1,170
510 POKEHP+CO,13:POKEHP,S:HP=NP:POKEHP+CO,1:POKEHP,H
520 IFEL>0THENEL=EL-1:GOTO550
530 POKETP,32
540 TP=TP-(PEEK(TP+1)=S)+(PEEK(TP-1)=S)+22*(PEEK(TP-22)=S)-22*(PEEK(TP+22)=S)
550 POKES1,0
560 FORT=1TO150-4*L:NEXT
570 RETURN
```



```

700 POKESR,12:L=L+1:EL=EL+1
710 X=INT((100+5*L)/10)
720 FORI=1TO10:POKES2,200:POKENP,I+128
730 SC=SC+X:GOSUB900:POKES2,0:NEXT:POKESR,10:NB=NB-1:POKENP,32
740 RETURN
750 POKESR,14:L=L+2:EL=EL+2

760 X=INT((150+6*L)/10)
770 FORI=1TO10:POKES2,220:POKENP,I+128
780 SC=SC+X:GOSUB900:POKES2,0:NEXT:POKESR,10:NF=NF-1:POKENP,32
790 RETURN
800 POKES2,180
810 SC=SC-10:GOSUB900
820 POKES2,0
830 RETURN
900 S$=STR$(SC):Q=LEN(S$):S$=LEFT$("0000",7-Q)+RIGHT$(S$,Q-1)
910 PRINT" SCORE: "S$
920 RETURN
1000 FORC=35TO46:FORZ=0TO7:READN:POKE7168+8*C+Z,N:NEXT:NEXT
1010 FORZ=8*32TO8*32+7:POKE7168+Z,0:NEXT
1020 RETURN
1100 DATA255,171,171,139,171,171,171,255,255,136,187,139,235,235,136,255
1120 DATA255,136,170,168,170,170,138,255,255,143,191,159,191,191,143,255
1140 DATA56,124,124,254,186,56,56,124,130,84,56,254,56,254,56,254
1160 DATA108,16,124,186,186,186,84,16,40,40,190,190,190,40,40
1180 DATA48,104,109,254,254,109,104,48,12,22,182,127,127,182,22,12
1200 DATA36,24,60,126,153,255,126,24,24,126,255,153,126,60,24,36
1500 POKE36869,255
1510 POKESR,10:PRINT" SCORE: "TAB(11)" SCORE: "
1520 FORI=0TO21:P1=TC+22+I:P2=TC+22*22+I:P3=TC+22*(1+I):P4=TC+21+22*(1+I)
1530 POKEP1,W:POKEP2,W:POKEP3,W:POKEP4,W:POKEP1+CO,5:POKEP2+CO,5:POKEP3+CO,5:PO
KEP4+CO,5
1550 NEXT
1560 CH=M:C=4:N=NM:GOSUB1700
1570 CH=B:C=5:NB=15:N=NB:GOSUB1700
1580 CH=F:C=7:NF=5:N=NF:GOSUB1700
1590 RETURN
1700 FORI=1TON
1710 P=TC+FNR(20)+1+22*(FNR(19)+3)
1720 IFPEEK(P)<>32THEN1710
1730 POKEP+CO,C:POKEP,CH
1740 NEXT
1750 RETURN
2000 POKES3,200:FORT=1TO250:POKESR,2:FORI=1TO10:NEXT:POKESR,10:NEXT
2050 FORVO=15TO0STEP-.1:POKEV,VO:NEXT
2060 POKES3,0:POKESR,12:POKE36869,240:POKE198,0
2070 PRINT" YOU WERE KILLED BY A DEADLY":PRINT" POISONED MUSHROOM "
2080 K=0:FORI=1TO5:IFSC>VAL(LEFT$(B$(I),6))THENK=I:I=5
2090 NEXT:IFK=0THENFORT=1TO1000:NEXT:GOTO2150
2100 PRINT" YOU ARE ONE OF THE TOP FIVE PLAYERS"
2110 PRINT" ENTER YOUR INITIALS":INPUTI$:IFLEN(I$)>3THEN2110
2120 FORI=4TOKSTEP-1:B$(I+1)=B$(I):NEXT
2130 B$(K)=S$+" "+I$
2150 POKESR,78:PRINT" HALL OF FAME"
2160 FORI=1TO5:PRINT" I "B$(I):NEXT
2170 PRINT" ANOTHER GO(Y/N)":INPUTY$
2180 IFY$="Y"THEN100
2190 IFY$="N"THENPRINT" THANKS FOR PLAYING":END
2200 GOTO2170
3000 POKESR,154:PRINT" SNAKE -- B Y T E "
3020 PRINT" USE THESE KEYS:"
3030 PRINT" UP DOWN LEFT RIGHT X DOWN"
3040 PRINT" EAT THE FLIES & BEETLES , BUT BEWARE"
3050 PRINT" THE DEADLY MUSHROOMS"
3060 GOSUB1000
3070 PRINT" HIT ANY KEY TO START";
3080 GETA$:IFA$=""THEN3080
3090 RETURN

```



```

175 POKES1,180
180 H=100-VAL(TI$)
185 IFD=>5THEN300
190 IFSC=>20THEN325
195 POKES1,170:IFTI$=>"000200"THEN295
200 POKEPO+T,F:POKEPO+30720+T,2:FORA=1TOH:NEXTA
205 IFPEEK(PO+T+22)=247THENPOKES1,200:GOTO260
210 POKEPO+T,11
215 IFPO=>8076THEN250
220 GOSUB230:POKES1,0:NEXTT
225 PP=22:PO=PO+T:P=110:GOTO170
230 PRINT"SC"TI$;"SC"SC;"D"D"
235 IFPEEK(197)=31ANDM=>1THENM=M-1:GOTO245
240 IFPEEK(197)=23ANDM=<15THENM=M+1
245 PRINTTAB(M)M$:RETURN
250 FORB=1TO20:POKES1,190:FORA=1TO20:NEXTA:POKES1,150:FORA=1TO20:
NEXTA,B:D=D+1
255 POKES1,0:GOTO130
260 IFF1=4THENMA$="KCDIK" GOTO270
265 MA$="KCHIK"
270 POKES1,230:GOSUB230:POKES1,0:IFM=3THEN285
275 PRINTTAB(M)MA$
280 GOTO260
285 POKE36877,180:PRINTL2$:FORA=1TO200:NEXTA:PRINTSF$:FORA=1TO300:NEXTA:SC=SC+1
290 POKE36877,0:GOTO130
295 POKE36869,240:POKES1+3,29:PRINT"TIME'S UP!":POKES1+2,0:GOTO310
300 POKE36869,240:PRINT"YOU DROPPED 5 THINGS!"
305 POKES1,0
310 PRINT" BUT YOU SCORED"
315 PRINT"SC"POINTS,"
320 PRINT" WELL DONE!":FORA=1TO5000:NEXTA:RUN
325 PRINT"TTTT"SPACE =FIRE          CRSR DOWN =LEFT          CRSR RIGHT=RIGHT"
POKES1,0
330 FORA=0TO505:POKE7680+A,11:NEXTA:PRINT":POKES1+3,8:POKES1+1,240
335 FORA=1TO2
340 PRINT"
345 PRINT"
350 NEXTA
355 BASE$=" | ^ " T=100
360 PRINT" ":M=7
365 POKE198,0:T=T-1:POKES1+2,4
370 IFPEEK(197)=31ANDM=>1THENM=M-1:GOTO385
375 IFPEEK(197)=32THENGOSUB405
380 IFPEEK(197)=23ANDM=<15THENM=M+1
385 PRINTTAB(M)BASE$
390 PRINT" "T"
395 IFT=<0THEN295
400 GOTO365
405 FORSP=440TO0STEP-22
410 POKE7681+SP+M,7
415 IFPEEK(7681+SP+M-22)=9THENSC=SC+10:POKES1+2,15:POKES1,180:FORA=1TO20:NEXTA
420 POKE7681+SP+M,11:POKES1,0
425 NEXTSP:T=T-5:RETURN
430 PRINT"STRIKER'S REVENGE"
435 PRINT"BY S.TWEED"
440 PRINT"CRSR DOWN =LEFTCRSR RIGHT =RIGHT"
445 PRINT"CATCH THE SPANNERS ANDTHE APPLES, AND CARRY THEM TO THE WAITING L
ORRY."
450 PRINT"YOU HAVE 2 MINUTES TO CATCH THEM."
455 PRINT"IF YOU CATCH 20 THEN YOU CAN MOVE ON TO THENEXT SCREEN"
460 POKE198,0
465 PRINT"PRESS SPACE TO PLAY"
470 GETA$:IFA$=" "THENRETURN
475 GOTO470

```

# ΤΡΙΣΔΙΑΣΤΑΤΟΣ ΛΑΒΥΡΙΝΘΟΣ

Συνήθως όλα τα προγράμματα αυτού του τύπου τρέχουν στον Vic 20 με επέκταση μνήμης. Αυτό είναι ένα πρόγραμμα για τον unexpanded Vic, με τρισδιάστατη απεικόνιση του λαβύρινθου.

Όταν τρέξετε το πρόγραμμα, θα φανερωθεί το σχέδιο του λαβύρινθου και ένα πράσινο αστεράκι στην κάτω αριστερή γωνία. Αυτό δείχνει τη θέση σας μέσα στον λαβύρινθο. Ο σκοπός του παιχ-

νιδιού είναι να βρείτε την έξοδο στην πάνω δεξιά γωνία. Πατήστε το "\*" και θα βρεθείτε στον πρώτο διάδρομο.

Στο κάτω μέρος της οθόνης, εμφανίζεται ένα μήνυμα, που γράφει

προς ποια κατεύθυνση ταξιδεύετε. Οι κινήσεις γίνονται πατώντας ένα πλήκτρο κάθε φορά, που συμβολίζει και από ένα σημείο του ορίζοντα (Βορράς: N, Ανατολή: E, Νότος: S, Δύση: W).

```

10 REM 3-D MAZE*****
20 REM FOR VIC-20****
30 REM BY J.HURLL****
100 O=7690:T=39400:L=419:HC=0:MC=0:M$="N":CN=0:LM=0:D$="NORTH"
105 POKE36879,31:POKE36878,15:GOSUB5200
115 GOSUB4200
120 GOSUB999
125 GETX$:IFX$<"*"THEN125
130 POKE36866,0:GOSUB999
140 GOSUB1510
150 GOSUB3050
170 GOSUB2000:GOSUB4000:LM=0
175 PRINT"*****YOU ARE FACING ";D$
180 POKE36866,150
190 M$=M$
200 POKE198,0
210 GETM$:IFM$=""THEN210
215 IFM$="H"THENHC=HC+1:GOTO120
218 GOSUB5200
220 POKE36866,0:GOSUB999
230 GOSUB3000:IFLM=1THENM$=M$:GOSUB5220:GOSUB5100:GOTO170
240 MC=MC+1:LM=0
250 IFM$="N"THENL=L-22:D$="NORTH"
260 IFM$="E"THENL=L+1:D$="EAST"
270 IFM$="S"THENL=L+22:D$="SOUTH"
290 IFM$="W"THENL=L-1:D$="WEST"
295 IFL=43THENPOKE36866,150:GOTO4100
298 GOTO140
999 GOSUB5000:POKEO+L,42:POKET+L,5:RETURN
1510 IFM$="N"THENA=O+L-1:B=O+L-22:C=O+L+1
1520 IFM$="E"THENA=O+L-22:B=O+L+1:C=O+L+22
1530 IFM$="W"THENA=O+L+22:B=O+L-1:C=O+L-22
1540 IFM$="S"THENA=O+L+1:B=O+L+22:C=O+L-1
1570 RETURN
2000 PRINT"
2005 PRINT"
2010 PRINT"
2015 PRINT"
2020 PRINT"
2025 PRINT"
2030 PRINT"
2035 FORI=1TO6:PRINT"
2036 PRINT"
2037 PRINT"
2040 PRINT"
2045 PRINT"
2050 PRINT"
2055 PRINT"
2060 PRINT"
2070 RETURN
3000 IFM$="N"ANDPEEK(O+L-22)=160THENLM=1
3010 IFM$="E"ANDPEEK(O+L+1)=160THENLM=1
3020 IFM$="W"ANDPEEK(O+L-1)=160THENLM=1
3030 IFM$="S"ANDPEEK(O+L+22)=160THENLM=1
3040 RETURN
3050 IFPEEK(A)=32ANDPEEK(B)=160ANDPEEK(C)=160THENCN=1
3060 IFPEEK(A)=160ANDPEEK(B)=160ANDPEEK(C)=32THENCN=2
3065 IFPEEK(A)=160ANDPEEK(B)=160ANDPEEK(C)=160THENCN=3
3070 IFPEEK(A)=32ANDPEEK(B)=32ANDPEEK(C)=160THENCN=4
3075 IFPEEK(A)=160ANDPEEK(B)=32ANDPEEK(C)=32THENCN=5
3080 IFPEEK(A)=32ANDPEEK(B)=32ANDPEEK(C)=32THENCN=6
3085 IFPEEK(A)=32ANDPEEK(B)=160ANDPEEK(C)=32THENCN=7
3090 IFPEEK(A)=160ANDPEEK(B)=32ANDPEEK(C)=160THENCN=8
3095 RETURN
3100 PRINT"*****"
3200 PRINT"*****"
3300 FORI=0TO7:FORJ=0TO110STEP22:POKEO+161+I+J,160:POKET+161+I+J,4:NEXT:NEXT
3310 RETURN
3400 PRINT"*****"
3410 PRINT"*****"
3420 RETURN
4000 IFCN=1THENGOSUB3200:GOSUB3300
4010 IFCN=2THENGOSUB3100:GOSUB3300
4020 IFCN=3THENGOSUB3100:GOSUB3200:GOSUB3300
4030 IFCN=4THENGOSUB3200:GOSUB3400
4040 IFCN=5THENGOSUB3100:GOSUB3400
4050 IFCN=6THENGOSUB3400
4060 IFCN=7THENGOSUB3300
4070 IFCN=8THENGOSUB3100:GOSUB3200:GOSUB3400
4080 RETURN
4100 PRINT"*****YOU'VE DONE IT!"
4110 PRINT"YOU COMPLETED THE MAZE"
4120 PRINT"IN";MC;" MOVES AND"
4130 PRINT"CALLED FOR HELP";HC
4140 PRINT"TIMES,"
4150 PRINT"*****ANOTHER GO ?"
4160 GET E$
4170 IFE$="Y"THENRUN
4180 IFE$="N"THENPRINT"J":POKE36879,27:POKE36878,0:END
4190 GOTO4160
4200 GOSUB2000:GOSUB3100:GOSUB3200
4210 PRINT"*****3-D MAZE"
4220 PRINT"*****FOR VIC-20"
4230 FOR I=1TO4000:NEXT:RETURN
5000 PRINT"
5010 PRINT"
5020 PRINT"
5030 PRINT"
5040 PRINT"
5050 PRINT"
5060 PRINT"
5070 PRINT"
5080 PRINT"
5090 PRINT"
5095 PRINT"PRESS '*' TO CONTINUE"
5099 RETURN
5100 POKE36866,150:PRINT"*****ILLEGAL MOVE, TRY AGAIN"
5110 FORI=1TO3000:NEXT
5120 POKE36866,0:RETURN
5200 FORS=200TO250STEP10:POKE36876,S:FORI=1TO80:NEXT:NEXT:POKE36876,0
5210 RETURN
5220 POKE36874,150:FORI=1TO500:NEXT:POKE36874,0:RETURN

```

Σ' αυτό το παιχνίδι προσπαθείτε να εμποδίσετε έναν αετό να φτάσει στη γη. Τα κοντρόλ είναι:

αριστερά ,  
δεξιά .  
πυροβόλο : F

Αν το παιχνίδι είναι εύκολο ή δύσκο-

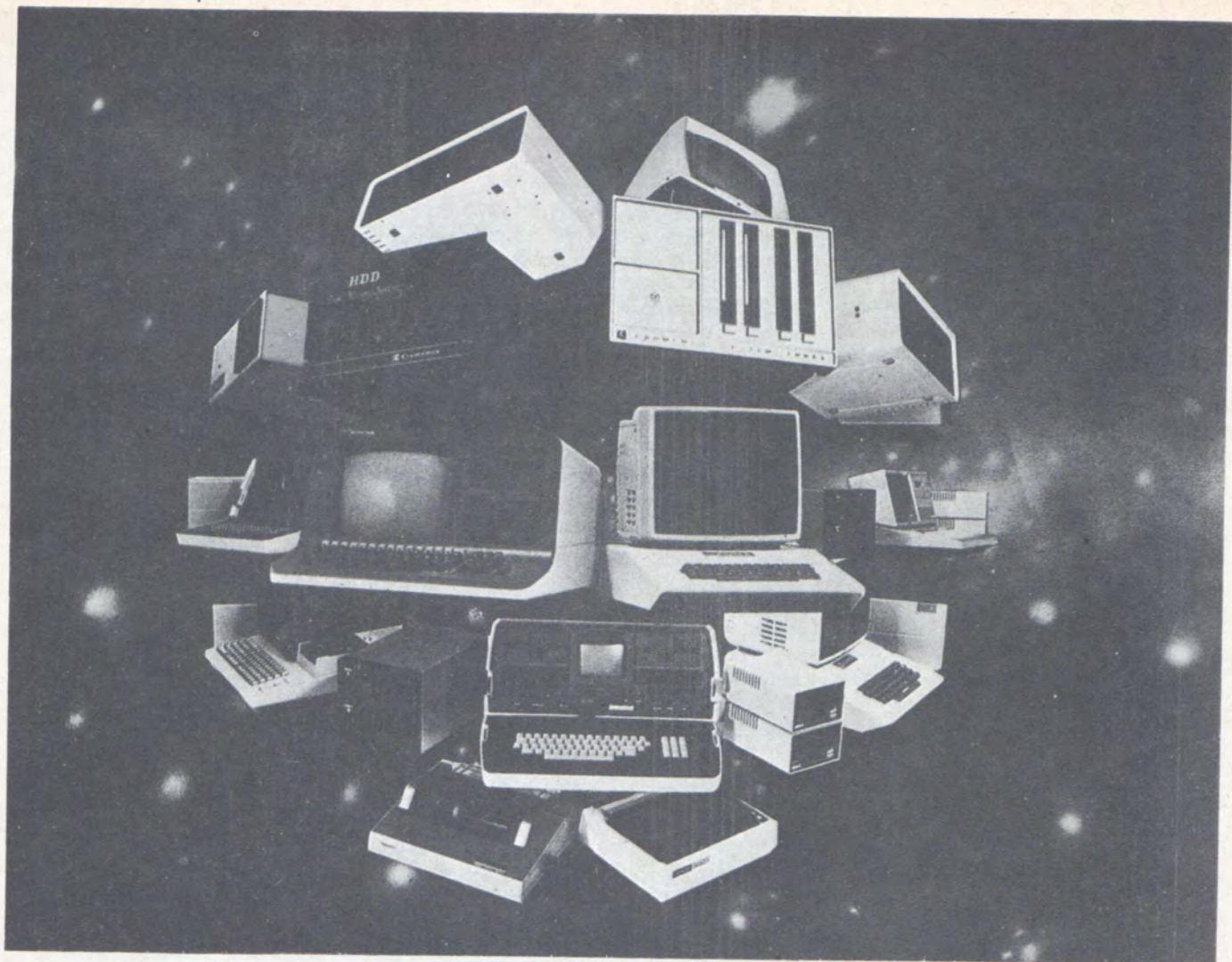
λο, αλλάξτε τον αριθμό 20 της γραμμής 502 (μικρότερος και το παιχνίδι γίνεται δυσκολότερο, μεγαλύτερος και γίνεται ευκολότερο). Τέλος, το πρόγραμμα χρησιμοποιεί περίπου το 95% της μνήμης του unexpanded Vic.

## ΕΙΣΒΟΛΕΙΣ

```

5 PRINT"J":HI=0:Q=0
10 POKE36879,2380126
30 PRINTTAB(7)"SEAHAWK"
40 PRINTTAB(7)"-----"
45 PRINT
50 PRINT"KILL THE SWOOPING SEAHAWK!BUT BE
  WARNED OUT OF ITS DEATHPLACE WILL
60 PRINT"COME ANOTHER."
65 PRINT"X"
70 PRINT"WATCH OUT FOR ITS DD'S<DEADLY DROPPINS>
  GET HIT 3 TIMES AND YOU'RE**DEAD**"
75 PRINT
80 PRINT"WHEN IT GETS TO THE BOTTOM YOU'RE
  KILLED AS WELL!"
90 PRINT"X00000 HIT ANY KEY ! ";
91 POKE198,0:WAIT198,1:POKE198,0
100 PRINT"J"
101 PRINT"YOU DO HAVE ONE THING ON YOUR SIDE YOU
  CAN FIRE BACK AT IT WITH YOUR ARROWS."
102 PRINT"X007YOU HAVE AN UNLIMITED SUPPLY OF THESE."
103 PRINT:PRINT
105 PRINT"THE CONTROLS ARE :-"
110 PRINT"X00F -FIRES THE ARROW"
120 PRINT"X, -MOVES TO THE LEFT"
130 PRINT"X, -MOVES TO THE RIGHT"
135 PRINT"X00PLEASE WAIT!!"
140 GOTO1000
150 PRINT"X00PRESS ANY KEY TO START";
160 POKE198,0:WAIT 198,1:POKE198,0
170 LI=3:SC=0
180 PS=7712:MA=8130
190 PC=38432:MC=38850
195 PRINT"J"
200 POKE36879,25
210 FORI=0TO43:POKE8142+I,63:POKE38862+I,3:NEXT
220 POKEPS,59:POKEPS+1,60:POKEPC,2:POKEPC+1,2
230 POKEMA,61:POKEMC,6
235 PRINT"X";SC;" ";LI
236 POKEMB,32
240 A=PEEK(197)
250 IFA=29THENMA=MA-1:MC=MC-1:POKEMA+1,32:IFMA<=
  8120THENMA=8120:MC=38840
260 IFA=37THENMA=MA+1:MC=MC+1:POKEMA-1,32:IFMA>=
  8141THENMA=8141:MC=38861
265 IFA=42THENGOTO300
270 POKEMA,61:POKEMC,6
275 GOTO500
280 GOTO235
300 M1=MC-22:MB=MA-22
305 M1=M1-22:MB=MB-22
310 POKEMB+22,32:POKEMA,61
320 POKEMB,30:POKEM1,0
330 IFPEEK(MB-22)>=59THENGOTO900
340 IFMB<=7724THENGOTO275
350 GOTO305
500 A=INT(RND(1)*2)+1
501 Q=Q+1
502 IFQ=10THENGOSUB550
510 IFA=1THENPS=PS+1:PC=PC+1:POKEPS-1,32:IFPS=
  7723THENPS=7722:PC=38442
520 IFA=2THENPS=PS-1:PC=PC-1:POKEPS+2,32:IFPS=
  7702THENPS=7703:PC=38423
530 POKEPS,59:POKEPS+1,60:POKEPC,2:POKEPC+1,2
534 B=RND(1)
535 IFB>.9THENGOTO600
540 GOTO280
550 Q=0
555 PC=PC+22
560 PS=PS+22:POKEPS-22,32:POKEPS-21,32:POKEPS,59:
  POKEPS+1,60
565 POKEPC,2:POKEPC+1,2
566 IFPS>8120THENGOTO2020
570 RETURN
600 PB=PS+22:P1=PC+22
610 PB=PB+22:P1=P1+22
620 POKEPB-22,32:POKEPS,59
630 POKEPB,62:POKEP1,7
640 IFPB>=8142THENPOKEPB,32:POKEPB,63:POKEP1,3:
  GOTO280
650 IFPEEK(PB+22)=61THENLI=LI-1:GOSUB2000:GOTO280
660 GOTO610
900 R=INT(RND(1)*3)+1:PS=PS+R:PC=PC+R:SC=SC+1
901 POKEPS,59:POKEPS+1,60:POKEPC,2:POKEPC+1,2:
  POKEMB,32:POKEPS-R,32:POKEPS-R+1,32
902 POKE36878,15:POKE36877,250:FORI=1TO250:
  NEXT:POKE36877,0
910 POKE36878,0:GOTO235
1000 POKE52,28:POKE56,28:CLR
1010 FORI=7168TO7679:POKEI,PEEK(I+25600):NEXT
1020 POKE36869,255
1025 A=7640
1030 FORI=ATORA+39:READU:POKEI,U:NEXT
1040 DATA7,29,15,7,3,3,7,5
1050 DATA156,184,240,224,192,192,224,160
1060 DATA0,36,36,60,24,24,126,102
1070 DATA0,24,80,118,60,24,8,0
1080 DATA255,255,255,255,255,255,255,255
1090 GOTO150
2000 FORI=255TO200STEP-1:POKE36878,15:POKE36877,I:
  NEXT:POKE36878,0:POKE36877,0
2005 POKEPB,32
2006 IFLI=0THEN2020
2010 RETURN
2020 PRINT"SORRY THE SEAHAWKS HAVE GONE AND
  GOT YOU"
2030 PRINT"YOU MANAGED TO KILL";SC;
2040 PRINT"SEAHAWKS."
2045 IFSC=HIGOTO2085
2050 IFSC>HITHEPRINT"X00YOU'RE THE NEW SEA-
  HAWK CHAMPION KILLER":HI=SC:GOTO2090
2060 IFSC=0THENPRINT"X00YOU'RE PATHETIC !":GOTO2100
2070 IFSC<HI/2THENPRINT"X00NOT VERY GOOD":GOTO2100
2080 IFSC>HI/2THENPRINT"X00NOT BAD":GOTO2100
2085 IFSC=HITHEPRINT"X00JOINT CHAMP!!":SC=HI:N#=J#
2090 INPUT"X00YOUR NAME CHAMP";N#
2100 PRINT"PRESENT CHAMP IS ";N#
2105 PRINT"CHAMPS SCORE IS ";HI
2110 PRINT"X00DO YOU WISH TO PLAY AGAIN (Y/N)!!"
2120 GETA$:IFA$=""THEN2120
2130 IFA$="Y"THENGOTO170
2140 IFA$="N"THENPRINT"J":END
2150 GOTO2120

```



# COMPUTER ΓΙΑ ΟΛΟΥΣ

Τό κατανοητό περιοδικό  
γιά τούς μικροϋπολογιστές

Πρωτότυπα θέματα • Τέστ κομπιούτερ • Επαγγελματικά προγράμματα • Μαθήματα γλώσσας • Σελίδες γιά άρχαριους • "Ετοιμα προγράμματα • Μεγάλοι διαγωνισμοί μέ πλούσια δώρα • Εξαρτήματα και περιφερειακά • Νέα προϊόντα • Οδηγός αγοράς • Ψυχαγωγικά προγράμματα...



**COMPUTER**  
ΓΙΑ ΟΛΟΥΣ

Βερανζέρου 15, ΑΘΗΝΑ-141

**ΔΕΛΤΙΟ ΕΓΓΡΑΦΗΣ ΣΥΝΔΡΟΜΗΤΗ  
(ΕΚΠΤΩΣΗ 25%)**

Παρακαλώ νά μέ έγνράψετε συνδρομητή στό περιοδικό COMPUTER ΓΙΑ ΟΛΟΥΣ γιά ένα χρόνο (11 τεύχη). Γιά τό σκοπό αυτό σάς απέστειλα τήν ταχυδρομική έπιταγή Νο..... μέ τό ποσόν τών 1.250 δρχ., αντί τών 1.650 τής κανονικής συνδρομής (έκπτωση περίπου 25%). \*Αν γιά όποιοδήποτε λόγο δέν μέινω εύχαριστημένος από τό περιοδικό, θά μπορώ νά διακόψω τή συνδρομή μου καί νά πάρω πίσω τό υπόλοιπο τών χρημάτων μου χωρίς τήν παραμικρή καθυστέρηση.

ΟΝΟΜΑΤΕΠΩΝΥΜΟ

ΔΙΕΥΘΥΝΣΗ

ΤΗΛΕΦΩΝΟ

T.T.

ΕΠΑΓΓΕΛΜΑ

Η ΣΥΝΔΡΟΜΗ ΜΟΥ Ν'ΑΡΧΙΖΕΙ ΑΠΟ ΤΟ ΤΕΥΧΟΣ Νο.....

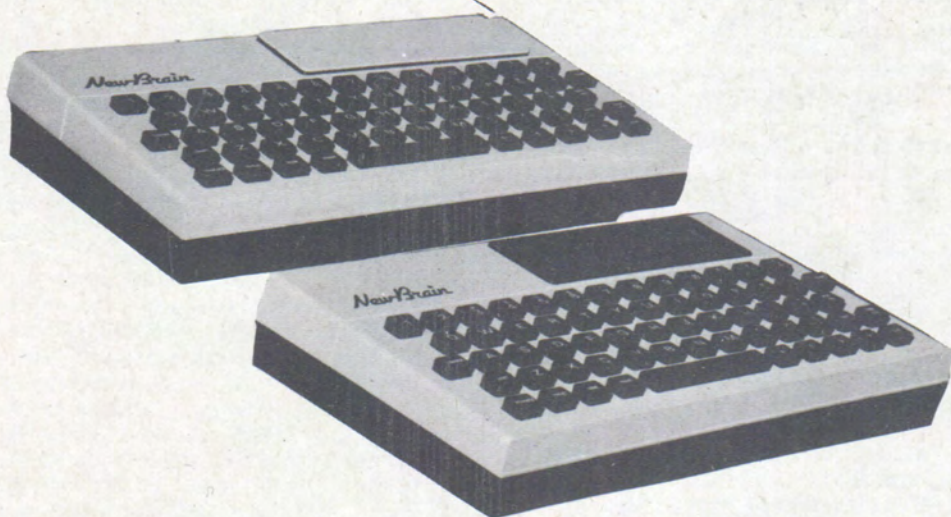
(\*Αν δέν θέλετε νά κόψετε τή σελίδα, μπορείτε νά χρησιμοποιήσετε ένα όποιοδήποτε φύλλο χαρτιού ή φωτοαντίγραφο)

ΚΥΚΛΟΦΟΡΕΙ ΚΑΙ ΓΙΝΕΤΑΙ ΑΝΑΡΠΑΣΤΟ!

Προγράμματα για τόν:

# NEW BRAIN

ΕΠΙΜΕΛΕΙΑ:  
Φώτης Καρατζιάς



Το πρόγραμμα γραφικών παραστάσεων που παρουσιάζουμε είναι μια προσφορά της GRUNDY στους αναγνώστες του "COMPUTER ΓΙΑ ΟΛΟΥΣ". Είναι τελείως φιλικό προς το χρήστη αφού εργάζεται με το σύστημα των πινάκων επιλογής και περιέχει πλήρη επεξηγηματικά σχόλια. Δίνει τη δυνατότητα σχεδιασμού κυκλικού διαγράμματος (γραμμή 1000), διαγράμματος μηνών (γραμμή 2000), διαγράμματος X-Y (γραμμή 3000), κανονικής κατανομής (γραμμή 4000), καμπύλης αποσβέσεως (γραμμή 5000), παραβολοειδούς

## ΓΡΑΦΙΚΕΣ ΠΑΡΑΣΤΑΣΕΙΣ

(γραμμή 6000) και δικών μας σχεδίων (ζωγραφική, στη γραμμή 10000). Φυσικά, τα σχέδια που θα κατασκευάσουμε στην οθόνη, επιλέγοντας μια από τις παραπάνω δυνατότητες, μπορούμε να τα εκτυπώσουμε και σε εκτυπωτή Seikosha. Για την εκτύπωση φροντίζει το πρόγραμμα. Εμείς, αρκεί να πούμε ένα ναι ή ένα όχι. Στη γραμμή 40000 υπάρχει μια ρουτίνα που μας ζητάει να δώσουμε την ώρα, ενώ στη 41000 αρχίζει η υπορουτίνα της ενημέρωσης του real time clock. Καλά σχέδια λοιπόν.

```

5 REM by C Georgopoulos, Ph.D. *** 23.Αύγ. 1982
10 FORi=1TO255:CLOSE#i:NEXTi
15 ON BREAK GOTO30000
20 OPEN#0,0,"s150":OPEN#11,11,"w150":s=5
30 OPEN#9,9,"2400":OPEN#5,5:OPEN#6,6:OPEN#20,0,20,"s":PUT#5,1:OPEN#52,0,52,"10"
50 s=5:GOSUB52000:REMdelete52000-52210
60 DIMh(12)
70 DATA:διαγραμμα x-y,Κυκλικο διαγραμμα,Ιστογραμμα μηνων,Κανονικη κατανομη,Καμπυλη
   η αποσβεσεως,Παραβολοειδης,Ζωγραφικη
75 RESTORE70:FORi=1TO7:READt#(i):NEXTi
80 GOSUB40000
100 REM ***** MAIN PROGRAM *****
110 PUT#20,31,22,2,2,143:FORi=1TO34:PUT#20,129:NEXTi:PUT#20,142:FORi=3TO20:PUT#2
0,22,2,I,150,22,37,I,130:NEXTI:PUT#20,22,2,20,141:FORi=1TO34:PUT#20,129:NEXTI:PU
T#20,140
115 PUT#20,22,3,4:?"#20,"
120 PUT#20,22,10,7:?"#20;"ΓΡΑΦΙΚΕΣ ΠΑΡΑΣΤΑΣΕΙΣ":PUT#20,22,10,8:?"#20,"
"
130 FORi=1TO7: PUT#20,22,8,9+i:?"#20,STR#(i)+CHR#(8)+" . ";t#(i):NEXTi
140 PUT#20,22,10,18:?"#20,"Ποιο θελετε; ";:k=1
150 GET#6,x:IFx=0GOSUB41000:GOTO150
152 IFx=42GOTO30010
155 IFx>48ANDx<56THEN170
160 IFk=1THEN?"#20,"*";CHR#(8):GOTO165
164 ?"#20," ";CHR#(8);
165 k=-k:FORl=0TO100:NEXTl:GOTO150
170 ON x-49 GOSUB3000,1000,2000,4000,5000,6000,10000:GOTO100
199 STOP
999 STOP
1000 REM ***** pie chart *****
1015 PUT#0,31:?"#0,,t#(2)
1020 plotwip,rng(8,4.4),cen(4,2),deg,pla(0,-1),turn(0),arc(2*PI,360)
1025 ?#0,"Τιτλος: ";:LINPUT#0,a#:l=LEN(a#)/10:plotpla(-1,2:1),a#
1030 ?#0,"συνολικο ποσο: ";:INPUT#0,t#:j=2:PUT#0,31:?"#0,"δωστε τμηματα και ποσα."
1035 PUT#0,22,1,3,30:?"#0,"υποτιτλος τμηματος ";:LINPUT#0,a#
1040 PUT#0,22,1,3,30:?"#0,"ποσο τμηματος ";a#:":":INPUT#0,x:PUT#0,2:IFr+x>t THE
N ?#0,"πορα πολυ (μονο ";t-r;" εχει μεινει)":GOTO1040

```



```

1045 a=360#r/t:b=360#x/t:plotpla(0,0),turn(a),dby(1,1),turn(a+b),dby(1,1),turn(a
+b/2),col(0),mby(1.2),col(1):IF pen(0)<0 THEN l=LEN(a$)*.19:plotpla(pen(0)-1,pen
(1))
1050 plotaf:j=j+1:plotpla(0,0):IF j<3 THEN 1060
1055 j=1:IF r+x<t AND x>t/100 THEN plotturn(a+b/2),mby(.9),col(1),fill
1060 a=a+b:n=r+x:IF r<t THEN 1035
1070 ?#0,"ελετε εκτυπωση;":GET#5,z#:IF z#="n" OR z#="N" PUT#9,27,76,2,10:CALL 128
1110 RETURN
2000 REM ***** duple screen bar chart *****
2010 PUT#1:plotwipe
2015 ?,t#(3)
2020 plotbck(0),wipe,col(1),rng(13.4,140),cen(1.5,20),pla(0,0),axe(0,10)
2025 plotpla(.1,-10),"Ια Φε Μα Απ Μω Ιο Ιλ Αυ Σε Οκ Νο Οε"
2026 plotpla(3,110),t#(3)
2030 PUT#0,31:"ποσοτα μηνων"
2040 FOR x=0 TO 11:?(x+1)[2]:INPUT(" ")h:h(x)=h:plotpla(x,0),mve(x,h),mve(x+.5,h),m
ve(x+.5,0):IF h>2.6 THEN plotpla(x+.2,1),fill
2050 plotpla(x,0),mve(x,h),mve(x+.5,h),mve(x+.5,0):IF h>2.1 THEN plotpla(x+.2,1),fi
ll
2060 NEXT x
2090 ?#0,"ελετε εκτυπωση;":GET#5,z#:IF z#="n" OR z#="N" PUT#9,27,76,2,10:CALL 128
2300 RET
3000 REM ***** x-y plot *****
3055 PUT#1:plotwipe:?,t#(1)
3060 INPUT("μεγιστο x: ")mx
3070 INPUT("μεγιστο y: ")my
3080 INPUT("ποσα σημεια: ")n
3090 plotrng(1.2*mx,1.2*my),cen(.1*mx,.1*my),pla(0,0),axe(mx/n,my/n)
3100 PUT#1:" x y ":FOR i=0 TO n:YN=mx*i/n:?(XN[3,2]:" "):INPUT(" ")YN:PLOTM
VE(XN,YN)
3110 NEXT I
3900 ?#0,"ελετε εκτυπωση;":GET#5,z#:IF z#="n" OR z#="N" PUT#9,27,76,2,10:CALL 128
3999 RET
4000 REM ***** normal distr *****
4055 PUT#1:plotwipe:?,t#(4)
4060 ?#0,"σ ":INPUT#0,t:s=SQR(t):h=LOG(1/t)/2.30258+1: ?#0,h:IF h<0 LETh=1
4070 ?#0,"μ ":INPUT#0,m
4110 plot#11,wip,rng(9,1.2*h),cen(4,.2*h),pla(0,0),axe(1,.5)
4111 plot#11,pla(-4,-.2*h),"x+μ=":FOR x=-3 TO 3:d=x+m:plot#11,pla(x,-.2*h),d:NEXT x:
plot#11,pla(-4,0)
4120 FOR x=-4 TO 4 STEP .1
4130 y=EXP((-5)*(x/s)^2)/s/SQR(PI)
4140 plot#11,mve(x,y)
4150 NEXT x
4200 ?#0,"ελετε εκτυπωση;":GET#5,z#:IF z#="n" OR z#="N" PUT#9,27,76,2,10:CALL 128
4400 RETURN
5000 REM ***** damping ***
5015 PUT#1:plotwipe: ?#0,t#(5)
5020 ?#0,"περιοδος: ":INPUT#0,t
5030 ?#0,"αποσβεση: ":INPUT#0,B
5040 plot#11,wip,rng(1.2*t,2.2),cen(.1*t,1),pla(0,0),axe(1,.5),COL(1)
5050 plot#11,pla(1.05*t,-.2),"T",pla(0,0)
5100 FOR I=0 TO 220 STEP 320/T/20
5105 X=I*T*2*PI/320
5110 Y=SINK(X)*EXP(-B*X)
5120 PLOT#11,MVE(I*T/320,Y)
5130 NEXT I
5125 ?#0,"ελετε εκτυπωση;":GET#5,z#:IF z#="n" OR z#="N" PUT#9,27,76,2,10:CALL 128
5150 RETURN
6000 REM ***** 3-d drawing *****
6010 PUT#0,31: ?#0,t#(6)
6020 plotwip,rng(32,20),cen(16,10),pla(0,0),axe(0,0)
6100 FOR z=-10 TO 10 STEP .5:plotpla(0,0)
6150 FOR t=0 TO 2.1*PI STEP .1
6200 x=(.1*z*z+5)*SINK(t):y=2*COS(t)+z*1.1:GOSUB 7000
6230 NEXT t:NEXT z

```

```

6240 ?#0,"θελετε εκτυπωση;":GET#5,z#:IFz#="n"ORz#="N"PUT#9,27,76,2,10:CALL129
6900 RETURN
7000 plotmve(x,y):RET
10000 CLOSE#11
10001 PUT#0,31:?"Οι ενδείξεις στην κορυφή της οθόνης δείχνουν την οριζόντια και
την κάθετη θέση του μολυβιού,τη γωνία γραφής του(σε θ μοίρες γραφει οριζόντια δε
ξιά).Αν η ένδειξη χρώματος είναι 1 το μολύβι ζωγραφίζει και αν την αλλάξετε σε 2
προχωρεί χωρίς να γραφει.Οι διαστάσεις της οθόνης είναι:οριζόντια -150 έως 150
και κάθετα -75 έως 75.Ζωγραφίζετε πατώντας SPACE BAR .Αλλάζετε κατεύθυνση 90 μοί
ρων με τα 4 βελάκια,καθως επίσης με τα πλήκτρα R και L για 5 μοίρες δεξιά ή αρι
στερά αντίστοιχα.Αν θελετε κύκλο πατήστε K.Για να χρωματίσετε ένα χώρο πατήστε F.
Για να μετακινήηθε χωρίς να γραφει πατήστε 2 για να αλλάξει το χρώμα.Όταν τελε
λώσετε το σχέδιο σας πατήστε E

```

πατήστε ένα πλήκτρο

```

για να συνεχίσουμε":GET#5,z
10007 OPEN#0,0,"S150":OPEN#11,11,"W150":plotwipe,rng(320,150),cen(160,75),deg,pl
a(0,0),COL(1),bck(0)
10008 PLOTPLA(-155,-70),MVE(-155,70),MVE(155,70),MVE(155,-70),MVE(-155,-70),PLA(
0,0)
10009 PUT#0,31,22,1,1:?"οριζόντια κάθετα γωνία χρώμα":PUT#6,1
10010 GET#6,z:IFz=0GOTO10010
10015 IFz>32GOTO10053
10020 IFz=26THEN d=0:plottrn(d):GOTO10066
10030 IFz=11THEN D=90:plottrn(D):GOTO10066
10040 IFz=8THEN d=180:plottrn(d):GOTO10066
10050 IFz=10THEN d=270:plottrn(d):GOTO10066
10053 IFz=75THENPUT#0,22,1,4:?"δωστε X,ψ,ακτίνα,τόξο :":x,y,z,u:PLOTPLA(x,
y),arc(a,u):PUT#0,22,1,4:PUT#0:GOTO10066
10058 IFz>47ANDz<52THENPLOTCOL(z-48):GOTO10066
10060 IFz=32THENPLOTMBY(5):GOTO10066
10061 IFz=70THENPUT#0,22,1,4:INPUT("δωστε X,ψ μέσα στο χώρο που θελετε να γεμίσω
":x,y:plotpla(x,y),fill:PUT#0,22,1,4:PUT#0:GOTO10066
10062 IFz=82THENLETD=D-5:PLOTTRN(D):GOTO10066
10064 IFz=76THENLETD=D+5:plottrn(d):GOTO10066
10065 IFz=69THEN GOTO10000
10066 PUT#0,22,1,2:?"0,pen(0)[4],pen(1)[4],pen(2),pen(3)
10067 K=PEN(0):L=PEN(1):PLOTPLA(K,L),DOT(K,L,1)
10070 GOTO10010
10080 ?#0,"θελετε εκτυπωση;":GET#5,z#:IFz#="n"ORz#="N"PUT#9,27,76,2,10:CALL129
10085 RET
20000 plotrng(320,150),cen(160,75),deg,pla(0,0),COL(1),bck(0),trn(90)
20002 FORt=40TO60STEP10:FORj=-150TO150STEP10
20004 plotpla(j,t),"*"
20006 NEXTj:NEXTt
20010 plotpla(x,-60),dby(140,2):GET#6,z:IFz=0GOTO20010
20020 IFz=11THENplotpla(x,-60),dby(140,1):GOTO20010
20028 IFz=9ANDx<-150THENx=-150:GOTO20010
20030 IFz=8THENplotpla(x,-70),col(2),"*:x=x-10:plotpla(x,-70),col(1),"*"
20038 IFz=26ANDx>150THENx=150:GOTO20010
20040 IFz=26THENplotpla(x,-70),col(2),"*:x=x+10:plotpla(x,-70),col(1),"*"
20060 GOTO20010
30000 RESUME 100
30010 ONBREAK GOTO0:STOP
30020 GOTO100
40000 REM ***# SET HOUR ***#
40010 PUT#52,22,5,5:?"#52,"δωστε μου την ώρα :":PUT#52,8,9
40015 INPUT#52,z#:IFNUM(z#)=FALSE GOTO40010
40018 th=VAL(z#):IFth>24GOTO40010
40025 PUT#52,22,5,7:?"#52,"δωστε μου λεπτα :":PUT#52,8,9
40028 INPUT#52,z#:IFNUM(z#)=FALSE GOTO40025
40029 tm=VAL(z#):IFtm>60GOTO40025
40050 ta=th*3600+tm*60:CALL 62383
40051 REMDELETE40000-40051
40055 RETURN
41000 REM ***# Real time clock update routine ***
41010 CALL 62399,x:tt=INT(x/50.09)+ta:IF ta>86399 THEN ta=ta-86400

```

```

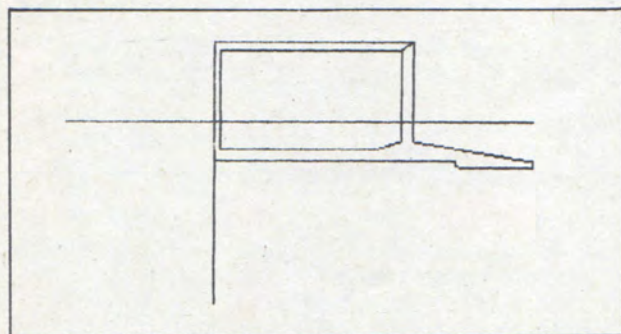
41050 th=INT(tt/3600):tm=INT((tt-th*3600)/60):ts=tt-th*3600-tm*60:t#=" "+STR$(th
[2])+" ":"+STR$(tm[2])+" ":"+STR$(ts[2])+" "
41120 PUT#20,22,3,3: ?#20,t#
41130 RETURN
51000 REM save graphic picture
51010 d=PEEK(tv+4):l=PEEK(tv+6):w=PEEK(tv+5):st=640+(d*1)+(w/20)+w:g=(25-d)*10:n
=g#w:STOP
51020 CLOSE#1:OPEN OUT#1,1:FOR x=st TO st+n:PUT#1,PEEK(x):NEXTx:CLOSE#1:RET
51500 REM read graphic picture
51510 CLOSE#1:OPEN IN#1,1:FOR x=st TO st+n:GET#1,q:POKE x,q:NEXTx:CLOSE#1:RET
52000 REM dump graphics to printer
52010 DATA23,0,0,229,237,75,126,0,42,124,0,84,93,19,54,0,237,176,225
52020 DATA62,198,50,172,0,245,1,8,0,237,91,124,0,62,70,50,167,0,245,203,0,40,4
52030 DATA235,203,0,235,19,241,129,254,127,56,237,35,16,232
52040 DATA235,33,0,0,241,237,82,235,56,4,129,48,200,63,245,229,33,252,0,1,4,0
52050 DATA205,240,0,6,90,30,9,175,231,48,16,249,42,124,0,237,75,126,0,205,240,0
52060 DATA62,10,30,9,231,48,225,241,63,56,148,201
52070 DATA 30,9,126,231,48,35,11,121,176,32,245,201,27,71,1,144
52100 RESTORE52010:FORx=128 TO 255:READ da:POKE x,da:NEXTx
52200 tv=PEEK(93)*256+PEEK(92):d=PEEK(tv+4):l=PEEK(tv+6):w=PEEK(tv+5):ex=0:IF s=
4 THEN ex=s
52250 IF w>40 THEN ex=ex*2
52260 g=(25-d)*10:st=((INT((PEEK(tv)+tv)/64)+1)*64)+(d*1)+(w/20)+ex:w=w*(s/5):n=
g#w:bl=w*8:h=INT(bl/256):POKE 127,h:l=bl-(h*256):POKE 126,l
52350 RESERVE bl:bs=TOP:h=INT(bs/256):POKE 125,h:l=bs-(h*256):POKE 124,l:h=INT(s
t/256):POKE 130,h:l=st-(h*256):POKE 129,l:se=st+n-1:h=INT(se/256):POKE 137,h:l=s
e-(h*256):POKE 186,l:POKE 155,w
52600 PUT#9,27,76,2,10: REM Set Seikosha to graphics line feed.
52999 RETURN
65200 CLOSE#2:OPENIN#2,2:MERGE#2:CLOSE#2
65210 SAVE FILE#
65220 DELETE-65150
65230 GOTOC5200

```



ΔΥΣΤΥΧΟΣ ΚΑΤΕΧΕΤΑΙ ΑΠΟ ΟΙΔΙΠΟΔΕΙΟ ΣΥΜΠΛΗΓΜΑ.

# ΠΡΟΓΡΑΜΜΑ ΣΤΟΙΧΕΙΩΝ ΣΥΜΜΕΤΡΙΚΩΝ ΔΙΑΤΟΜΩΝ



Το πρόγραμμα που ακολουθεί, βρίσκει το κέντρο βάρους, το εμβαδόν και τη ροπή αδράνειας μιας συμμετρικής διατομής. Επιπλέον σχεδιάζει το ένα από τα δυο συμμετρικά κομμάτια της διατομής και σημειώνει τη θέση του κέντρου βάρους.

Σαν εισόδους δέχεται τον αριθμό των κορυφών της διατομής (η) και τις συντεταγμένες τους (y(i), z(i)). Σαν παράδειγμα εφαρμογής δίνουμε μια διατομή με 16 κορυφές. Το πρόγραμμα είναι προσφορά του CAT COMPUTERS (Ιπποκράτους 57, τηλ. 3643-044).

## παράδειγμα εφαρμογής

i	y(i)	z(i)
1	2.00	0.00
2	5.30	0.00
3	5.30	2.40
4	1.30	2.40
5	1.28	2.50
6	0.00	2.53
7	0.00	2.40
8	2.00	2.00
9	2.00	0.00
10	2.20	.15
11	2.20	2.00
12	2.60	2.15
13	5.20	2.15
14	5.20	.15
15	2.20	.15
16	2.00	0.00

## αποτελέσματα

κέντρο βάρους σε απόσταση = 1.576  
εμβαδόν διατομής A= 2.50  
ροπή αδράνειας I= 2.15

```

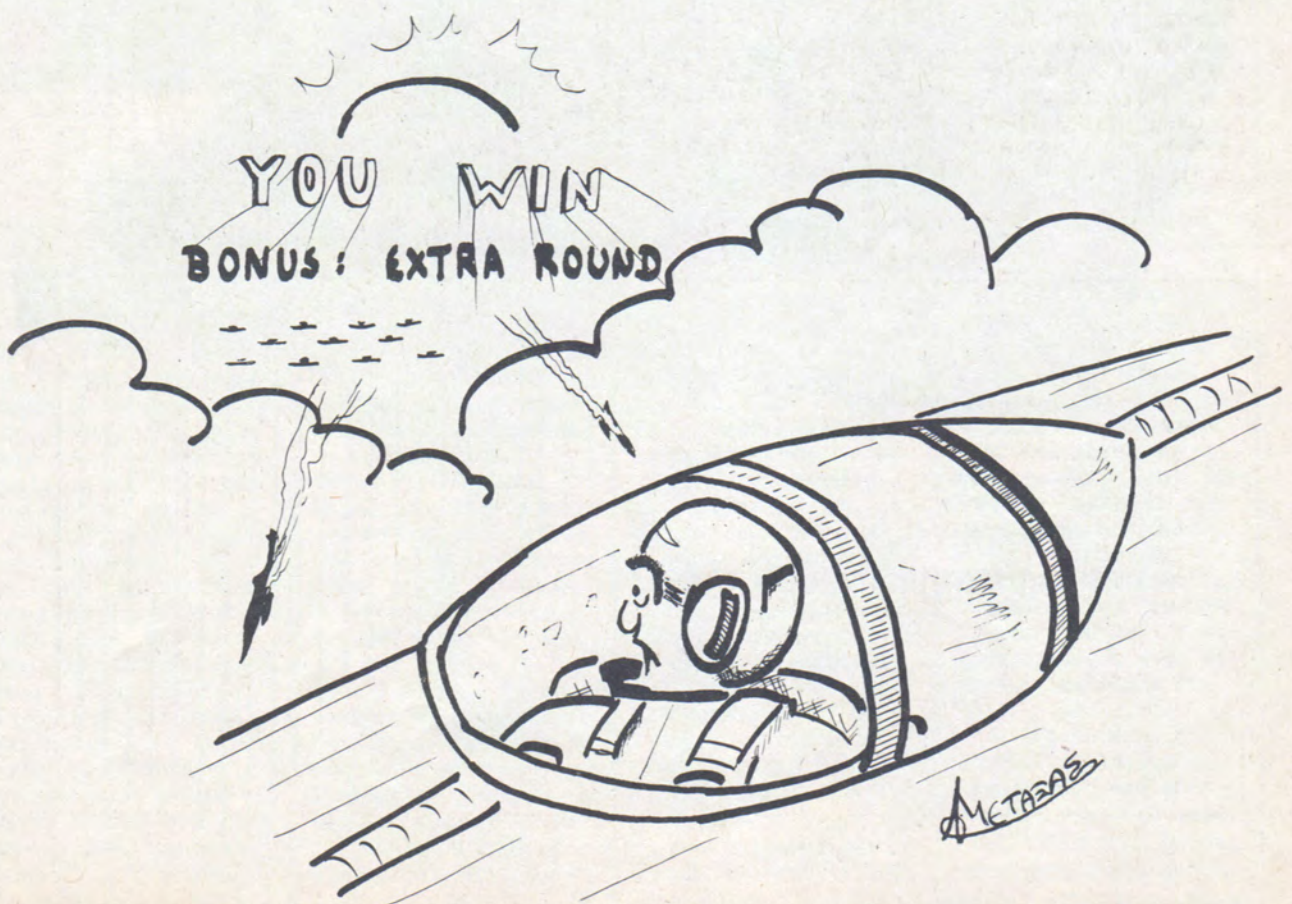
10 REM Cat Computers
20 REM Ιπποκράτους 57-Αθήνα-3643044
30 ?"πρόγραμμα στοιχείων συμμετρικών διατομών"
40 INPUT("πλήθος κορυφών ") n
50 DIM y(n+1),z(n+1)
60 ?"συντεταγμένες κορυφών: y(i),z(i)"
70 FOR i=1 TO n
80 ?i: INPUT(" ") y(i),z(i)
90 NEXT i
93 GOSUB 500
94 PUT 31
95 GOSUB 1000
100 y(n+1)=y(1):z(n+1)=z(1)
110 a=0:s=0:i0=0
120 FOR i=1 TO n
130 x=(y(i)*z(i+1)-y(i+1)*z(i))/2

```

```

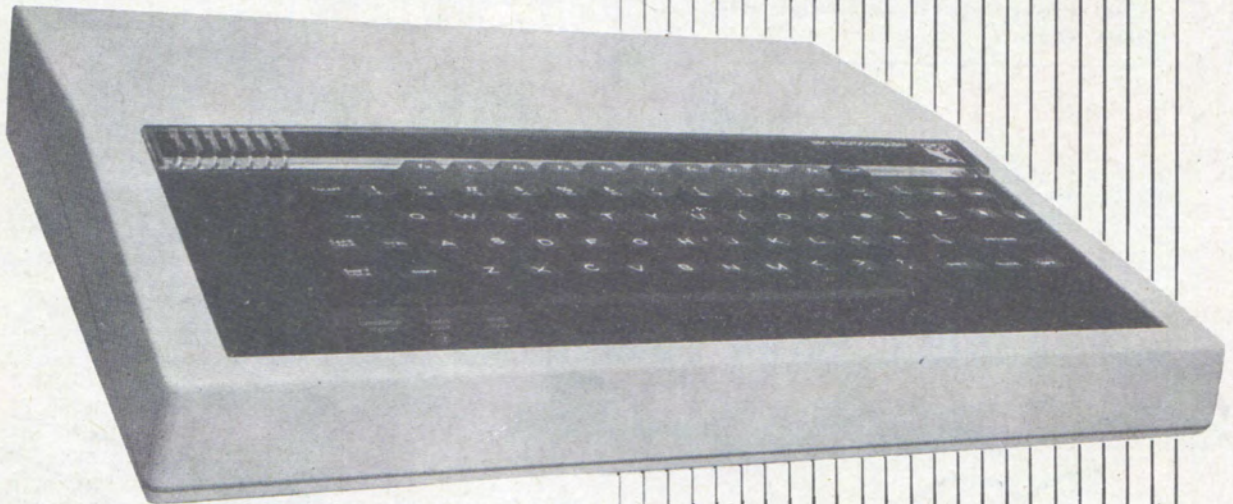
140 a=a+x
150 s=s+1/3*x*(z(i)+z(i+1))
160 i0=i0+1/6*x*(z(i)^2+z(i)*z(i+1)+z(i+1)^2)
170 NEXT i
180 zs=s/a
190 i1=i0-a*zs^2
200 ?"zs=";zs
210 ?"A =" ;a
220 ?"I =" ;i1
300 END
500 p=y(1):r=z(1)
510 p1=0:r1=0
520 FOR i=1 TO n
530 IF y(i)<p THEN 550
540 p=y(i):p1=i
550 IF z(i)<r THEN 570
560 r=z(i):r1=i
570 NEXT i
580 IF p<r THEN 590
585 r=p:GOTO 599
590 p=r
599 RET
1000 FOR x=1 TO 255:CLOSE #x:NEXT x
1005 OPEN #0,0,"s160": OPEN #11,11,"w160"
1010 plotbackground(0),wipe,range(2*p,1*r),centre(0,0),place(y(1),z(1))
1020 FOR i=2 TO n+1
1030 plotmove(1*y(i),1*z(i))
1040 NEXT i
1045 plotplace(p,0),move(p,r)
1046 plotplace(0,zs),move(2*p,zs)
1050 RET

```



Προγράμματα  
για τον:

BBC



ΕΠΙΜΕΛΕΙΑ:  
Δημήτρης Τσουροπλής

## ΤΑΞΙΔΙ ΣΤ' ΑΣΤΕΡΙΑ



Το πρόγραμμα που έχετε μπροστά σας θα δώσει την ευκαιρία στους φίλους της γνωστής τηλεοπτικής σειράς (και κατόχους του BBC...) να αναλάβουν την διοίκηση του διαστημοπλοίου Enterprise και να εξερευνήσουν μ' αυτό τους γαλαξίες, χωρίς να σηκωθούν από την καρέκλα τους...

Το παιχνίδι έχει συγκεκριμένους στόχους, αντίθετα από άλλες εκδόσεις όπου πολεμάτε τους klingons μέχρι αφανισμού (δικού τους ή δικού σας...). Η αποστολή σας είναι να οδηγήσετε το Enterprise εκεί "που δεν έχει πατήσει άνθρωπος", αναζητώντας νέες μορφές ζωής και πολιτισμού. Οι klingons δυστυχώς δεν βλέπουν

με καλό μάτι τις φιλοδοξίες σας, αλλά γι' αυτό υπάρχουν τα phasers... Το παιχνίδι μπορεί να διαρκέσει μέχρι μια ώρα, μέσα στην οποία μπορείτε να ακολουθήσετε όποια τακτική νομίζετε καλύτερη. Αν δεν συμπαθείτε τις μακρόχρονες αποστολές μπορείτε να προσγειωθείτε στο αρχηγείο του αστροστόλου (HQ) και να πληροφορηθείτε την επίδοσή σας. Οι συντομογραφίες των εντολών είναι:

Ο έλεγχος ζημιών αναλύει το μέγεθος της ζημιάς που έχει υποστεί κάθε σύστημα του Enterprise. Όσο μεγαλύτερος είναι ο αριθμός δίπλα σε κάθε σύστημα, τόσο μεγαλύτερη ζημιά έχει υποστεί. Το

ίδιο το σύστημα ελέγχου ζημιών μπορεί να έχει ζημιά, οπότε η εντολή δεν εκτελείται.

Η κατάσταση συναγερμού αναφέρεται στην κατάσταση των ασπίδων σας. Οι ασπίδες δεν είναι ενεργές σε Πράσινο συναγερμό. ενεργοποιούνται σε μισή ισχύ κατά τον Κίτρινο συναγερμό και σε πλήρη ισχύ κατά τον Κόκκινο συναγερμό. Με ζημιά στις ασπίδες, δεν απενεργοποιείται ο Κόκκινος συναγερμός. Οι ασπίδες προστατεύουν από επιθέσεις αλλά καταναλώνουν πολλή ενέργεια κι έτσι δεν είναι σκόπιμο να μένετε για πολύ σε Κόκκινο συναγερμό.

Η επισκευή των ζημιών επιταχύνει την πορεία των εκτελούμε-

νων επισκευών (οι επισκευές γίνονται αυτόματα και χωρίς την εντολή) όταν υπάρχει έκτακτη ανάγκη.

Η Φωτοταχύτητα χρησιμεύει για την μετακίνηση από τομέα σε τομέα, μέσα στο διάστημα. Μετά την εντολή, ζητούνται οι συντεταγμένες του τομέα προορισμού και η ταχύτητα μετάβασης εκεί. Όσο πιο μεγάλη η ταχύτητα, τόσο πιο γρήγορα φθάνετε, αλλά καταναλώνετε και πιο πολλή ενέργεια. Οι συντεταγμένες δίνονται σαν X, Y, Z, με τιμές από 0-7. Όπως βλέπετε πρόκειται για τρισδιάστατο γαλαξία 8X8X8 τομέων. Οι συντεταγμένες του τομέα που βρίσκεστε υπάρχουν στον πίνακα με τις διάφορες ενδείξεις.

Οι κινητήρες σταθεροποίησης τροχιάς χρησιμεύουν στην προσέγγιση σε μια αστροβάση ή στο αρχηγείο. Μπαίνοντας σ' έναν τομέα που έχει αστροβάση, πρέπει να προσεδάφιστείτε σ' αυτήν κινούμενοι αριστερά-δεξιά με τα πλήκτρα κίνησης του δρομέα. Με την προσεδάφιση η ενέργειά σας ανανεώνεται, παίρνετε πίσω Φωτοτορπίλλες και επισκευάζονται μερικές ζημιές. Μετά από αυτά η αστροβάση εξαφανίζεται. Προσέξτε να μην προσγειωθείτε κατά λάθος στο αρχηγείο, που μοιάζει με αστροβάση, γιατί το παιχνίδι τελειώνει. (Ζητείστε τις συντεταγμένες του αρχηγείου από τον κομπιούτερ, για να πάτε εκεί μόνο στο τέλος.)

Ο ερευνητικός κομπιούτερ καταγράφει πληροφορίες από κάθε πλανήτη που συναντάτε στον τομέα σας. Πρέπει να εξερευνήσετε με την εντολή αυτή όλους τους

πλανήτες που θα βρεί-  
τε, για να κάνετε ψη-  
λό σκορ.

Πληροφορίες για  
τους διάφορους τομείς  
και τα περιεχόμενά  
τους, καταγράφει και ο  
κομπιούτερ του διαστη-  
μόπλοιου. Οι πληροφο-  
ρίες που περιέχει προ-  
έρχονται από τις ανιχν-  
εύσεις που μπορείτε  
να κάνετε με την εντο-  
λή L. Οι ανιχνευτήρες  
δίνουν μια οπτική πε-  
ριγραφή των τομέων γύ-  
ρω απ' αυτόν που βρι-  
σκόμαστε, και καταχω-  
ρούν τις πληροφορίες  
στον κομπιούτερ του  
διαστημόπλοιου για  
μελλοντική χρήση.

Τα phasers χρησι-  
μούνται για την εξόντω-  
ση των klingons που θα  
βρεθούν αντιμετώπι-  
σας κατά καιρούς. Αν  
και πάντοτε πετυχαί-  
νουν τον στόχο τους,  
μπορεί να χρειαστούν  
πολλά χτυπήματα για  
την καταστροφή του.  
Επίσης είναι πολύ  
ενεργόβαρα σαν εντολή

Η φωτοπίλλη εί-  
ναι το ύστατο όπλο.  
Καταστρέφει αμέσως τον  
εχθρό, αρκεί να τον  
σημαδέψετε στο κέντρο.  
Ο αριθμός των τορπίλ-  
ων είναι περιορισμέ-

νος, και καλό θα είναι  
να μην γίνεται σπατάλη  
στην χρήση τους. Μπαί-  
νοντας σ' έναν τομέα  
που έχει εχθρό, θα σας  
γίνει αμέσως επίθεση.

Προσπαθείστε λοιπόν να  
έχετε ανεβασμένες τις  
ασπίδες, άφθονη ενέρ-  
γεια, λίγες ζημιές και  
τουλάχιστον μια τορ-  
πίλλη. Υπάρχει ακόμη,  
κάπου στον γαλαξία,  
μια άσπρη (!!) τρύπα  
που θα σας καταστρέψει  
αν πάτε στον τομέα που  
βρίσκεται. Αν βρίσκε-  
στε κοντά της μπορεί  
να σας παρασύρει σε  
άλλο τομέα. Έτσι απο-  
φεύγετε την όσο μπο-  
ρείτε.

Λόγω του μεγέθους  
του, το πρόγραμμα χρη-  
σιμοποιεί την MODE 7  
για κείμενα και  
graphics, οι δε περιο-  
χές του διαστήματος  
αποθηκεύονται απευθεί-  
ας σε περιοχή της μνή-  
μης (2% 512 γραμμή  
2590). Η περιοχή αυτή  
περιέχει ταυτόχρονα  
το ελδος των πλανητών,  
την επιθετικότητα των  
klingons κ.ά. Η πλη-  
κτρολόγηση είναι αρκε-  
τά επίπονη, αλλά το  
σύμπαν δεν κατακτιέται  
χωρίς κόπο!.. Καλά τα-  
ξίδια captain Kirk!

```

XL
 10 MODE7
 20 PRINTTAB(5,10);"Please wait a few seconds."
 30 ENVELOPE 1,7,2,1,1,1,1,1,1,1,21,-10,-5,-2,120,120
 40 X1=10:Y1=10
 50 A=RNDX(-TIME)
 60 ON ERROR GOTO 3690
 70 SC=0:EN=6000:PH=5:CD=1:SH=4000
 80 XQ=0:YQ=0:ZQ=0
 90 GOSUB 2590
100 CLS
110 VDU23;8202;0;0;0;
120 TIME=0
130 VDU 26
140 PROCSCREEN
150 #FX4,0
160 VDU 26
170 PRINTTAB(30,1);"7";TAB(30,1);
180 B0=""
190 A0=GET#
200 PRINTA0;
210 IF A0=CHR(127) THEN B0=LEFT(B0,LEN B0-1):GOTO 190
220 IF A0=CHR(13) THEN 250
230 B0=B0+A0:GOTO 190
240 DATA DAMAGE,ALERT,REPAIR,WARP,IMPULSE,PHASER,PHOTON,SCIENCE,SHIP,LANG
250 RESTORE 240
260 I=1
270 READ A0:IF B0=LEFT(A0,LEN B0) THEN 300
280 I=I+1:IF I>11 THEN 270
290 GOTO 130
300 VDU 20,22,17,30,3

```

```

310 IF D<(I)>0 THEN FOR I1=0 TO 1000:NEXT CLS:PROCDAM(I):GOTO 130
320 ON I GOTO 330,400,500,600,1990,1440,1700,950,730,1130
330 CLS
340 PRINT"DAMAGE CONTROL"
350 PRINT"-----"
360 I=2:REPEAT
370 IF D<(I)>0 THEN PROCDAM(I)
380 I=I+1:UNTIL I=11
390 GOTO 130
400 CLS
410 PRINT"ALERT CONTROL"
420 PRINT"-----"
430 PRINT"I=GREEN"
440 IF SH>500 THEN PRINT"2=YELLOW"
450 IF SH>1500 THEN PRINT"3=RED"
460 CD=GET#48:IF CD<1 OR CD>3 THEN 460
470 IF SH<500 THEN CD=1
480 IF SH<1500 AND CD=3 THEN CD=2
490 GOTO 130
500 CLS
510 PRINT"REPAIR CONTROL"
520 PRINT"-----"
530 PRINT"THE DAMAGE IS NOW"
540 I=1:REPEAT
550 D<(I)=D<(I)-RND(5):IF D<(I)<0 THEN D<(I)=0
560 IF D<(I)>0 THEN PROCDAM(I)
570 I=I+1:UNTIL I=11
580 TIME=TIME+RND(20)*1000:EN=EN-100
590 GOTO 130
600 CLS
610 PRINT"WARP DRIVE"
620 PRINT"-----"
630 PRINT"TO WHAT QUADRANT X,Y,Z?"
640 INPUT X,Y,Z
650 IF X<0 OR X>7 OR Y<0 OR Y>7 OR Z<0 OR Z>7 THEN 600
660 PRINT"WHAT SPEED (1 TO 9)?"
670 A=GET#48:IF A<1 OR A>9 THEN 670
680 D=(ABS(X-XQ)+ABS(Y-YQ)+ABS(Z-ZQ))
690 EN=EN-D*A
700 ZQ=INT(Z):YQ=INT(Y):XQ=INT(X)
710 TIME=TIME+1000*D/A
720 GOTO 130
730 CLS
740 PRINT"SHIP'S COMPUTER"
750 PRINT"-----"
760 PRINT"A DATA BASE SCAN FOR WHAT?"
770 PRINT"1=STARBASE"
780 PRINT"2=KLINGON"
790 PRINT"3=BLACK HOLE"
800 PRINT"4=M CLASS PLANET"
810 PRINT"5=G CLASS PLANET"
820 PRINT"6=STARFLEET HQ"
830 A=GET#48:IF A<1 OR A>6 THEN 830
840 IF A=3 THEN 930
850 IF A=6 THEN PRINT"STARFLEET HQ IS AT ";XQ;YQ;ZQ:GOTO 130
860 W=5:IF A=4 THEN W=10
870 IF A=5 THEN W=3
880 IF A=2 THEN W=-1
890 FOR Z=0 TO 7:FOR Y=0 TO 7:FOR X=0 TO 7
900 G=7*(Z*X+Y*8+Z*64)
910 IF G=W OR (G>20 AND G<41 AND A=2) THEN PRINT"AT";X;Y;Z
920 NEXT X,Y,Z:GOTO 130
930 IF 7*(Z*X+Y*8+Z*64)=17 THEN 130
940 PRINT"BLACK HOLE AT";XBL;YBL;ZBL:GOTO 130
950 CLS
960 PRINT"SCIENCE COMPUTER"
970 PRINT"-----"
980 Q=7*(Z*X+Y*8+Z*64)
990 PRINT"COMPUTER INDICATE"
1000 IF Q=7 OR Q=17 THEN PRINT"A BLACK HOLE":GOTO 130
1010 IF Q=6 OR Q=16 THEN PRINT"A STARBASE":GOTO 130
1020 IF Q=10 THEN PRINT"AN EXPLORED M CLASS PLANET":GOTO 130
1030 IF Q=9 THEN PRINT"AN EXPLORED G CLASS PLANET":GOTO 130
1040 IF Q=20 THEN PRINT"AN M CLASS PLANET"
1050 IF Q=19 THEN PRINT"A G CLASS PLANET"
1060 IF Q>40 THEN PRINT"A KLINGON WARSHIP!";7*(Z*X+Y*8+Z*64)=Q-20:GOTO 130
1070 IF Q=20 THEN PRINT"A KLINGON WARSHIP!":GOTO 130
1080 IF Q=0 THEN PRINT"VOID":GOTO 130
1090 7*(Z*X+Y*8+Z*64)=Q-10
1100 SC=SC+25:IF Q=20 THEN SC=SC+25:MP=MP-1 ELSE GP=GP-1
1110 PROCPLANET
1120 GOTO 130
1130 CLS
1140 PRINT"LONG RANGE SCAN"
1150 PRINT"-----"
1160 VDU 20,1,19,20,1
1170 CLS
1180 M=3:N=2
1190 PRINT";XQ-1";";XQ";";XQ+1"
1200 FOR Y=YQ-1 TO YQ+1
1210 PRINTTAB(0,N+1);Y;
1220 FOR X=XQ-1 TO XQ+1
1230 IF X<0 OR X>7 OR Y<0 OR Y>7 THEN B10="" :B20="" :B30="" :GOTO 130
1240 B10=CHR(146)+CHR(32)+CHR(248)+CHR(228)+CHR(32)
1250 B20=CHR(146)+CHR(42)+CHR(253)+CHR(244)+CHR(165)
1260 B30=CHR(146)+CHR(32)+CHR(162)+CHR(161)+CHR(32)
1270 Q=7*(Z*X+Y*8+Z*64)
1280 IF Q=0 THEN B10=CHR(140)+CHR(0) :B20=CHR(140)+CHR(0) :B30=CHR(140)+CHR(0)
1290 IF Q>10 AND Q<19 THEN Q=Q-10
1300 IF Q>40 THEN Q=Q-20
1310 7*(Z*X+Y*8+Z*64)=Q
1320 IF Q=7 THEN B10=CHR(151)+CHR(184)+CHR(185)+CHR(115)+CHR(836):B20=CHR(151)+CHR(106)
+CHR(42)+CHR(36)+CHR(119):B30=CHR(151)+CHR(41)+CHR(185)+CHR(172)+CHR(185)
1330 IF Q=5 THEN B10=CHR(145)+CHR(224)+CHR(32)+CHR(32)+CHR(32):B20=CHR(145)+CHR(186)+C
HR(248)+CHR(17)+CHR(52):B30=CHR(149)+CHR(171)+CHR(255)+CHR(255)+CHR(39)
1340 IF Q=20 THEN B10=CHR(150)+CHR(32)+CHR(180)+CHR(32)+CHR(180)+CHR(32)+CHR(180):B20=CHR(150)+CHR(32)
+CHR(170)+CHR(150)+CHR(32):B30=CHR(150)+CHR(32)+CHR(170)+CHR(175)+CHR(32)
1350 PRINTTAB(N,N);B10;CHR(135);
1360 PRINTTAB(N,N+1);B20;CHR(135);
1370 PRINTTAB(N,N+2);B30;CHR(135);
1380 M=M+5
1390 NEXT X
1400 M=3:N=N+4
1410 NEXT Y
1420 VDU29,22,17,30,3
1430 GOTO 130
1440 CLS
1450 PRINT"PHASER BANK"
1460 PRINT"-----"
1470 PRINT"PHASERS READY AND FIRING."
1480 Q=7*(Z*X+Y*8+Z*64):IF Q<21 THEN PRINT"WHAT ARE YOU FIRING AT?":GOTO 130
1490 PRINT"PRESS SPACE TO FIRE"
1500 REPEAT UNTIL GET#=""
1510 CLS
1520 VDU 20,1,19,20,1
1530 X1=5:Y1=9
1540 PROCK
1550 FOR Y=17 TO 12 STEP -1
1560 PRINTTAB(10-Y,Y);CHR(151);CHR(50);CHR(135);
1570 SOUND(1,17,Y,3,2

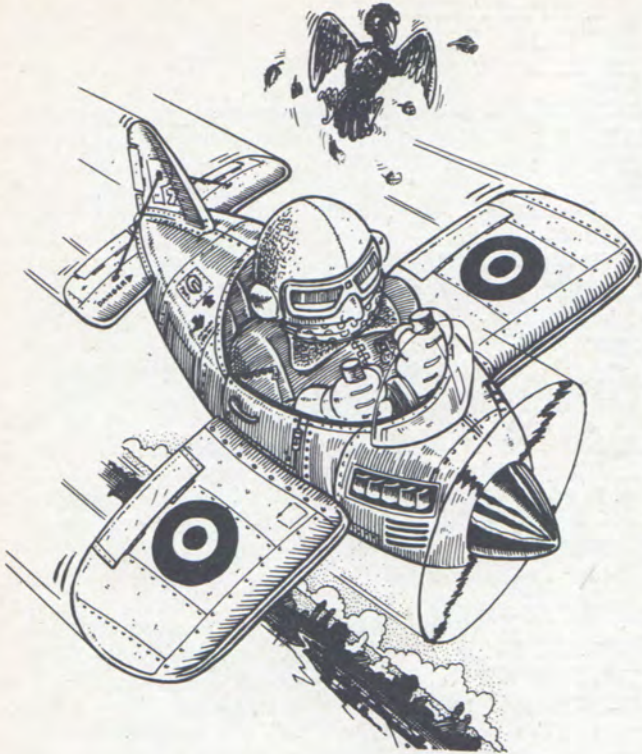
```



```
1598 PRINTTAB(Y,Y);CHR0151;CHR097;CHR0135;
1599 FOR I=0 TO 500:NEXT
1600 NEXT Y
1610 EN=EN-50
1620 CLS
1630 V=(Z%+XQ+YQ08+ZQ064):B=V-RND(5)-3
1640 IF B>21 OR (B>41 AND V>40) THEN 7(Z%+XQ+YQ08+ZQ064)=B:GOTO 130
1650 PROCXPLOSION
1660 SC=SC-RND(50)+20
1670 7(Z%+XQ+YQ08+ZQ064)=0
1680 KL=KL-1
1690 GOTO 130
1700 CLS
1710 PRINT"PHOTON TORPEDO"
1720 PRINT"-----"
1730 IF PH=0 THEN PRINT"YOU RINT GOT NONE LEFT!":GOTO 130
1740 Q=7(Z%+XQ+YQ08+ZQ064):IF Q<21 THEN PRINT"WHAT ARE YOU
0 130 FIRING AT ?":GOT
1750 PRINT"Press 'SPACE' to fire torpedo."
1760 PRINT"Press 'SPACE' to start sweep of sights."
1770 REPEAT UNTIL GET#=""
1780 CLS
1790 VDU 20,1,19,20,1
1800 KI=RND(5):Y1=RND(10)
1810 X=0:Y=0
1820 PROCK
1830 PRINTTAB(X,Y);"X";
1840 X=X+1:IF X=20 THEN X=0:Y=Y+1
1850 IF Y=20 THEN Y=0
1860 IF INKEY#(0)=" " THEN 1890
1870 GOTO 1820
1880 PH=PH-1
1890 FOR I=0 TO 20
1900 SOUND2,17,1/3,(20-I)*3,1
1910 NEXT
1920 IF X<1+5 OR X>1+5 OR Y<1+1 THEN 130
1930 PROCXPLOSION
1940 SC=SC-RND(50)+20
1950 7(Z%+XQ+YQ08+ZQ064)=0
1960 KL=KL-1
1970 CLS
1980 GOTO 130
1990 CLS
2000 PRINT"IMPULSE ENGINES"
2010 PRINT"-----"
2020 Q=7(Z%+XQ+YQ08+ZQ064)
2030 IF Q=10 OR Q=9 THEN PRINT"ORBIT SUSTAINED":SC=SC-RND(10):GOTO 130
2040 IF Q<6 AND Q>15 THEN PRINT"THERE IS NOTHING TO ORBIT":GOTO 130
2050 PRINT"PREPARE FOR DOCKING PROCEDURE"
2060 PRINT"USE THE CURSOR CONTROL KEYS TO MOVE YOUR SHUTTLE"
2070 PRINT"Press 'SPACE' to start."
2080 REPEAT UNTIL GET#=""
2090 XS=RND(10)+5:YS=0
2100 VDU 20,1,19,20,1
2110 #FX 4,1
2120 KI=RND(10)+5:Y1=RND(10)+5
2130 CLS
2140 PRINTTAB(X1,Y1);"O";TAB(XS,YS);"A";
2150 A=INKEY#(20):IF A=CHR#(88) THEN XS=XS-1
2160 IF A=CHR#(89) THEN XS=XS+1
2170 YS=YS+1
2180 IF YS=18 THEN 130
2190 IF XS<0 THEN XS=0
2200 IF XS>19 THEN XS=19
2210 IF XS=KI AND YS=Y1 THEN 2230
2220 EN=EN-1:GOTO 2130
2230 VDU 20,22,17,30,3
2240 CLS
2250 PRINT"DOCKED."
2260 EN=6000:PH=5:7(Z%+XQ+YQ08+ZQ064)=0
2270 SOUND 1,17,100,4
2280 IF XQ=HQ0 AND YQ=YHQ0 AND ZQ=ZHQ0 THEN 3500
2290 I=1
2300 D=RND(10)
2310 D<D=D<D>=5:IF D<D><0 THEN D<D>=0
2320 I=I+1:IF I<6 THEN GOTO 2300
2330 SH=SH+RND(500):IF SH>4000 THEN SH=4000
2340 GOTO 130
2350 DEFPROCSCREEN
2360 LOCAL I,B#
2370 PRINTSTRING(40,"#");
2380 PRINTTAB(21,2);STRING(19,"#");
2390 PRINTTAB(0,23);STRING(40,"#");
2400 PRINTTAB(0,20);STRING(22,"#");
2410 PRINTTAB(22,10);STRING(10,"#");
2420 FOR I=0 TO 23:PRINTTAB(0,I);"X";TAB(39,I);"X";
2430 IF I<=20 THEN PRINTTAB(21,I);"X";
2440 NEXT I
2450 PRINTTAB(1,21);"SCORE=";SC;TAB(13,21);"ENERGY=";EN; " ";TAB(26,21);"TORPEDO
S=";PH;
2460 PRINTTAB(1,22);"SHIELD=";SH; "QUADRANT=";XQ;YQ;ZQ; "KL=";KL; "PL=";MP;GP;"
";
2470 B=CHR#(130)+GREEN+CHR#(135):IF CD=2 THEN B=CHR#(131)+YELLOW"
2480 IF CD=3 THEN B=CHR#(136)+CHR#(129)+RED+CHR#135
2490 PRINTTAB(22,19);"CONDITION=";B;
2500 PRINTTAB(22,20);"STARDATE=";INT(TIME/1000);
2510 PRINTTAB(22,1);"COMMAND=";
2520 IF 7(Z%+XQ+YQ08+ZQ064)>20 THEN PROCKLINGON
2530 D=RND(10):D<D=D<D>=RND(5):IF D<D><0 THEN D<D>=0
2540 EN=EN-1-CD*50+50:IF EN<0 THEN 3400
2550 G=ABS(XQ-XBL)+ABS(YQ-YBL)+ABS(ZQ-ZBL)
2560 IF RND(0.85)=1 THEN PROCBUCK
2570 IF XQ=XBL AND YQ=YBL AND ZQ=ZBL THEN 3490
2580 ENDPROC
2590 DIM Z% 512,D%(10)
2600 FOR I=0 TO 512:Z%I=0:NEXT
2610 KL=20+RND(10):GP=RND(10)+5:MP=RND(10)+5
2620 FOR I=1 TO KL:PROCRRND:7(Z%+X+Y+Z+264)=40+RND(20):NEXT
2630 FOR I=1 TO MP:PROCRRND:7(Z%+X+Y+Z+264)=20:NEXT
2640 FOR I=1 TO GP:PROCRRND:7(Z%+X+Y+Z+264)=19:NEXT
2650 FOR I=1 TO RND(5)+2:PROCRRND:7(Z%+X+Y+Z+264)=16:NEXT
2660 HQ=X:YHQ=Z
2670 PROCRRND:7(Z%+X+Y+Z+264)=17:XBL=X:YBL=Y:ZBL=Z
2680 RETURN
2690 DEFPROCRRND
2700 X=RND(8)-1:Y=RND(8)-1:Z=RND(8)-1
2710 IF 7(Z%+X+Y+Z+264)>0 THEN 2700
2720 ENDPROC
2730 DEFPROCRRND
2740 LOCAL I,B#
2750 RESTORE 2760
2760 DATA DAMAGE CONTROL,ALERT,REPAIR,WARP DRIVE,IMPULSE ENGINES,PHASER BANK,PH
JTON TORP.,SCIENCE COMP.,SHIP COMP.,LONG SCAN
2770 FOR I=1 TO D:READ B#:NEXT
2780 PRINTB#;" ";D%(D)
2790 ENDPROC
2800 DEFPROCPLANET
2810 LOCAL H,S,A,M,T,B#,I
2820 H=RND(20)-1:S=RND(20)-.5:#100:A=RND(5):W=RND(100):T=RND(10)
2830 PRINT"LIFE=";H;" BILLION"
2840 PRINT"RADIUS=";S;" K"
```

```
2850 RESTORE 2860
2860 DATA CORROSIVE,DENSE,NORMAL,THIN,NONE
2870 FOR I=1 TO A:READ B#:NEXT I
2880 PRINT"ATMOSPHERE:";B#
2890 PRINT"W:";W;" WATER"
2900 PRINT"TECH LEVEL=";T
2910 SC=SC+T-8+H-10
2920 ENDPROC
2930 DEFPROCXPLOSION
2940 PRINTTAB(X1+4,Y1+2);CHR0145;CHR0255;CHR0255;CHR0135
2950 SOUND0,1,1,10
2960 SOUND0,1,2,10
2970 SOUND0,1,3,10
2980 FOR I=0 TO 1000:NEXT
2990 PRINTTAB(X1+3,Y1);
3000 VDU 150,114,170,36,50,135
3010 PRINTTAB(X1+2,Y1+1);
3020 VDU 150,164,110,189,168,135
3030 PRINTTAB(X1+3,Y1+2);
3040 VDU 150,50,104,52,57,135
3050 FOR I=0 TO 1000:NEXT
3060 PRINTTAB(X1+1,Y1);:VDU 147,33,36,49,96,40,135
3070 PRINTTAB(X1+1,Y1+1);:VDU 147,33,225,241,50,160,135
3080 PRINTTAB(X1+1,Y1+2);:VDU 147,96,250,55,36,49,135
3090 PRINTTAB(X1+1,Y1+3);:VDU 147,50,166,36,225,100,135
3100 FOR I=0 TO 1000:NEXT
3110 CLS
3120 ENDPROC
3130 DEFPROCCLINGON
3140 Q=7(Z%+XQ+YQ08+ZQ064)
3150 IF Q<41 THEN W=(Q-20)*5 ELSE W=(Q-40)*5
3160 VDU 20,1,19,20,1
3170 X1=5:Y1=0
3180 PROCK
3190 FOR Y1=4 TO 18 STEP 2
3200 PRINTTAB(9,Y1);
3210 VDU 147,53,135
3220 SOUND3,1,(20-Y1)*3,2
3230 FOR KI=0 TO 200:NEXT
3240 NEXT Y1
3250 VDU 20,22,17,30,3
3260 CLS
3270 IF RND(5)<3 THEN ENDPROC
3280 IF SH<0 THEN 3430
3290 IF W>RND(50*CD) THEN 3350
3300 PRINT"YOU HAVE SUFFEREDA MINOR HIT"
3310 SH=SH-RND(50)
3320 IF SH<0 THEN 3430
3330 D=RND(10):D<D>=D<D>+1
3340 ENDPROC
3350 PRINT"YOU HAVE SUFFEREDA MAJOR HIT"
3360 SH=SH-RND(300)
3370 LOCAL I
3380 I=1
3390 D=RND(10)
3400 D<D>=D<D>+RND(3)
3410 I=I+1:IF I<6 THEN 3390
3420 ENDPROC
3430 B#="What's doing battle against the klingons"
3440 VDU 25,CL#
3450 PRINT" B# " on stardate "+STRINT(TIME/1000)+", "+B#+", the enterprise
& crew were lost in space."
3460 PROCRESK(M#)
3470 GOTO 3640
3480 B#="after depleting its energy supply":GOTO 3440
3490 B#="after flying into a black hole":GOTO 3440
3500 VDU 25:CLS
3510 M#=" Admiral Fitzpatrick here... captain, after studying your log and da
ta tapes, and taking into account that you were "
3520 IF TIME/1000000 THEN M#="on time":SC=SC*1.5 ELSE M#="late":SC=SC*1.5
3530 M#="and that you "
3540 IF KL=0 AND MP=0 AND GP=0 THEN M#="completed":SC=SC*2 ELSE M#="faile
d":SC=SC/2
3550 M#=" your mission."
3560 PROCRESK(M#)
3570 PRINT"
3580 M#="I'm going to recommend to starfleet that you be "
3590 RESTORE 3600
3600 DATA PROMOTED,2000,DECORATED,1500,REASSIGNED,1000,DEMOTED,500,RESIGNED,250
,EXECUTED,-1
3610 READ B#,S:IF SC<S THEN 3610
3620 M#="B#+" after getting a total score of "+STRINTSC
3630 PROCRESK(M#)
3640 PRINT"
3650 PRINT"Another game (y/n) ?"
3660 A=GET#:IF A="Y" THEN RUN
3670 IF A="N" THEN END
3680 GOTO 3660
3690 IF ERR=17 THEN GOTO 130
3700 CLS
3710 PRINT""
3720 REPORT
3730 PRINT" in line ";ERL
3740 END
3750 DEFPROCBUCK
3760 XQ=XQ-(XQ(XBL)+X(XQ)XBL)
3770 YQ=YQ-(YQ(YBL)+Y(YQ)YBL)
3780 ZQ=ZQ-(ZQ(ZBL)+Z(ZQ)ZBL)
3790 VDU 20,22,17,30,3
3800 FOR I1=0 TO 1000:NEXT:CLS
3810 PRINT"WARNING:"
3820 PRINT"-----"
3830 PRINT" the enterprise is being drawn""through space""towards a black""
hole."
3840 ENDPROC
3850 DEFPROCCK
3860 CLS
3870 PRINTTAB(X1,Y1);
3880 VDU 151,163,41,48,32,53,32,56,163,33,135
3890 PRINTTAB(X1+2,Y1+1);
3900 VDU 151,234,243,119,251,135
3910 PRINTTAB(X1+1,Y1+2);
3920 VDU 151,40,33,34,163,32,41,135
3930 ENDPROC
3940 DEFPROCRESK(M#)
3950 LOCAL X%,I%,J%
3960 @LAB0=" "+M#
3970 X%=0
3980 I%=1
3990 J%=@LAB0
4000 IF J%I%=32 OR J%I%=13 THEN 4020
4010 I%=I%+1:GOTO 4000
4020 IF X%I%>39 THEN PRINT:X%=0
4030 FOR U%=1 TO I%
4040 VDU J%I%U%
4050 NEXT
4060 J%=J%+1%
4070 X%=@LAB0
4080 I%=1
4090 IF J%&@LAB0=@LENA# THEN ENDPROC
4100 GOTO 4000
```

# ΠΤΗΣΗ ΑΥΤΟΚΤΟΝΙΑΣ



Το πρόγραμμα αυτό για τον BBC model B, είναι το γνωστό "BLITZ" με μερικά επιπλέον χαρακτηριστικά στην παρουσίασή του. Όπως και στο πρωτότυπο παιχνίδι βρισκόμαστε μέσα σ'ένα βομβαρδιστικό το οποίο χάνει συνέχεια ύψος πάνω από μια εχθρική πόλη. Ο μόνος τρόπος να το προσεγγίσετε με ασφάλεια είναι να ισοπεδώσετε με βομβαρδισμό την πόλη δημιουργώντας έτσι μια επίπεδη επιφάνεια για να χρησιμοποιήσει σαν διάδρομος προσγείωσης. Οι βόμβες ρίχνονται με το πάτημα του "SPACE", και είναι ο μόνος χειρισμός που μπορείτε να κάνετε. Αν βρείτε το παιχνίδι πολύ γρήγορα ή πολύ αργό, μπορείτε να αλλάξετε την ταχύτητα πτήσεως, αλλάζοντας την τελική τιμή του βρόχου καθυστέρησης στη γραμμή 35θ. Συνολικά το πρόγραμμα δουλεύει ως εξής: 3θ προγραμματίζει το

fθ για να τρέχει ξανά το πρόγραμμα

4θ-14θ ορισμός χαρακτήρων

15θ ήχος βόμβας

16θ, 17θ οδηγίες

18θ ορίζει την βόμβα

19θ-30θ στήσιμο οθόνης

31θ αρχικές τιμές μεταβλητών

32θ-42θ κυρίως παιχνίδι

35θ-55θ procedure οδηγιών

56θ-60θ procedure καταστροφής

61θ-63θ procedure βόμβας

64θ-66θ procedure γκρεμίσματος κτιρίων

67θ-72θ αρχικές τιμές μεταβλητών

73θ-74θ υπολογισμός θέσης

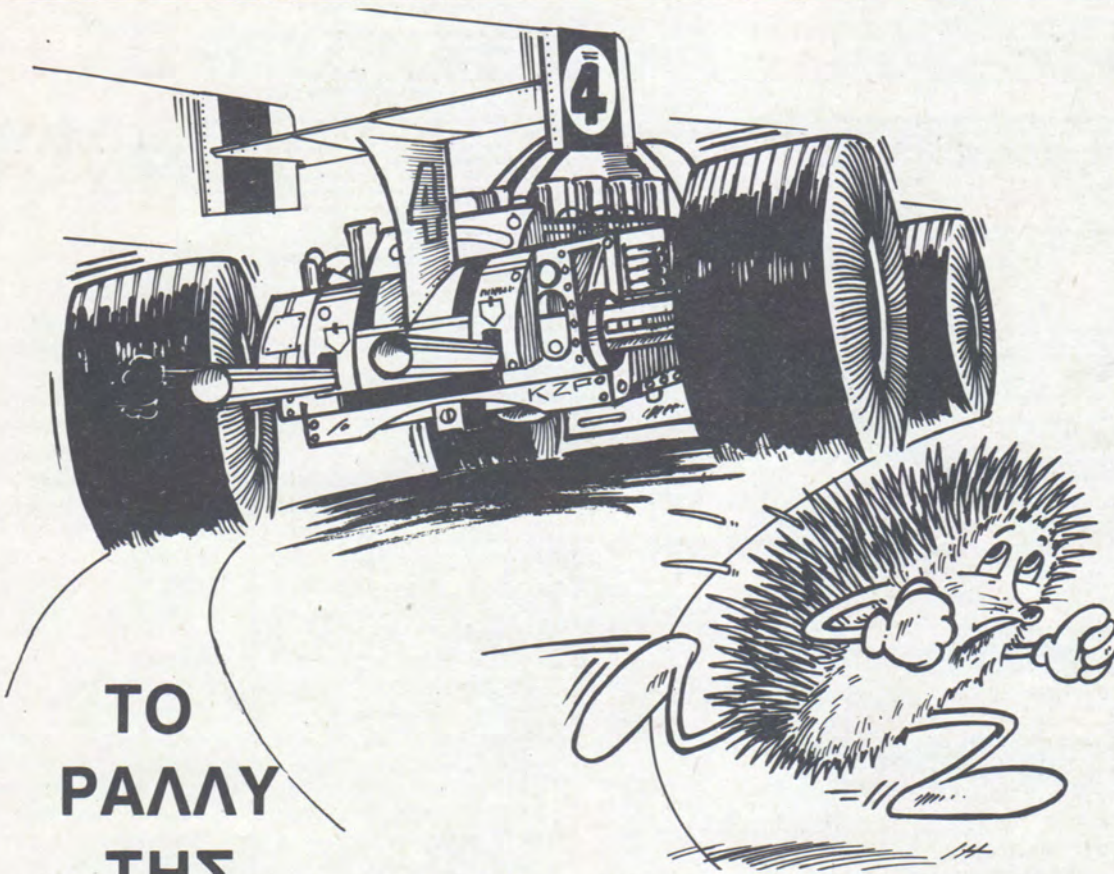
75θ-79θ procedure προσγείωσης

80θ-87θ procedure σύγκρουσης

```

30 #KEY0"RUN!"
40 VDU23,224,90,126,90,126,90,126,90,126
50 VDU23,225,90,126,90,90,126,90,90,126
60 VDU23,225,102,126,102,126,102,126,102,126
70 VDU23,227,0,0,24,24,24,36,126,90,126
80 VDU23,220,60,60,24,24,60,90,126,90
90 VDU23,229,24,24,24,24,60,126,102,126
100 VDU23,230,0,32,112,240,252,127,63,0
110 VDU23,231,0,0,0,1,241,255,253,1
120 VDU23,232,126,60,24,60,126,126,60,24
130 VDU23,233,32,124,254,127,63,31,31,31
140 VDU23,234,0,0,4,102,249,240,252,252
150 ENVELOPE1,1,11,-6,1,10,30,60,127,0,0,-127,126,0
160 MODE7
170 PROCINSTRUCTIONS
180 IFDL=1 THEN DESTRUCT@=" "+CHR0(10)+CHR0(9)+" " ELSE DESTRUCT@=" "+CHR0(10)
+CHR0(9)+" "+CHR0(10)+CHR0(9)+" "
190 MODE2
200 COLOUR134:CLS
210 VDU23:10,32;0;0;0;
220 FOR BUILD%2 TO 10
230 COLOUR0
240 A=RND(3)+223
250 FOR HEIGHT%29 TO RND(C%4)+20-C%2 STEP -1
260 PRINTTAB(BUILD%,HEIGHT%);CHR0(A)
270 NEXT HEIGHT%
280 PRINTTAB(BUILD%,HEIGHT%+1);CHR0(A+3)
290 SOUND1,-15,RND(200),1
300 NEXT BUILD%
310 PROCINIT
320 FIRE@=INKEY0):IF FIRE@=" " THEN PROCBOMB
330 COLOUR1:PRINTTAB(X,Y);AER0@:SOUND0,-5,100,2
340 VDU26:IF POINT(X+2)*64+32,(30-Y)*32)=0 AND POINT(X+3)*64+32,(30-Y)*32)=0
AND POINT(X+4)*64+32,(30-Y)*32)=0 THEN PROCCLAND
350 FOR SPEED%1 TO 200:NEXT SPEED
360 IF POINT(X+4)*64+32,(30-Y)*32)=0 THEN PROCRCRASH:PROCDROP
370 PROCMOVE
380 IF FIRE=0 THEN 320
390 COLOUR3:PRINTTAB(XB,YB);BOMB@
400 YB=YB+1:IF YB>20 OR POINT(XB*64+32,(30-YB)*32)=0 THEN PROCDESTRUCT ELSE 33
410 #FX15,0
420 GOTO 320
430 DEFPROCINSTRUCTIONS
440 PRINTTAB(13,10);CHR0(141)+CHR0(132);"BLITZ"
450 PRINTTAB(13,11);CHR0(141)+CHR0(132);"BLITZ"
460 PRINTTAB(0,13);CHR0(135);"YOU MUST DESTROY THE BUILDINGS FLAT"
470 PRINTTAB(3,14);CHR0(135);"ENOUGH TO LAND YOUR AEROPLANE"
480 PRINTTAB(5,10);CHR0(135);"PRESS SPACE BAR TO DROP BOMB"
490 INPUT" INPUT DESTRUCT LEVEL (1 TO 2)";DL
500 IF DL<1 OR DL>2 THEN 490
510 INPUT" INPUT DIFFICULTY LEVEL (1 TO 5)";C
520 IF C<1 OR C>5 THEN 510
530 PRINTTAB(5,21);"PRESS F0 TO RESTART"
540 PRINTTAB(5,22);"PRESS ANY KEY TO START":ST0=GET0
550 ENDPROC
560 DEFPROCCRASH:SOUND0,-15,100,10
570 FOR CRASH%1 TO 10
580 FOR RIGHT%1 TO 10:VDU23:13,RIGHT,0;0;0;:NEXT RIGHT
590 FOR LEFT%10 TO 1 STEP-1:VDU23:13,LEFT,0;0;0;:NEXT LEFT
600 NEXT CRASH:ENDPROC
610 DEFPROCBOMB
620 FIRE=1:XB=X+2:YB=Y:IF XB>=20 THEN PROCDESTRUCT
630 ENDPROC
640 DEFPROCDESTRUCT
650 SOUND1,1,100,2:FIRE=0:PRINTTAB(XB,YB);DESTRUCT@:SOUND0,-5,100,2
660 ENDPROC
670 DEFPROCINIT:X=1:Y=3:FIRE=0
680 BOMB@=" "+CHR0(10)+CHR0(9)+CHR0(232)
690 AER0@=" "+CHR0(230)+CHR0(231)
700 DROP@=" "+CHR0(10)+CHR0(9)+CHR0(233)
710 DROPT@=" "+CHR0(10)+CHR0(9)+CHR0(234)
720 ENDPROC
730 DEFPROCMOVE:X=X+1:IF X>=20 THEN X=0:Y=Y+1
740 ENDPROC
750 DEFPROCLAND
760 PRINTTAB(1,2);"HELL DONE":RESTORE700:FOR MUSIC%1 TO 10:READ A,B
770 SOUND2,-10,A,B:NEXT MUSIC:END:ENDPROC
780 DATA 129,10,117,5,121,5,129,10,101,10
790 DATA 121,5,129,5,137,5,145,5,149,5
800 DEFPROCDROP:XD=X+2:XDT=X+3:YD=Y-2:YDT=Y-2
810 FOR DELAY%1 TO 300:NEXT DELAY
820 YD=YD+1:PRINTTAB(XD,YD);DROPT@
830 VDU26
840 IF POINT(XD*64+32,(29-YD)*32)=0 OR YD>=20 THEN 850 ELSE 820
850 YDT=YDT+1:PRINTTAB(XDT,YDT);DROPT@
860 VDU26
870 IF POINT(XDT*64+32,(29-YDT)*32)=0 OR YDT>=20 THEN END ELSE 850:ENDPROC

```



# ΤΟ ΡΑΛΛΥ ΤΗΣ ΚΑΤΑΣΤΡΟΦΗΣ

Άλλο ένα κλασικό παιχνίδι, σας παρουσιάζουμε εδώ σε μια παραλλαγή για τον BBC. Ο κομπιούτερ θα σχεδιάσει μια τετραγωνισμένη πίστα και θα εμφανιστούν δυο αυτοκίνητα. Εσείς ελέγχετε το πράσινο και ο κομπιούτερ το κόκκινο. Το αυτοκίνητο του κομπιούτερ θα κινείται στην πίστα με αντίθετη φορά από την δική σας και με έναν καθόλου φιλικό σκοπό... θέλει

να σας τρακάρει όσο το δυνατόν γρηγορότερα, για να μην μεγαλώσετε το σκορ σας, τρώγοντας τις κουκίδες που βρίσκονται στις λωρίδες. Για να το αποφύγετε πρέπει να αλλάξετε λωρίδα στις διασταυρώσεις, πατώντας το "A" για να μπειτε σε πιο κεντρική λωρίδα ή το "D" για εξωτερική. Έχετε μόνον τρεις ευκαιρίες κι' έτσι φροντίστε να είστε προσεκτικοί!

```

10 high=0
20 MODE7:PROCINSTRUCT
30 MODE2:PROCSETUP
40 SC=0:ENVELOPE1,3,0,0,0,0,0,0,121,-10,-5,-2,120,120
50 M=0:ON ERROR MODE7:END
60 VDU 23,1,0,0,0,0
70 GOSUB 150:GOSUB 150
80 PROCDRAW:PROCFILLING
90 PROCMOVE1
100 IF YD%OY AND YR%OX THEN 1040
110 PROCMOVE2
120 IF YD%OY AND YR%OX THEN 1040
130 GOTO 90
140 END
150 VDU 23,250,0,230,60,255,255,60,230,0
160 VDU 23,251,24,90,125,90,24,90,126,90
170 VDU 23,126,255,255,255,255,255,255,255,255
180 YD%=15:YR%=7:CAR2=CHR#251
190 OX=2:OY=2:CAR2=CHR#251
200 MX=1:MY=0:HM=0:VM=1
210 RETURN
220 DEFPROCDRAW
230 COLOURbackcol:COLOURforecol
240 CLS
250 PRINT " SCORE:"
260 PRINT "-----"
270 PRINT ""
280 PRINT ""
290 PRINT ""
300 PRINT ""
310 PRINT ""
320 PRINT ""
330 PRINT ""
340 PRINT ""
350 PRINT ""
360 PRINT ""
370 PRINT ""
380 PRINT ""
390 PRINT ""
400 PRINT ""
410 PRINT ""
420 PRINT ""
430 PRINT ""
440 PRINT ""
450 PRINT ""
460 PRINT ""
470 PRINT ""
480 PRINT ""
490 PRINT ""
500 PRINT ""
510 PRINT ""
520 PRINT ""

```

```

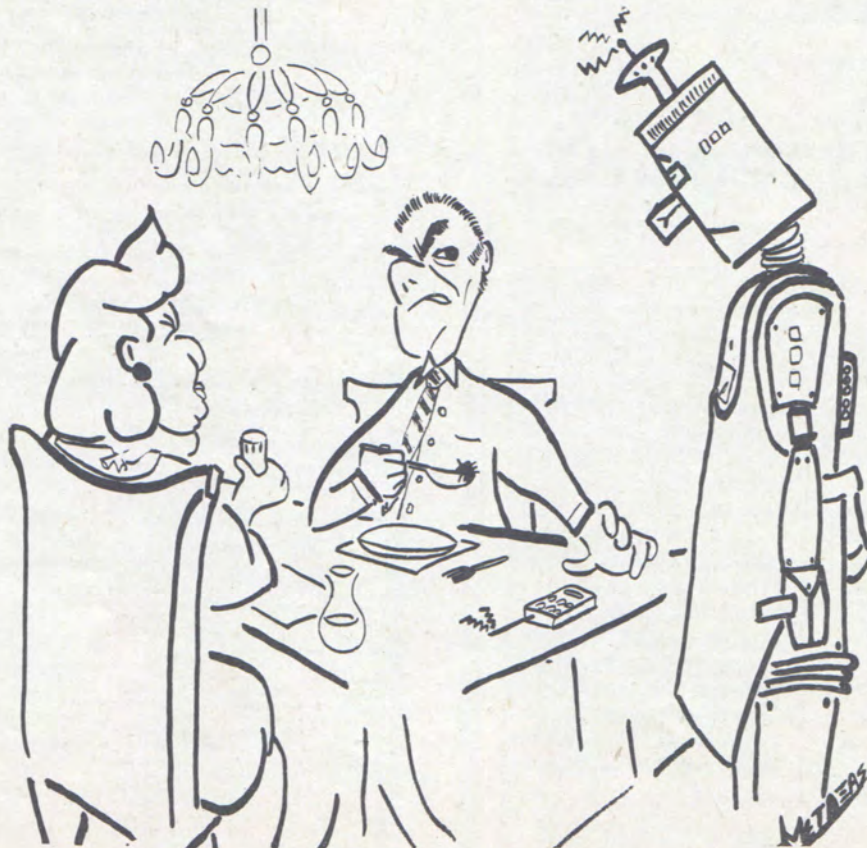
530 PRINT"*****"
540 ENDPROC
550 DEFPROC MOVE1
560 A=7236:M=0:FX15,0
570 PRINTTAB(YA,X,YD%)" ":IF A=0 THEN 610
580 M=(SGN(M)<0)-(SGN(VM)<0)>2*(A=170)-(A=193)
590 IF FNREADCH(YA+VM*SGN(M),YD+HM*SGN(M))<>" " AND FNREADCH(YA+VM*M,YD+HM*
M)*" " THEN YA=YA+VM*HM:YD=YD+HM*HM:IF FNREADCH(YA-VM*M/2,YD-HM*M/2)" " T
HEN PRINTTAB(YA-VM*M/2,YD-HM*M/2)" " :S=SC+score:COLURscr:COL:PRINTTAB(0,0);S
C
600 REM
610 REM
620 TY=YD+VM:TX=YA+HM:AB=FNREADCH(TX,TY)
630 IF AB="." THEN SC=SC+score:COLURscr:COL:PRINTTAB(0,0);SC
640 IF AB<"." AND AB>"." PROCCHNG
650 YD=YD+VM:YA=YA+HM:SOUND1,1,100,1
660 COLURscr:COL:PRINTTAB(YA,YD);CARB
670 ENDPROC
680 DEFPROCCHNG
690 IF VM=1 AND HM=0 THEN VM=0:HM=1:CARB=CHR(250):ENDPROC
700 IF VM=0 AND HM=1 THEN VM=-1:HM=0:CARB=CHR(251):ENDPROC
710 IF VM=-1 AND HM=0 THEN VM=0:HM=-1:CARB=CHR(250):ENDPROC
720 IF VM=0 AND HM=-1 THEN VM=1:HM=0:CARB=CHR(251):ENDPROC
730 DEFPROCCHNG(X,Y)
740 LOCAL A%,LASTX,LASTY,C
750 LASTX=POS(LASTY+VPOS
760 VDU 31,X,Y
770 A%=135:C=USR( &FFF4) AND &FFF
780 C=C DIV &100
790 VDU 31,LASTX,LASTY
800 =CHR(C)
810 DEFPROC MOVE2
820 TX=MX+OX:TY=MY+OY
830 AB=FNREADCH(TX,TY)
840 IF AB="." THEN PROCCHNG
850 COLURdotcol:PRINTTAB(OX,OY);" "
860 COLURmycol:PRINTTAB(OX,OY);" " :SOUND1,-8,50,1
870 OX=TX:OY=TY:PROCCHNG(LANE
880 ENDPROC
890 DEFPROCCHNG
900 IF MX=0 AND MY=1 THEN MX=-1:MY=0:CARB=CHR(250):GOTO 940
910 IF MX=-1 AND MY=0 THEN MX=0:MY=-1:CARB=CHR(251):GOTO 940
920 IF MX=0 AND MY=-1 THEN MX=1:MY=0:CARB=CHR(250):GOTO 940
930 IF MX=1 AND MY=0 THEN MX=0:MY=1:CARB=CHR(251):GOTO 940
940 TX=MX+OX:TY=MY+OY:ENDPROC
950 DEFPROCCHNG(LANE
960 DI=RND(3)-2:IF DI=0 THEN 960
970 AB=FNREADCH(OX+MY*DI,OY+MX*DI)
980 IF AB="." THEN ENDPROC
990 AB=FNREADCH(OX+MY*DI*2,OY+MX*DI*2)
1000 IF AB="." THEN ENDPROC
1010 COLURdotcol:PRINTTAB(OX,OY);" "
1020 OX=OX+MY*DI*2:OY=OY+MX*DI*2
1030 COLURmycol:PRINTTAB(OX,OY);CARB:ENDPROC
1040 REM
1050 VDU 23,240,0,0,24,24,0,0
1060 VDU 23,241,0,0,60,36,60,0,0
1070 VDU 23,242,0,126,66,66,66,126,0
1080 VDU 23,243,255,129,129,129,129,255
1090 X=OX:Y=OY
1100 FOR T=1 TO 45
1110 VDU 5:GCOLB,ENDCOL
1120 MOVE X#64,1020-Y#32
1130 IF T AND 1<>0 THEN PRINTCHR(240):MOVE X#64,1020-Y#32
1140 IF T AND 2<>0 THEN PRINTCHR(241):MOVE X#64,1020-Y#32
1150 IF T AND 4<>0 THEN PRINTCHR(242):MOVE X#64,1020-Y#32
1160 IF T AND 8<>0 THEN PRINTCHR(243):MOVE X#64,1020-Y#32

```

```

1170 VDU 4:PRINTTAB(OX,OY);" " :SOUND1,1,255-T#255/45,1:NEXT
1180 CLS:#FX15,0
1190 MODE7:PRINTTAB(5,10);"You scored ";SC
1200 IF high=SC THEN high=SC
1210 PRINTTAB(5,11);"High score:";high
1220 PRINT" Do you want another go?"
1230 AB=GETB
1240 IF AB="Y" THEN 30
1250 IF AB="N" THEN CLS:END
1260 PRINTTAB(5,13);"Just a simple Y or N is sufficient.":GOTO 1230
1270 DEFPROC FILLING
1280 RESTORE 1440:GCOLB,dotcol:VDU 5
1290 MOVE 0,up
1300 FOR T=1 TO 26:READ AB:PRINT " ";AB:NEXT
1310 VDU 4:ENDPROC
1320 DEFPROC INSTRUCT
1330 PRINTTAB(13);CHR(141);"DODGEMS"
1340 PRINTTAB(13);CHR(141);"DODGEMS"
1350 PRINTTAB(14);"*****"
1360 PRINTTAB(3);"Welcome to the game of dodgms."
1370 PRINT" In this you have a picture of a square race track. There are also tw
o cars the red car and the green car. The only control you have over your c
ar (the green one) is that of either moving in to a smaller"
1380 PRINT" track or out to a larger one. The red car goes around the track in an
opposite direction to you. To avoid it you must use these controls:"
1390 PRINT" A-Move in."
1400 PRINT" D-Move out."
1410 PRINT" Happy dodgming!!"
1420 PRINT" Press SPACE to start"
1430 REPEAT AB=GETB:UNTIL AB=" ":ENDPROC
1440 DATA.....
1450 DATA.....
1460 DATA.....
1470 DATA.....
1480 DATA.....
1490 DATA.....
1500 DATA.....
1510 DATA.....
1520 DATA.....
1530 DATA.....
1540 DATA.....
1550 DATA.....
1560 DATA.....
1570 DATA.....
1580 DATA.....
1590 DATA.....
1600 DATA.....
1610 DATA.....
1620 DATA.....
1630 DATA.....
1640 DATA.....
1650 DATA.....
1660 DATA.....
1670 DATA.....
1680 DATA.....
1690 DATA.....
1700 DEFPROC SETUP
1710 up=999
1720 backcol=132
1730 forecol=6
1740 yourcol=2
1750 dotcol=3
1760 mycol=1
1770 scr:col=1
1780 score=10
1790 endcol=3
1800 ENDPROC

```



EINAI TO ΔΕΥΤΕΡΟ ΤΣΙΠ ΜΟΥ ΒΡΙΣΚΟ ΣΤΗ ΣΟΥΠΑ ΜΟΥ!

# ΤΟ ΜΕΓΑΛΟ ΦΑΓΟΠΟΤΙ

Ο τίτλος που δώσαμε στο παιχνίδι αυτό δέχεται δυο διαφορετικές ερμηνείες. Αν ελάσσετε αρκετά επιδέξιως θα τρώτε συνείδηση ολο και περισσότερες κουκίδες και νέες πίστες, ενώ ο αρχάριος θα γίνει σύντομα μεζές για ένα από τα πελνασμένα φαντάσματα... Το παιχνίδι είναι πολύ ευχάριστο και εθιστικό, είναι δε λίγο διαφορετικό από το πρωτότυπο arcade game (packman). Η διαφορά είναι ότι υπάρχουν δυο μόνον φαντάσματα, που μπορούν όμως να μετακινούνται και μέσα από τους τοίχους. Τρώγοντας ένα από τα χάπια ενέργειας τα φαντάσματα ακινητοποιούνται για μερικά δευτερόλεπτα, ώστε να μπορέσετε να φάτε κι άλλες κουκίδες για να προχωρήσετε σε νέα πίστα. Σε καμιά περίπτωση δεν μπορείτε να φάτε εσείς τα φαντάσματα. Κάθε φορά που τελειώνετε μια πίστα, παίρνετε επιπλέον βαθμούς.

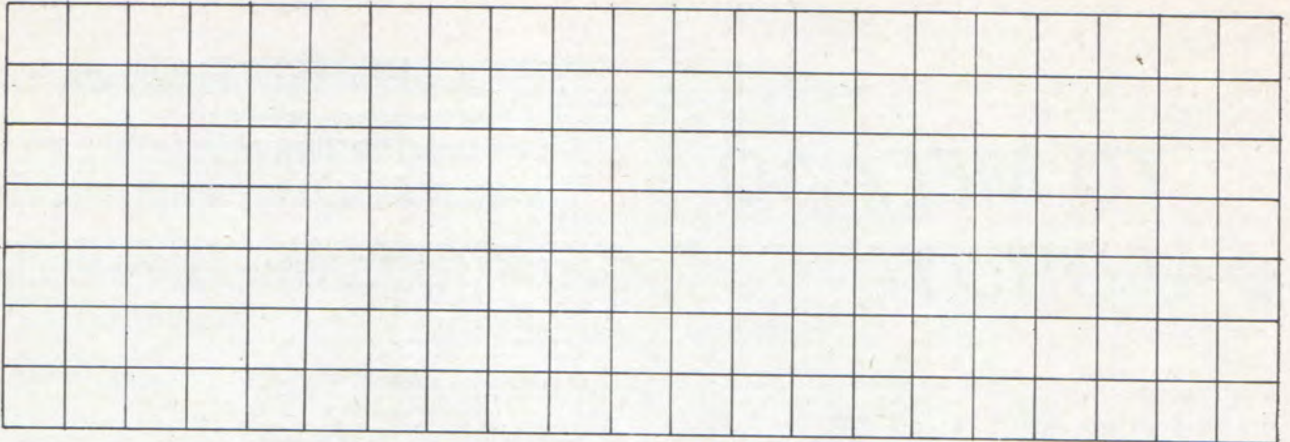
Ανά 5 πίστες υπάρχει και μια άρατη που προσθέτει δυσκολία στο παιχνίδι. Εδώ πρέπει να θυμόμαστε που είναι οι διάδρομοι για να τους ακολουθείτε. Σε τέτοιες πίστες, μόλις φαγωθεί ένα χάπι ενέργειας, οι διάδρομοι γίνονται ορατοί για λίγο. Τα πλήκτρα κίνησης είναι: Α επάνω, Ζ κάτω, Ν αριστερά και Μ δεξιά. Κάθε κουκίδα που τρώτε δίνει 100 πόντους και κάθε χάπι 100 πόντους.

Έξτρα πόντοι δίνονται με την συμπλήρωση κάθε πίστας, με το να πολλαπλασιάζεται ο αριθμός της πίστας επί 1000 Π.χ. με το τελευταίο της έκτης πίστας παίρνετε 6000 πόντους, στην 12η 12000 πόντους κ.ο.κ. Ανά 10000 πόντους κερδίζετε μια "ζωή" ακόμη, ενώ στην αρχή έχετε τρεις "ζωές".

Στην αρχή τα φαντάσματα κινούνται αργά, σχετικά με εσάς και το παιχνίδι είναι εύκολο. Σε κάθε νέα πίστα όμως η ταχύτητά τους μεγαλώνει, ώπου στην 11η πίστα γίνεται ίση με το μισό της δικής σας και παύει να αυξάνεται. Μέχρι να φτάσετε όμως εκεί, θα θυμηθείτε αρκετές φορές τον τίτλο του παιχνιδιού... Σας συμβουλεύουμε να τρώτε όσο το δυνατόν περισσότερες κουκίδες χωρίς να πάρετε χάπι. Στις πρώτες πίστες ξεκινείτε από το κέντρο, που είναι πιο δύσκολο. Όταν, αργότερα, τα φαντάσματα τρέχουν γρήγορα, καθαρίστε πρώτα τις κουκίδες στην περιφέρεια κρτώντας τα χάπια για να σας βοηθήσουν στο καθάρισμα του κέντρου.

Μην αφήνετε λίγες κουκίδες, εδώ κι εκεί, αλλά προσπαθείτε να καθαρίζετε με την μια μεγάλες περιοχές. Τέλος μην διατάζετε και ούτε να πανικοβάλλεστε γιατί είναι εύκολο να γλυτώσετε στις περισσότερες περιπτώσεις. Καλή όρεξη!

```
9 DIMGX(6),GY(6),GX(6),SC(20,20):HSCN:=0:HB="NOBODY"
10 ENVELOPE1,1,1,1,-1,5,5,23,127,-4,0,-2,127,60:ENVELOPE2,1,10,20,-30,10,5,15
127,-2,0,-1,127,100:MODE7:PROCINSTR:MODE7:VDU 23:BG2:0,0,0:SC1:=0:LC:=1:SCX:=0
:MAN:=0:SHEET:=1
11 LIFE:=3:NG:=1:DGX:=0:DGX:=0:PROCHAZE:PROCBETUP
12 PROCMEN
13 COLOUR 2:PRINTTAB(0,0):"SHEET ";SHEETX:COLOUR7:PRINTTAB(0,1):"SCORE ";SCX
14 #FX 11 0
15 #FX 12 1
16 FOR VX=0 TO NGX:FOR NX=1 TO 12-SHEETX:PROCMAN:IF MANX=1 PROCSPURT
17 NEXT:PROCHOST:NEXT:IF LIFE:=0 GOTO 62
18 GOTO 16
19 DEFPROCBETUP
20 FLAX:=0:GHX:=254:DX:=0:DY:=0:FOR VX=0 TO NGX:GX(VX)=12-VX:GY(VX)=12:GX(VX
)=247:NEXT:GX(NGX+1)=12:GY(NGX+1)=12
21 VDU 23,254,60,126,15,7,7,15,126,60,23,253,60,126,240,224,224,240,126,60,23
,252,0,66,195,195,231,255,126,60,23,251,60,126,255,231,195,195,66,0,23,245,56,12
4,254,146,210,254,254,146
22 NX:=2:YX:=3
23 ENDFROC
24 DEFPROCMAN:IF LIFE:=0 ENDFROC
25 SC(KX,YX)=" ":AB=INKEY(0)
26 #FX 15 0
27 IF AB="N" DX=-1:GHX=254
28 IF AB="M" DX=1:GHX=253
29 IF AB="A" DY=-1:GHX=252
30 IF AB="Z" DY=1:GHX=251
31 IF DX<>0 OR THEN 39
32 TX=ASC(SC(KX+DX),(YX+DY))
33 IF TX=246 THEN 45
34 IF TX=245 THEN PROCEND:ENDPROC
35 IF TX=247 SOUND 1,1,ASC(AB),1
36 IF TX=ASC("X") MANX=1:SOUND3,2,10,100:VDU 19,4,7,0,0,0:FOR DELX=1 TO 100:
NEXT:VDU 19,4,COLORX,0,0,0:IF SHEETX DIV 5=SHEETX%5:FLAX=1:TIME=0:VDU 19,4,4,0,0
,0
37 IF FLAX=1 AND TIME>500:FLAX=0:VDU 19,4,0,0,0,0
38 IF TX=247 OR TX=ASC("X") NDXX=NDX-1:IF NDXX=0 THEN PROCSSHEET:ENDPROC
39 IF DXX<>0 OR DYX<>0 PRINTTAB(KX,YX):" "
40 KX=KX+DXX:YX=YX+DYX
41 IF YX<29 YX=29
42 IF YX>29 YX=29
43 IF KX<0 KX=19
44 IF KX>19 KX=19
45 DX=0:DY=0:COLOUR3
46 PRINTTAB(KX,YX):CHR(0):CHR(0):"X"
47 ENDFROC
48 DEFPROCHOST
49 IF LIFE:=0 ENDFROC
50 SOUND2,1,10,1
51 IF YX>NGX ENDFROC
52 SC(KX,YX)=0:FOR VX=0 TO 19:SC(KX,VX)=CHR(0):NEXT:PROCMOVE:COLOUR1:IF DGX(VX)=246 COLOUR4
53 PRINTTAB(DGX(VX),GYX(VX)):CHR(0):CHR(0):"X"
54 IF DGX(VX)=246 AND GYX(VX)=DGX(VX)+DGX(VX)
55 THEN PROCEND:ENDPROC
54 GX(KX,YX)=GX(KX,YX)+DGX(KX,YX):GY(KX,YX)=GY(KX,YX)+DGX(KX,YX):SC(KX,YX)=ASC(SC(KX,YX),GY(KX,YX))
55 COLOUR VX+9:PRINTTAB(DGX(VX),GYX(VX)):CHR(0):CHR(0):"X"
56 ENDFROC
57 DEFPROCHAZE
58 VDU 23,247,0,0,0,24,24,0,0,0,23,246,255,255,255,255,255,255,255,255,255:COLOUR
2:FOR YX=0 TO 29:FOR KX=0 TO 19:SC(KX,YX)=CHR(247):NEXT:PRINTTAB(KX,YX):"X"
59 VDU19,4,COLORX,0,0,0:COLOUR4:RESTORE:FOR Z=1 TO 66:READ KX,YX:GOSUB 63:YX=
31-YX:GOSUB 63:NEXT
60 DATA 0,2,1,2,2,2,3,2,4,2,5,2,6,2,7,2,8,2,0,3,0,4,0,5,0,6,0,7,0,8,0,9,0,10,
0,11,0,12,0,13,0,14,2,4,3,4,4,4,6,4,0,3,0,4,0,5,2,5,3,5,4,5,6,5,1,14,2,14,2,7,3,
7,4,7,2,9,2,10,2,11,2,12,6,7,7,7,7,9,7,9,9,9,4,0,4,9,4,10,4,12,4,13,4,14
59 DATA 4,15,0,10,7,9,15,9,13,7,11,6,12,0,14,0,15,9,14,9,15,7,14,0,11
60 NDY:=291:COLOUR14:FOR AB=1 TO 4:READ X,Y:PRINTTAB(X,Y):"X",TAB(0,0):SC(KX,Y
)="X":NEXT
61 DATA 1,4,10,4,1,27,10,27
62 ENDFROC
63 GOSUB 64:KX=19-KX
64 PRINTTAB(KX,YX):CHR(246):"X":SC(KX,YX)=CHR(246):RETURN
65 DEFPROCK:DGX(KX)=KX:GX(KX)=KX:GX(KX)=KX:ENDPROC
66 DEFPROCDY:GY(KX)=KX:GX(KX)=KX:GX(KX)=KX:ENDPROC
67 DEFPROCDX:TX=ASC(SC(KX,GYX(KX)))>GX(KX):GX(KX)=TX:GX(KX)=TX:ENDPROC
68 DEFPROCDY:DYX(KX)=KX:GY(KX)=KX:GY(KX)=KX:ENDPROC
69 PROC:PROCLOOK:IF TX=245 DGX(KX)=0
70 PROCY:PROCLOOK:IF TX=245 DGX(KX)=0
71 ENDFROC
72 DEFPROCCOL:COL=0,-15,4,10:PROCCOL:CLS:IF SC1X>10000 LIFE=LIFE+1:SC1X=SC
1X-10000
73 VX=0:LIFE=LIFE-1:IF LIFE:=0 ENDFROC
74 SHEETX=SHEETX+1:PROCHAZE:PROCBETUP:PROCMAN:PROCMAN:COLOUR2:PRINTTAB(0,0):"
SHEET ";SHEETX:COLOUR7:PRINTTAB(0,1):"SCORE ";SCX:ENDPROC
75 DEFPROCSHEET:IF SC1X>10000 LIFE=LIFE+1:SC1X=SC1X-10000
76 VX=0:CLS:SHEETX=SHEETX+1:SCX=SCX+1:SCX(SHEETX-1):SC1X=SC1X+1:SCX(SHEETX-1):COLOURS:PRINTTAB(7,10):"BONUS":COLOUR15:PRINTTAB(6,12):SHEETX-1:" X 10000" IF
OR AB=1 TO 10000:NEXT
77 PROCMAZE:PROCBETUP:PROCMAN:PROCMAN:COLOUR2:PRINTTAB(0,0):"SHEET ";SHEETX:
COLOUR7:PRINTTAB(0,1):"SCORE ";SCX:ENDPROC
78 DEFPROCMEN:IF LIFE:=1 ENDFROC
79 COLOURS:FOR RMX=2 TO LIFE:PRINTTAB(RX,30):CHR(254):TAB(0,0):NEXT:ENDPROC
80 DEFPROCCOL:FOR C=0 TO 15:FOR N=1 TO 100:NEXT:VDU 19,4,C,0,0,0,0:NEXT:VDU 19,
4,4,0,0,0:ENDPROC
81 DEFPROCSPURT:FOR SPX=1 TO (RNX-100)>20:PROCMAN:NEXT:MANX=0:ENDPROC
82 MODE7
83 #FX 15 0
84 #FX 11 0
85 PRINTTAB(10,5):CHR(141):CHR(136):"YOU'RE DEAD!!":TAB(10,6):CHR(141):CHR
(136):"YOU'RE DEAD!!"
86 PRINTTAB(0,10):"YOU SCORED ";SCX
87 IF SCX>HSCX PRINTTAB(0,12):"THE HIGHEST SO FAR" ELSE PRINTTAB(4,12):"THE B
EST WAS ";HSCX:" BY ";HB
88 IF SCX>HSCX INPUT"" YOUR NAME "H":HSCX=SCX
89 PRINTTAB(5,22):CHR(129):CHR(141):"ANOTHER GAME ?":TAB(5,23):CHR(129):CH
R(141):"ANOTHER GAME ?"
90 REPEAT AB=GET$ UNTIL AB="Y" OR AB="N"
91 IF AB="Y" THEN 10
92 END
93 DEFPROCINSTR
94 PRINTTAB(9,0):"MAZE MUNCHER"
95 PRINT"" In this game you must move a man around a maze, eating dots
as you go so.He is being pursued by two evil ghosts who can only be stopped by
eating a flashing thunderbuster, and even then for a short while only!"
96 PRINT"" After clearing the maze of dots and thunderbusters, a new one is
given. Every fifth maze is invisible, just to add extra interest and diffi
culty.
97 PRINT"" An extra man is given every 10,000 points."
he following keys :"" "A" To go UP"" "Z" To go DOWN""
"N" To go LEFT"" "M" To go RIGHT""
98 PRINT"" PRESS ANY KEY TO PLAY... "
99 ENDFROC
```



Για να ξεφύγετε λίγο από τους κομπιούτερ, αποφασίζετε να κουρέψετε το γρασίδι στον κήπο σας, με την χλοοκοπτική μηχανή που έχετε. Ο κήπος όμως δεν είναι τόσο ήσυχος όσο φαίνεται... Η μηχανή δεν πρέπει να πέσει πάνω στις πέτρες που βρίσκονται εδώ κι εκεί ή πάνω στα μπαλάκια του κρίκετ που έχουν αφήσει τα παιδιά. Τέτοια αντικείμενα καταστρέφουν την μηχανή και το γραφίδι θα μείνει ακούρευτο αν χαλάσετε και τις τρεις μηχανές που έχετε στη διάθεσή σας. Ακόμα κι αν αποφεύγετε τα εμπόδια, υπάρχει ο κίνδυνος να τελειώσει το λάδι της χλοοκοπτικής και να πάθει πάλι ανεπανόρθωτη ζημιά. Πρέπει συχνά λοιπόν να βάζετε νέο λάδι, περνώντας απλώς πάνω από τα δοχεία λαδιού που είναι διασκορπισμένα στον κήπο.

Την πρώτη φορά που "τρέχετε" το παιχνίδι θα δείτε τους διαφόρους χαρακτήρες (πέτρες, μπαλάκια, λάδια κλπ) και τα πλήκτρα κίνησης της μηχανής. Επίσης το πρόγραμμα ζητάει την ταχύτητα της μηχανής, το επίπεδο δυσκολίας (αναλογία εμποδίων / δοχείων λαδιού) και την ένταση των θορύβων και της μουσικής.

Μετά από μια σύντομη μουσική εισαγωγή, το παιχνίδι αρχίζει και η μηχανή σας εμφα-

## ΤΟ ΚΟΨΙΜΟ ΤΗΣ ΧΛΟΗΣ

```

10 ON ERROR GOTO 30
20 hi%=0:screen%=1
30 MODE 1:PROCinit:PROCinst
40 live=2
50 score=0:screen%=1
60 VDU 23;10,32;0;0;17,130,12
70 PROCwall:PROCobst:PROCCans:PROCintro
80 PROCinit
90 REPEAT
100 PROCprint_mower
110 PROCwait(wait%)
120 PROCdisplay
130 PROCmusic
140 IF INKEY(-98)=-1 THEN A$="LEFT"
150 IF INKEY(-67)=-1 THEN A$="RIGHT"
160 IF INKEY(-73)=-1 THEN A$="UP"
170 IF INKEY(-105)=-1 THEN A$="DOWN"
180 VDU 17,3,31,X%,Y%,233
190 IF A$="LEFT" THEN X%=X%-1
200 IF A$="RIGHT" THEN X%=X%+1
210 IF A$="UP" THEN Y%=Y%-1
220 IF A$="DOWN" THEN Y%=Y%+1
230 VDU 26:col%=POINT(X%*32+16,(31-Y%)*32+16):IF col%<2 THEN PROCcheck
240 score=score+0.1:oil=oil-0.1:IF score>50*screen% THEN PROCupdate
250 IF oil<=3 THEN SOUND 3,2,1,1 ELSE SOUND &13,-0,0,0
260 IF oil<=0 THEN PROCend
270 UNTIL FALSE
280 *****
290 DEFPROCinit
300 RESTORE 1150
310 A$="RIGHT":X%=15:Y%=15:oil=40
320 ENVELOPE 1,1,6,6,6,255,255,255,126,-1,0,-10,126,80
330 ENVELOPE 2,4,6,-3,-3,4,2,2,0,0,-1,0,63,50
340 VDU 23,224,255,255,255,255,255,255,255,255,255
350 VDU 23,225,126,129,255,129,126,24,24,60
360 VDU 23,226,60,24,24,126,129,255,129,126
370 VDU 23,227,112,168,169,175,175,169,168,112
380 VDU 23,228,14,21,149,245,245,149,21,14
390 VDU 23,229,60,126,255,255,255,126,60
400 VDU 23,230,0,0,28,62,124,254,255,127
410 VDU 23,231,24,24,24,255,126,126,126,126
420 VDU 23,232,73,42,8,99,8,42,73,0
430 VDU 23,233,238,219,179,153,245,158,109,170
440 VDU 19,2,2;0;19,3,3;0;
450 ENDPROC
460 DEFPROCintro:RESTORE 500
470 FOR music=1 TO 18:READ pitch,duration:SOUND 1,-(loudness%),pitch,duration
NEXT music
480 PRINT TAB(16,15);" " " :PROCwait(3000)
490 ENDPROC
500 DATA 101,5,121,3,137,5,121,2,129,2,121,2,129,2,121,4,117,3
510 DATA 101,5,121,3,137,5,121,2,129,2,121,2,129,2,121,4,117,3
520 DEFPROCwait(T%)
530 FOR delay%=1 TO T%:NEXT delay%
540 ENDPROC
550 DEFPROCwall
560 VDU 17,1,31,2,30:PRINT"OIL ";TAB(2,31);"SCORE ";

```

νίζεται στο κέντρο του κήπου. Εσείς τώρα πρέπει να την κατευθύνετε πάνω στο άκοπο γρασίδι αποφεύγοντας τα εμπόδια και ανανεώνοντας τα λάδια όταν τελειώσουν. Όσο η μηχανή κινείται, τα λάδια εξαντλούνται και είναι σκόπιμο να μην πηγαίνετε πάνω σε κομμένο γρασίδι γιατί εκεί το σκορ δεν αυξάνει. Αν είστε αρκετά ικανός, ο κήπος θα αλλάξει μετά από κάποιο σκορ και θα εμφανιστεί νέος, πιο δύσκολος.

Το σκορ ανανεώνεται συνέχεια κατά το παιχνίδι, ενώ ακούγεται μια μουσική υπόκρουση, στην ένταση που διαλέξατε στην αρχή (Loudness). Για να αλλάξετε επίπεδο πατήστε ESCAPE, οπότε το πρόγραμμα ξαναρχίζει χωρίς να χάνεται το "High score".

Στο listing οι γραμμές 10-270 είναι το κυρίως πρόγραμμα, οι γραμμές 290-450 είναι οι αρχικές ρυθμίσεις, οι 460-540 παίξουν την μουσική στην αρχή και καθυστερούν λίγο το ξεκίνημα της μηχανής, με τις 550-650 σχεδιάζεται ο κήπος, τα εμπόδια δημιουργούνται στις 660-730, τα δοχεία λαδιού στις 740-790, στις 800-850 τυπώνεται η χλοοκοπτική, στις 860-900 γίνεται ο έλεγχος του χρώματος μπροστά από τη μηχανή, το τύπωμα των ενδείξεων γίνεται με τις 910-940, στις 950-1010 είναι η procedure του τέλους, στις 1020-1180 γίνεται η αλλαγή για καινούριο κήπο και παίζεται η μουσική, στις δε 1190-1390 βρίσκονται οι οδηγίες. Πριν βεβαιωθείτε ότι το πρόγραμμα "τρέχει" σωστά μην γράψετε την γραμμή 10 ώστε να αναφέρονται τα πιθανά σφάλματα. Προσθέστε την γραμμή αφού όλα είναι εντάξει, για να πηγαίνει το πρόγραμμα στις οδηγίες όταν πατάτε ESCAPE.

```

570 PRINT TAB(15,30);"MOWERS LEFT ";live;TAB(30,31);"SCREEN ";screen%;
580 PRINT TAB(15,31);"HI-SCORE ";TAB(15,0);"<<MOWER>>";
590 FOR wall=1 TO 38
600 VDU 31,wall,1,224,31,wall,28,224
610 NEXT wall
620 FOR wall=1 TO 28
630 VDU 31,1,wall,224,31,38,wall,224
640 NEXT wall
650 ENDPROC
660 DEFPROCobst:COLOUR1
670 FOR obst=1 TO screen%*4+6
680 VDU 31,RND(34)+1,RND(26)+1,229
690 VDU 31,RND(34)+1,RND(26)+1,230
700 IF loudness%>0 THEN SOUND 0,-15,6,1
710 PROCwait(100)
720 NEXT obst
730 ENDPROC
740 DEFPROCcans:COLOUR0
750 FOR cans=1 TO level%:VDU 31,RND(34)+1,RND(26)+1,231
760 IF loudness%>0 SOUND 2,1,1,1
770 PROCwait(700):NEXT cans
780 VDU 17,0,31,15,15,228,17,3,31,16,15:PRINT" READY!":PROCwait(600)
790 ENDPROC
800 DEFPROCprint_mower
810 IF A#="RIGHT"THEN VDU 17,0,31,X%,Y%,228
820 IF A#="LEFT"THEN VDU 17,0,31,X%,Y%,227
830 IF A#="UP"THEN VDU 17,0,31,X%,Y%,225
840 IF A#="DOWN"THEN VDU 17,0,31,X%,Y%,226
850 ENDPROC
860 DEFPROCcheck
870 IF col%=0 oil=oil+2:IF loudness%>0 SOUND 2,1,1,1
880 IF col%=3 THEN score=score-.1
890 IF col%=1 THEN PROCend
900 ENDPROC
910 DEFPROCdisplay:COLOUR1
920 IF score>hi% THEN hi%=score
930 PRINT TAB(8,30);INT(oil);" ";TAB(8,31);INT(score);TAB(24,31);hi%;
940 ENDPROC
950 DEFPROCend:SOUND 0,1,1,1
960 SOUND &11,-0,0,0:SOUND &13,-0,0,0:VDU 17,0,31,X%,Y%,232:PROCwait(3000)
970 live=live-1:IF live=0 THEN 60:ENDPROC
980 score=0:FX 15,0
990 VDU 17,0,30:PRINT"DEAD!":PRINT"press SPACE BAR to play again!"
1000 REPEAT UNTIL GET=32:GOTO 40
1010 ENDPROC
1020 DEFPROCupdate
1030 SOUND &11,-0,0,0
1040 PROCprint_mower:VDU 17,0,31,11,14
1050 PRINT"SCREEN ";screen%;" COMPLETED!":PROCnewscreen
1060 PROCwait(5000):screen%=screen%+1:GOTO 60:ENDPROC
1070 DEFPROCnewscreen:RESTORE 1100
1080 FOR music=1 TO 6:READ pitch,duration:SOUND 1,-(loudness%),pitch,duration
1090 NEXT music:ENDPROC
1100 DATA 81,3,101,3,117,3,129,5,117,3,129,5
1110 DEFPROCmusic:IF ADVAL(-6)<1 THEN ENDPROC
1120 READ pitch%,duration%:IF pitch%=0 THEN volume=0 ELSE volume=-loudness%
1130 IF duration%=999 THEN RESTORE 1150:ENDPROC
1140 SOUND 1,volume,pitch%,duration%:ENDPROC
1150 DATA 101,5,121,5,0,0,121,5,117,3,109,2,117,4,121,5,101,4,0,1,101,4,0,1
1160 DATA 101,4,121,5,0,0,121,5,117,3,109,2,117,4,121,5,101,4,0,1,101,4,0,1
1170 DATA 101,4,89,5,0,0,89,5,81,5,0,0,81,5,73,6,0,7,101,5,89,5,0,0,89,5
1180 DATA 81,5,0,0,81,5,73,6,0,10,999,999
1190 DEFPROCinst
1260 VDU 17,130,12,17,1,31,12,0:PRINT"MOWER CHARACTERS"
1270 VDU 17,1,31,5,3,224:PRINT" -----wall...."
1280 VDU 17,0,31,5,5,228:PRINT" -----you...."
1290 VDU 17,1,31,5,7,229:PRINT" -Explosive cricket ball...."
1300 VDU 31,5,9,230:PRINT" -----Rock...."
1310 VDU 17,0,31,5,11,231:PRINT" -----oil can...."
1320 COLOUR1:PRINT"" CONTROLS:-"
1330 PRINT" 2 -----LEFT..":PRINT" X -----RIGHT.."
1340 PRINT" : -----UP..":PRINT" / -----DOWN.."
1350 COLOUR 0:PRINT" input speed 1(FAST) to 9(SLOW) ";:REPEAT I%=INSTR("123456789",GET#):UNTIL I%:PRINT;I%:wait%:I%*10
1360 PRINT" input level 1(HARD) to 5(EASY) ";:REPEAT I%=INSTR("12345",GET#):UN
TIL I%:PRINT;I%:level%=I%*4+3
1370 PRINT" input loudness 0(SILENT) to 9(LOUD) ";:REPEAT I%=INSTR("123456789",GET#):UNTIL I%:PRINT;I%:loudness%=(I%-1)
1380 PRINT"" press SPACE BAR to play game!";:REPEAT UNTIL GET=32
1390 ENDPROC

```

# ΑΠΟΣΤΟΛΗ ΣΤΟΝ ΠΛΑΝΗΤΗ ΔΕΛΤΑ

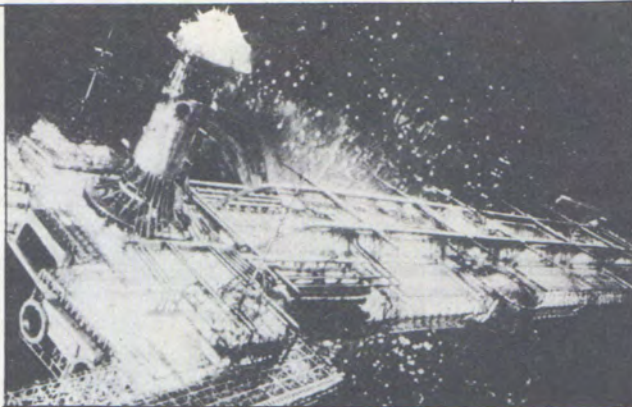
Το έτος είναι 2090. Όλα τα αποθέματα ενέργειας έχουν εξαντληθεί, κι' έτσι οι κάτοικοι τη Γης έχουν καταφύγει σε μια νέα πηγή ενέργειας, το Ζυθόνιο. Αυτό είναι ένα εξαιρετικά αξιοποιησιμο στοιχείο και βρίσκεται μόνον στα βάθη του πλανήτη Δέλτα. Κάθ ένας γενναίος πιλότος στέλνεται στον Δέλτα για να μαζέψει ένα φορτίο κρυστάλλων Ζυθονίου, πετώντας μέσα στις δαιδαλώδεις στοές του υπέδαφους του πλανήτη. Αυτό το χρόνο είσατε εσείς ο εκλεκτός στη Γη.

Ξαφνικά τα πάντα στο διαστημόπλοιο τρελαίνονται και ο αυτόματος πιλότος μαζί με τ' άλλα συστήματα αχρηστεύονται! Το σκάφος σας τρέχει με ιλιγγιώση ταχύτητα μέσα στις στοές κι' εσείς προσπαθείτε απεγνωσμένα να μην συγκρουσθεί-

τε στα τοιχώματα! Το σκάφος σας έχει ενέργεια για 10 μόνο δευτερόλεπτα πτήσης και ο μόνος τρόπος ανεφοδιασμού είναι να μαζεύετε τους κρυστάλλους που αιωρούνται στις στοές. Πόσο όμως μπορείτε ν' αντέξετε;

Το σκάφος ανεβαίνει πατώντας το "A" και κατεβαίνει πατώντας το "Z". Το πρόσδευτερόλεπτα πτήσης έχετε, καθώς και η απόσταση που έχετε διανύσει τυπώνονται συνέχεια στο κάτω μέρος της οθόνης. Τα καύσιμα αρχίζουν να εξαντλούνται μόνον αφού το σκορ περάσει τα 40. Για πιο εύκολους διαδρόμους πρέπει να αλλάξετε κατάλληλα τις δυνατές τιμές των μεταβλητών UP και DOWN

Μετά το τέλος κάθε παιχνιδιού, πατήστε ένα πλήκτρο για να αρχίσει καινούρια αποστολή.



```

10 MODE7
20 F%#0
30 ENVELOPE 1,1,-26,-26,255,255,255,50,0,0,-50,49,0
40 ENVELOPE 2,1,1,1,1,10,10,100,100,-1,0,0,100,0
50 PROCINIT
60 MODE7:Y%#1:VDU 23;0202;0;0;0;
70 SOUND 1,1,255,255
80 PRINTTBK(0,21)"CAVE FLIGHT"
90 UP#2:DOWN#15
100 X#10:FOR I%#0 TO 20:PRINTTBK(0,I%);CHR$(69):NEXT
110 TIME#0
120 REPEAT
130 FOR I%#0 TO 20:PRINTTBK(39,I%) " ":NEXT
140 IF F%#40 THEN TIME#0
150FOR I%#0 TO UP:PRINTTBK(39,I%);CHR$(255):NEXT
160FOR I%#DOWN TO 20:PRINTTBK(39,I%);CHR$(255):NEXT
170PRINTTBK(1,X)" "
180A#INKEY$(0):IF A#="A" AND X>2 THEN X#X-1
190IF A#="Z" AND X<20 THEN X#X+1
200J%#HMEM#3+400X:J%#7J%
210IF J%#255 OR TIME>1000 THEN 910
220IF J%#R$(X) THEN TIME#0:SOUND 2,2,100,20
230 IF RND(10)>1 THEN PRINTTBK(37,RND(DOWN-UP)+UP);CHR$(61+RND(6));"X";CHR$(69)
1)
240 Y%#Y%+1:IF Y%#30 THEN PRINTTBK(37,RND(DOWN-UP)+UP);CHR$(61);"X";CHR$(69):
Y%#1
250 CALL START
260 PRINTTBK(1,X);CHR$(244);CHR$(240)
270 PRINTTBK(0,22)"TIME LEFT ":10-(TIME DIV 100);" "
280 PRINTTBK(0,23)"DIST. TRAVELLED:";F%#F%+1
290 UP#UP+RND(3)-2:DOWN#DOWN+RND(3)-2:IF DOWN<5 THEN DOWN#DOWN+RND(5)
300 IF UP>15 THEN UP#UP-RND(5)
310 IF UP<1 THEN UP#1
320 IF DOWN>19 THEN DOWN#19
330 IF DOWN<UP+2 THEN DOWN#UP+4
340 UNTIL FALSE
350DEFPROCINIT
360DIM X 140
370FOR PASS#0 TO 2 STEP 2
380P%#X
390 IOPT PASS
400.START LDX #2
410 .PP LDR HMEM,X
420STA HMEM-1,X
430LDR HMEM+40,X
440STA HMEM+39,X
450LDR HMEM+80,X
460STA HMEM+79,X
470LDR HMEM+120,X
480STA HMEM+119,X
490LDR HMEM+160,X
500STA HMEM+159,X
510LDR HMEM+200,X
520STA HMEM+199,X
530LDR HMEM+240,X
540STA HMEM+239,X
550LDR HMEM+280,X
560STA HMEM+279,X
570LDR HMEM+320,X
580STA HMEM+319,X
590LDR HMEM+360,X
600STA HMEM+359,X
610LDR HMEM+400,X
620STA HMEM+399,X
630LDR HMEM+440,X
640STA HMEM+439,X
650LDR HMEM+480,X
660STA HMEM+479,X
670LDR HMEM+520,X
680STA HMEM+519,X
690LDR HMEM+560,X
700STA HMEM+559,X
710LDR HMEM+600,X
720STA HMEM+599,X
730LDR HMEM+640,X
740STA HMEM+639,X
750LDR HMEM+680,X
760STA HMEM+679,X
770LDR HMEM+720,X
780STA HMEM+719,X
790LDR HMEM+760,X
800STA HMEM+759,X
810LDR HMEM+800,X
820 STA HMEM+799,X
830 INK
840 CPX #40
850 SEQ P
860 JMP PP
870 .P RTS
880 J
890 NEXT PASS
900 ENDPROC
910 PRINT TBK(0,11);STRING$(40," ");TBK(10,11)"GAME OVER"
920 PRINTTBK(0,12);STRING$(40," ");TBK(9,12);CHR$(67);"YOUR SCORE ";F%
930 PRINTTBK(0,13);STRING$(40," ");TBK(9,13);CHR$(67);"HI SCORE ";H%
940 IF F%>H% THEN PRINTTBK(0,14);STRING$(40," ");TBK(8,14);CHR$(67);CHR$(68);"
YOU GOT THE HIGH SCORE!!":H%#F%
950 #FX15,0
960 FOR I=1 TO 10000:NEXT:#FX15,0
970 GG=GET:RUN

```





## ΕΠΙΚΙΝΔΥΝΟΙ ΜΕΤΕΩΡΙΤΕΣ

Για μια ακόμα φορά βρίσκεστε αντιμέτωποι με τις δυνάμεις της φύσης, καθώς αγωνίζεστε να σώσετε το σκάφος σας από σύγκρουση με μετεωρίτη. Το ευχάριστο είναι ότι μόνον ένας μετεωρίτης σας απειλεί κάθε φορά και μπορείτε να τον καταστρέψετε με το Laser του σκάφους πριν σας χτυπήσει. Δεν είναι όμως τόσο εύκολο να τον πετύχετε έγκαιρα γιατί η ταχύτητά του είναι μεγάλη. Πάλι όμως μπορείτε να τον ξεφύγετε προχωρώντας το σκάφος σας ή, σε ύστατη ανάγκη, να περάσετε στο Υπερδιάστημα, μη γνωρίζοντας όμως που ακριβώς θα επιστρέψετε.

Το πρόγραμμα θα δοκιμάσει σκληρά τις δυνατότητές σας στην πλοήγηση, ενώ απόκομοι ήχοι θα πολιορκούν τα αυτιά σας! Ευτυχώς μπορείτε να σταματήσετε τους θορύ-

βους, στην αρχή του παιχνιδιού, για πιο ήσυχο παιχνίδι... Για πιο μεγάλη ταχύτητα στο "τρέξιμο" χρησιμοποιούνται μεγάλες σειρές πολλαπλών εντολών, κάτι που προσθέτει λίγη δυσκολία κατά την πληκτρολόγηση. Μπορείτε, στην αρχή, να παραλείψετε την γραμμή 17 ώστε να αναφέρονται τυχόν λάθη και να μην "τρέχει" αυτόματα το πρόγραμμα.

Για να αποφύγετε την καταστροφική σύγκρουση με μετεωρίτη χρησιμοποιήστε τα παρακάτω πλήκτρα:

f0: αριστερόστροφη περιστροφή

f1: δεξιόστροφη περιστροφή

f2: άμεση στροφή 180 μοιρών

f3, f4, f5: υπερδιάστημα

f6: προώθηση

f7: Laser

```

> L
1 REM *****
2 REM * ASTEROIDS *
3 REM *****
10 ON ERROR RUN
20 MODE7:PROCOFF:FX15,0
30 DIM S(90),CO(90):FOR A%=0 TO 360 STEP 4:PRINTAB(12,12):INT((360-A%)/36):
  " :S(1/4)=SIN RAD(A%):CO(1/4)=COS RAD(A%):NEXT A%
40 KKEY10:OLDIM RUNIM
50 PROCOFF:VDU23,224,24,24,60,60,126,126,231,231,23,225,60,69,153,165,165,153
69,60:NUM="CHR$(224):FX15,0
60 PRINT CHR$(12):TAB(0,0):STRING(40,"*"):TAB(0,1)* P S Y C H I C A S T
E R O I D S *:TAB(0,2):STRING(40,"*"):FX15,0
70 PRINT""Do you want background noises (Y/N) :":GG=GET:FX9,5
80 ON ERROR GOTO 160
90 ENVELOPE1,-10,10,100,200,10,200,120,-100,-4,-1,120,127:FX10,5
100 FOR X=100 TO 120: SOUND1,1,X,1: SOUND2,1,(225-X),1:NEXT X:X=INKEY(20)
110 HIGHSCORE=550:HIGHSCORER="ACORN":R=" ":HS=550:MF=1,5
120 ENVELOPE1,0,-0,-3,122,227,245,194,123,-76,-1,-6,126,126:ENVELOPE2,129,2,4,
6,20,14,7,0,0,0,-00,00,00
130 ENVELOPE4,1,1,2,0,0,123,1,-110,0,-1,-20,125,70:ENVELOPE3,1,1,2,1,3,1,2,12,-
1,1,-1,-1,126,126
140 XX=NRND(90):STX=640:RTX=512:R="M":YX=RTX:XX=STX
150 SCORE=0:NLEFT=0:JX=20:PROCCastinit:STV255,2
160 MODE4:VDU19,0,1,0,0,0:PROCCScore:PROCCastars(400):PROCCdraw
170 VDU24,000,000,000,000,0FF,004,00A,003
180 REPEAT PROCCcontrol:PROCCasteroid:UNTIL NLEFT<=0:PROCCend:GOTO 140
190 DEFPROCCcontrol
200 A%=7236:B%=7237:IF A%=160 OR B%=160:XX=XX+8:NRND(2):PROCCdraw:PROCCtest(A%,
B%,C):IF NRND(3)=1 GOTO 200
210 A%=7236:IF A%=241 OR B%=241:XX=XX+8:NRND(2):PROCCdraw:PROCCtest(A%,B%,C):IF
NRND(3)=1 GOTO 210
220 IF A%=242 OR B%=242:XX=XX+100:PROCCdraw:PROCCtest(A%,B%,C):PROCCasteroid
230 IF A%=245 OR B%=245:PROCCdistance:IF JX<25%:PROCCthrust(70):PROCCasteroid
240 IF A%=140 OR A%=243 OR A%=244:R="HYPERSPACE":PROCCdraw:PROCCtest(A%,B%,C)
250 IF A%=140 OR A%=243 OR A%=244:IF NRND(3)=1:NLEFT=NLEFT-1:VDU7:PROCCScore
260 IF A%=150 OR B%=150 AND NRND(1)>.35:PROCCdistance:IF JX<40:PROCCfire:PROCCast
eroid
270 ENDPROC
280 DEFPROCCcalc:XX=(360+(XX<0)*ABS(XX)-(XX>0)*ABS(XX)) MOD 360:XI=(XX+18
0) MOD 360:X2=(XX+215) MOD 360:X3=(XX+145) MOD 360
290 IF RTX<945:RTX=5:STX=(1200-STX) ELSE IF RTX<0:RTX=940:STX=(1200-STX):GOTO
310
300 IF STX<1200:STX=5:RTX=(1024-RTX) ELSE IF STX<0:STX=1275:RTX=(1024-RTX)
310 YX=S(1/4)*38+RTX:YX=S(1/4)*4*42+RTX:YX=S(1/4)*4*42+RTX
320 X2=CO(1/4)*38+STX:X2=CO(1/4)*4*42+STX:X2=CO(1/4)*4*42+STX
330 CX=CO(1/4)*4*42+STX:CY=S(1/4)*38+RTX:CY=S(1/4)*4*42+STX:ENDPROC
340 DEFPROCCdraw:IF NRND(3)=1 AND R="HYPERSPACE":PROCCasteroid
350 MOVECX,YX:PL0T7,X1,X2:YX:PL0T7,CX,CY:PL0T7,XX,YY:PL0T7,X2,YX:PL0T7,X2,YX
360 IF R="HYPERSPACE":RTX=NRND(90):STX=NRND(1200):XX=NRND(360)
370 R="M":PROCCcalc:MOVE CX,YX:DRAW X1,X2:YX:DRAW CX,CY:DRAW XX,YY:DRAW X2
,YX:ENDPROC
380 DEFPROCCthrust(B%):SOUND0,-15,100,0:RTX=RTX+SI(XX/4)*30:STX=STX+CO(X/4)
4)30:PROCCdraw:ENDPROC
390 DEFPROCCfire: SOUND1,1,1,18:VDU5
400 FX=YX+S(1/4)*XJ:FY=CO(X/4)*XJ+XK
410 FOR NN=0 TO 1:GC0L0,1:MOVE X0,Y0:DRAW FX,FY:PL0T 0,0,16:PRINT CHR$(225
420 GC0L 0,0:MOVE X0,Y0:DRAW FX,FY:PL0T 0,0,16:PRINT CHR$(225
430 NEXT:VDU4:PROCOFF:GC0L0,1:IF SQK(FX-A%)*2+(FY-B%)*2<25%:PROCCast
440 ENDPROC
450 DEFPROCCasteroid:IF NRND(3)=1:PROCCdiff
460 IF G=89 AND NRND(4)=1: SOUND NRND(3),3, NRND(255),2
470 PROCCast2:AX=XK+KTX:BY=BY+JTY:CM=CM+NF:MF=MF+NF:MOVE (A%-10%),(B%-10%)
480 DRAW (A%+20%),(B%-5%):DRAW (A%+25%),(B%+25%):DRAW (A%+5%),(B%+30%)
490 DRAW (A%-10%),(B%-10%):PROCCtest(A%,B%,C)
500 IF A%<1200 OR B%<1024 OR A%<0 OR B%<0:PROCCast2:PROCCastinit
510 ENDPROC
520 DEFPROCCstars(N)
530 FOR X=0 TO N:JX=NRND(1200):KX=NRND(1024):MOVE JX,KX:PL0T 69,JX,KX:NEXT:ENDP
ROC
540 DEFPROCCScore:PROCOFF:IF SCORE>HS:HS=SCORE
550 PRINT CHR$(30):SCORE="":SCORE:HIGH SCORE="":HS:TAB(25,1):STRING(40:NLEFT
,NUM):" "
570 ENDPROC
580 DEFPROCCdestru:VDU19,0,0,0,0,0:NLEFT=NLEFT-1:PROCCScore:FX15,0
590 SOUND0,-15,5,50:FORX=1 TO 5:FORX=1 TO 20 STEP 2:D=INKEY(2):VDU23,13,XP
KX,0,0:NEXT:FORX=1 TO 20 STEP 2:D=INKEY(2):VDU23,13,XP,KX,0,0:NEXT:R="M
HYPERSPACE":PROCCdraw:SOUND0,-15,6,12:R="M":VDU19,0,1,0,0,0:ENDPROC
600 DEFPROCCend:VDU2,0,0,0,0
610 VDU5:MOVE352,525:PRINT"GAME OVER":VDU4:FX15,0
620 G=INKEY(100):FX15,0
630 VDU2,7:PRINT""Your score is ";SCORE;" points""
640 PRINT"High score is ";HIGHSCORE;" points by";CHR$(129);HIGHSCORER""
650 IF SCORE>HIGHSCORE:HIGHSCORE=SCORE:PRINT"Yours is the highest score":INPU
T"Please enter initials ",HIGHSCORER""
660 HIGHSCORER=LEFT$(HIGHSCORER,4):INPUT"PRESS RETURN":R=HIGHSCORE:ENDPROC
670 DEFPROCCastinit:NRND(90):NRND(1200)
680 C=(NRND(30)+10)*3E-2
690 REPEAT:POSX=NRND(4):A%=NRND(1200)*K-1:POSX=2 OR POSX=4)-1200*(POSX=3)
700 B%=NRND(960)*K-1:POSX=1 OR POSX=3)-950*(POSX=2):UNTIL ABS(STX-A%)+3
710 PROCCdiff:ENDPROC
720 DEFPROCCdiff:DIFFX=(STX-A%):DIFFY=(RTX-B%):IF DIFFY=0 OR DIFFX=0:ENDPROC
C
730 DX=DIFFX/ABS(DIFFX):DY=DIFFY/ABS(DIFFY)
740 IF ABS(DIFFX)>20:KTX=NRND(5)+0:KDY:RTX=DIFFX/DIFFX+KTX:ENDPROC
750 IF ABS(DIFFY)>20:JTY=NRND(5)+0:KDY:KTX=DIFFY/DIFFY+JTY:ENDPROC
760 KTX=1000:JTY=1000:ENDPROC
770 DEFPROCCastdest:PROCCast2:SC=INT((SCORE-150-INT(50%)))/10*10
780 PROCCextra:SCORE=SC:PROCCScore:PROCCexplosion(A%,B%,65,2,2):PROCCastinit:PROCA
steroid:ENDPROC
790 DEFPROCCast2:MOVE (A%-10%),(B%-10%):PL0T7,(A%+20%),(B%-5%):PL0T7,(A%+2
5%),(B%-25%)
800 PL0T7,(A%+5%),(B%+30%):PL0T7,(A%-10%),(B%-10%):ENDPROC
810 DEFPROCCdistance:IF NRND(3)=1:FF=2 ELSE FF=1
820 JX=SQK(A%-X0)*2+(B%-Y0)*2:IF JX<500 THEN JX=JX/FF ELSE JX=JX/2
830 ENDPROC
840 DEFPROCCexplosion(XXX,YYY,ZZZ,WWW):IF C<.7:VDU19,1,15,0,0,0,19,0,11,0,0,0:J
KK=67 ELSE JKK=69
850 FOR D=-22 TO 0:IF D=-46:PPX=0/3 ELSE PPX=-15
860 SOUND 0,PPX,6,1:T=(150+D)*0.0001:PL0T JKK,RND(T)-T/2+XXX,RND(T)-T/2+YYY:NEXT:
VDU19,0,1,0,0,0,19,1,7,0,0,0:ENDPROC
870 DEFPROCCextra:GP=(SC DIV 5000)*5000:IF SCORE>GP:ENDPROC ELSE NLEFT=NLEFT
+1:VDU7:ENDPROC
880 DEFPROCCtest(Q,W,E)
890 PROCCdistance:IF JX<40:PROCCdestru:PROCCast2:PROCCastinit:PROCCasteroid
900 ENDPROC
910 DEFPROCOFF:VDU 23,1,0,0,0,0:ENDPROC

```

Προγράμματα για τόν:

# EPSON HX-20



ΕΠΙΜΕΛΕΙΑ: Αντώνης Αϊδίνης

# H BASIC TOY EPSON KAI H MICROSOFT BASIC

Η BASIC του EPSON είναι συμβατή με την MBASIC-80 που συναντάμε σε συστήματα με δισκέτες. Από τον πίνακα σύγκρισης που παραθέτουμε ελπίζουμε να βοηθηθεί ο αναγνώστης που θα θελήσει ίσως να μεταφέρει τα προγράμματα σε άλλο σύστημα, εγχείρημα καθόλου εύκολο αξιολόγησης της ιδιομορφίας του HX-20.

Όπως φαίνεται και από τον πίνακα σύγκρισης, η EPSON BASIC περιλαμβάνει αρκετές επιπλέον εντολές που δεν είναι δυνατό να συναντήσει κανείς σε άλλο σύστη-

μα. Υπάρχουν εντολές για να χρησιμοποιηθεί σαν αρχείο ένα τμήμα της RAM μνήμης (εντολές που χρησιμοποιούνται στο πρώτο πρόγραμμα), εντολές για να εκμεταλλευθεί κανείς το ψηφιακό μικροκασετόφωνο που μπορεί να κινείται εμπρός-πίσω με γρήγορη ταχύτητα από πρόγραμμα, εντολές για τη δημιουργία GRAPHICS (όπως χρησιμοποιούνται στο δεύτερο πρόγραμμα) και εντολές για τη χρησιμοποίηση του ίδιου του μηχανήματος, όπως η LOGIN 1,2,3,4,5 που χρησιμοποιείται για τη μετάβαση από πρόγραμμα σε πρόγραμμα.

## \*EPSON\*

## \*MBASIC-80\*

### \*COMANDS-STATEMENTS\*

1. AUTO	1. AUTO
	2. CALL
	3. CHAIN
2. CLEAR	4. CLEAR
	5. CLOAD
3. CLOSE	6. CLOSE
4. CLS	
5. COLOR	7. COMMON
	8. CONT
6. CONT	
7. COPY	9. CSAVE
	10. DATA
8. DATA	11. DEFFN
9. DEFFN	
10. DEFFIL	12. DEFINT
11. DEFINT	13. DEFSNG
12. DEFSNG	14. DEFDBL
13. DEFDBL	15. DEFSTR
14. DEFSTR	

15. DEFUSR	16. DEFUSR
16. DELETE	17. DELETE
17. DIM	18. DIM
	19. EDIT
18. END	20. END
19. ERASE	21. ERASE
20. ERROR	22. ERROR
21. EXEC	
	23. FIELD
22. FILES	
23. FOR. TO.	24. FOR. TO.
24. GCLS	
	25. GET
25. GET%	
26. GOSUB. RE	26. GOSUB. RE
27. GOTO	27. GOTO
28. IF. THEN.	28. IF. THEN.
29. IF. GOTO.	29. IF. GOTO.
30. INPUT	30. INPUT
31. INPUT#	31. INPUT#
32. KEY	
33. KEY LIST	
34. KEY LLIS	
	32. KILL
35. LET	33. LET
36. LINE	
37. LINE INP	34. LINE INP
38. LINE INP	35. LINE INP
39. LIST	36. LIST
40. LLIST	37. LLIST
41. LIST<FIL	
42. LIST COM	
43. LOAD	38. LOAD
44. LOAD COM	
45. LOADM	
46. LOAD?	
47. LOCATE	
48. LOCATES	
49. LOGIN	
50. LPRINT	39. LPRINT
51. LPRINT U	40. LPRINT U
	41. LSET/RSE
52. MEMSET	

53. MERGE	42. MERGE
54. MERGE CO	
55. MID\$	43. MID\$
56. MON	
57. MOTOR	
	44. NAME
58. NEW	45. NEW
	46. NULL
59. ON ERROR	47. ON ERROR
60. ON. GOSUB	48. ON. GOSUB
61. ON. GOTO	49. ON. GOTO
62. OPEN	50. OPEN
63. OPEN COM	
64. OPTION B	51. OPTION B
	52. OUT
65. PCOPY	
66. POKE	53. POKE
67. PRESET	
68. PRINT	54. PRINT
69. PRINT US	55. PRINT US
70. PRINT#	56. PRINT#
71. PRINT#US	57. PRINT#US
72. PSET	
	58. PUT
73. PUT%	

74. RANDOMIZ	59. RANDOMIZ
75. READ	60. READ
76. REM	61. REM
77. RENUM	62. RENUM
78. RESTORE	63. RESTORE
79. RESUME	64. RESUME
80. RUN	65. RUN
81. RUN COMO	
82. SAVE	66. SAVE
83. SAVE COM	
84. SAVEM	
85. SCREEN	
86. SCROL	
87. SOUND	
88. STAT	
89. STOP	67. STOP
90. SWAR	68. SWAR
91. TITLE	
92. TRON/TRO	69. TRON/TRO
	70. WAIT
	71. WHILE. WE
93. WIDTH	72. WIDTH
94. WIDTH<DE	
95. WIND	
	73. WRITE
	74. WRITE#

## ΥΠΕΝΘΥΜΙΣΤΗΣ

Το πρόγραμμα αυτό εκμεταλλεύεται το εσωτερικό ρολόι του EPSON που κρατάει την ώρα ακόμα κι όταν το σύστημα είναι κλειστό. Η εντολή MOTOR ON/OFF ανοίγει και κλείνει ένα εσωτερικό ρελαί το οποίο δίνει έξοδο στο βύσμα REM (χρησιμοποιείται κανονικά για έλεγχο μοτέρ κοινού κασετόφωνου) και το οποίο θα μπορούσε να χρησιμοποιηθεί για να ανάψει οποιαδήποτε εξωτερική συσκευή

(είτε άμεσα, αν πρόκειται για χαμηλής ισχύος, συνεχούς ρεύματος είτε έμμεσα μέσα από ισχυρότερα ρελαί).

Η ημερομηνία και η ώρα δίνονται μια φορά κατά την "ψυχρή εκκίνηση" του συστήματος και στη συνέχεια παρακολουθούνται αυτόματα. Η ώρα είναι διαθέσιμη στην BASIC σαν αλφαριθμητική μεταβλητή TIME και η ημερομηνία σαν αλφαριθμητική μεταβλητή DATE.

```

10 CLS:LINEINPUT"ΩΡΑ ":A$
20 LINEINPUT"ΗΜ/ΝΙΑ ":B$
30 LOCATE50,0:LOCATE0,2,0
40 PRINT"ΩΡΑ :":TIME$
50 PRINT"ΗΜ/ΝΙΑ :":DATE$
60 IF A$=TIME$ AND B$=DATE$ THEN S=1:MOTOR ON
70 IF S=1 THEN SOUND 10,10
80 QQ$=INKEY$:IF QQ$="" THEN 30
90 S=0:MOTOR OFF
100 CLS:PRINT"1. ΝΕΑ ΩΡΑ":PRINT"2. ΤΕΛΟΣ"
110 QQ$=INKEY$:IF QQ$<"1" OR QQ$>"2" THEN 110
120 ON VAL(QQ$) GOTO 10,200
200
200 END

```

## ΤΗΛΕΦΩΝΙΚΟΣ ΚΑΤΑΛΟΓΟΣ ΣΤΟΝ EPSON

Το παρακάτω πρόγραμμα χρησιμοποιεί δυο χαρακτηριστικές ιδιότητες του EPSON HX-20 για να μας επιτρέψει να έχουμε κάθε στιγμή ένα τηλεφωνικό κατάλογο 200 και πλέον ονομάτων: Η πρώτη είναι η λεγόμενη RAM FILE, μια περιοχή δηλαδή στη μνήμη την οποία η BASIC βλέπει σαν αρχείο στο οποίο μπορεί να κάνει άμεση προσπάθεια σε ελάχιστο χρόνο για να διαβάσει τα στοιχεία του ντιστού RECORD. Η δεύτερη ιδιότητα είναι η ικανότητα του μηχανήματος αυτού να διατηρεί το περιεχόμενο της μνήμης του ακόμα κι όταν το κλείσουμε. Έτσι μπορούμε να έχουμε τα τηλέφωνα που χρειαζόμαστε ανά πάσα στιγμή.

Η εντολή στη γραμμή 10, δηλαδή η CLEAR 500, 4050 ορίζει ακριβώς με το νούμερο 4050 το επιθυμητό για αυτή την εφαρμογή μέγεθος σε BYTES του RAM FILE. Το πρώτο νούμερο, το 500, ορίζει ένα WORK-SPACE όπου θα γίνεται η επεξεργασία των αλφαριθμητικών μεταβλητών του προγράμματος.

Με την εντολή DEFFIL 20,0 ορίζουμε ότι τα RECORDS μέσα στο RAM FILE έχουν το καθένα μήκος 20 BYTES (12 γράμματα για το όνομα και 8 για το τηλέφωνο) και ότι αρχίζουν από την αρχή της RAM FILE.

Με την εντολή GET%0 διαβάζουμε από το μηδενικό RECORD το πλήθος των ονομάτων που ήδη υπάρχουν στο αρχείο μας. Το μηδενικό RECORD χρησιμοποιείται δηλαδή εδώ σαν δείκτης του πόσο γεμάτο είναι το αρχείο μας ώστε ένα καινούριο όνομα που τυχόν θα προσθέσουμε να μη σβήσει κανένα προηγούμενο.

Το πρόγραμμα έχει συνολικό μήκος 756 BYTES και οι απαιτήσεις του σε μνήμη είναι 500+4050 BYTES ή 4550 BYTES. Συνολικά δηλαδή η χρήση του προγράμματος απορροφά 5306 BYTES και αφήνει ελεύθερα περίπου 8K στο βασικό HX-20 ή 24K BYTES στο HX-20 με την επέκταση μνήμης (τα ελεύθερα BYTES έχουν ειδικά στον EPSON μεγάλη σημασία γιατί μπορούν θαυμάσια να χρησιμοποιηθούν από άλλα προγράμματα). Μάλιστα, σ' αυτό το μηχάνημα υπάρχει η δυνατότητα να διατηρούμε στη μνήμη 5 (πέντε) διαφορετικά προγράμματα, αρκεί μόνο το σύνολο των BYTES που καταλαμβάνουν να μη ξεπερνά τη μνήμη RAM που έχουμε διαθέσιμη. Αυτά τα προγράμματα μπορούμε να τα έχουμε μόνιμα στο μηχάνημα γιατί δεν χάνονται όταν το κλείσουμε. Από το ένα πρόγραμμα στο άλλο περνάμε με την εντολή LOGIN 1 ή LOGIN 2 ή LOGIN 3,4,5.

```
10 CLEAR 500,4050
20 DEFFIL 20,0
30 GET%0,C#,D$:IFC#>200
  THEN C#=0
35 C%=C#
```

```
40 PRINT"1.WRITE"
50 PRINT"2.READ"
60 PRINT"3.EXAMINE"
65 PRINT"@.EXIT PRGRM";
70 QQ$=INKEY$:IF QQ$=""
  THEN 70
75 IF QQ$="@" THEN GOTO
  5000
80 IF VAL(QQ$)<1 OR VAL(
  QQ$)>3 THEN GOTO 70
90 ON VAL(QQ$) GOTO 1000
  ,2000,3000
999 REM *** WRITE ***
1000 CLS
1010 INPUT"ONOMA";N$
1020 IF N$="@" THEN C#=C
  %:PUT%0,C#,T$:GOTO 40
1030 INPUT"TELEPHONE";T#
1040 C%=C%+1
1050 PUT%C%,T#,N$
1060 GOTO 1010
1999 REM ** READ **
2000 CLS:INPUT"ONOMA";N1
  $
2005 IF N1$="@" THEN GOTO
  40
2010 PUT%201,T#,N1$:GET%
  201,T#,N1$
2020 FOR I=1 TO C%
2030 GET%I,T#,N$
2040 IF N$=N1$ THEN PRIN
  T"TELEPHONE ";T#
2050 NEXT I
2060 PRINT"NOT FOUND"
2070 QQ$=INKEY$:IFQQ$=""
  THEN 2070 ELSE 2000
2990 REM ** EXAMINE **
3000 CLS:FOR I=1 TO.C%
3010 GET%I,T#,N$
3020 PRINT"ONOMA ";N$
3030 PRINT"TELEPHONE ";T
  #
3040 QQ$=INKEY$:IFQQ$=""
  THEN 3040
3045 IF QQ$="@" THEN GOT
  0 40
3050 NEXT I
3060 GOTO 40
5000 END
```

# ΕΠΙΤΟΚΙΑ ΗΜΕΡΟΜΗΝΙΕΣ

Το πρόγραμμα αυτό συνδυάζει δυο χρήσιμες λειτουργίες: η μία είναι εκείνη του ημερολογίου-υπολογιστή και η άλλη του υπολογιστή επιτοκίων. Αναλυτικότερα, ο τρόπος χρήσης του είναι ο εξής: Επιλέγοντας από το MENU τη λειτουργία "Ι. Επιτόκια" το σύστημα μας ρωτάει διαδοχικά:

- α) Την αρχική αξία της κατάθεσής μας,
- β) την μελλοντική αξία που θα έχει η κατάθεσή μας,
- γ) την χρονική διάρκεια της κατάθεσης,
- δ) το επιτόκιο

Η ιδιομορφία του προγράμματος είναι τέτοια που μας επιτρέπει να κάνουμε υπολογισμούς θεωρώντας οποιοδήποτε από τα (α), (β), (γ), (δ) σαν άγνωστο. Απλά, εκεί που μας ρωτάει το πρόγραμμα να του δώσουμε την αξία, αν μας είναι άγνωστη δεν έχουμε παρά να δώσουμε 0 (μηδέν). Συμπληρώνοντας τις υπόλοιπες γνωστές τιμές το πρόγραμμα θα μας δώσει την άγνωστη. Μ'άλλα λόγια, έχουμε την ευχέρεια να υπολογίζουμε τον αναγκαίο χρόνο για να γίνει ένα ποσό 100 δρχ. με επιτόκιο 0.16 μελλοντικά ποσό 120 δρχ., ή το αναγκαίο επιτόκιο για να γίνει ποσό 100 δρχ. μέσα σε 1 χρόνο ποσό

170 δρχ. ή πόσες δραχμές θα είναι ένα ποσό 100 σημερινών δραχμών τοκιζόμενο με 0.80 σε 400 ημέρες.

Επιλέγοντας από το MENU τη λειτουργία "2: ημερομηνίες" μπορούμε να δώσουμε μια ημερομηνία με τη μορφή MM/HH/XX όπου MM μήνας, π.χ. 01 για Ιανουάριο, HH ημέρα, π.χ. 31, και XX χρόνος. Η διαχωριστική κάθετος είναι απαραίτητη. Το σύστημα μας δίνει την ημέρα, π.χ. Κυριακή ή Τρίτη εκείνης της ημερομηνίας και στη συνέχεια μας ρωτά για μια δεύτερη.

Αν δώσουμε μια δεύτερη ημερομηνία μας υπολογίζει το διάστημα σε ημέρες που ακριβώς μεσολαβεί ανάμεσα στις δυο ημερομηνίες.

Το πρόγραμμα αυτό καταλαμβάνει 2160 BYTES και δεν χρειάζεται για τη λειτουργία του πάνω από 600 ακόμα BYTES για πίνακες και μεταβλητές. Ειδικά αυτό το πρόγραμμα μπορεί να μεταφερθεί ευκολότερα σε άλλα συστήματα γιατί δεν κάνει χρήση των ιδιομορφιών του EPSON. Στον EPSON πάντως είναι δυνατό να υπάρχουν μόνιμα στη μνήμη και τα 4 προγράμματα που δίνουμε εδώ και να χρησιμοποιείται κάθε φορά εκείνο που ενδιαφέρει.

```
10 CLS
20 PRINT "1: επιτοκιο":PRINT
  INT "2: ημερομηνιες":INPUT
  CI
30 IF CI=2 GOTO 1000
```

```
40 PRINT "σημερ. αξια":PRINT
  NT "(0 αν αδγνωστη)":INPUT
  PU
45 LPRINT "αξια σημερα":
  PU
50 PRINT "μελλον. αξια":PR
  INT "(0 αν αδγνωστη)":INPU
  TB
55 LPRINT "αξια μελ. ";
  B
60 PRINT "1: ημερες 2: χρον
  ια":PRINT "3: ημερομηνιες"
  :INPUT DD
70 IF DD=3 GOTO 1000
80 IF DD=1 THEN INPUT "ημ
  ερες",N ELSE GOTO 100
90 C=N/365:GOTO 110
100 INPUT "χρονια",C
110 PRINT "επιτοκιο (0 αν
  αδγ.) ":INPUT D
120 IF PU=0 THEN PU=LOG(
  B)-C*LOG(D/100+1):PU=EXP
  (PU):PRINT "σημ. αξια",PU:
  LPRINT "ημ. αξια ":PU:GOT
  O 40
130 IF B=0 THEN B=PU*(1+
  D/100):C:PRINT "μελ. αξια",
  B:LPRINT "μελ. αξια":B:GO
  TO 40
140 IF C=0 THEN C=(LOG(B
  )-LOG(PU))/LOG(D/100+1):
  PRINT "χρονος",C:LPRINT "
  χρονος":C:GOTO 40
150 IF D=0 THEN D=(LOG(B
  )-LOG(PU))/C
160 D=(EXP(D)-1)*100:PRI
  NT "επιτοκιο ",D:LPRINT "
  επιτοκιο ":D
170 INPUT A$:GOTO 10
1000 GOSUB 1470
1010 CLS
1020 PRINT "πρωτη ημερομ.
  (M/D/Y)"
1030 GOSUB 1530
1040 M=DT(1):D=DT(2):Y=D
  T(3)
1050 GOSUB 1180: IF ER T
  HEN 1020
```

```

1060 PRINT IN$:LPRINT:LP
RINT IN$
1070 GOSUB 1390:A1=A
1080 PRINT"δευτερη ημερ.
(M/D/Y)"
1090 GOSUB 1530
1100 M=DT(1):D=DT(2):Y=D
T(3)
1110 GOSUB 1180:IF ER TH
EN 1080
1120 PRINTIN$:LPRINT IN$
1130 GOSUB 1390
1140 N=ABS(A-A1)
1150 PRINT"διαφορα=";N;"
ημερες":LPRINT "διαφορα
";N;" ημερες":IF DD=3 GO
TO 90
1160 PRINT
1170 INPUT A$:GOTO 10
1180 ER=0
1190 ON M GOTO 1220,1240
,1220,1300,1220,1300,122
0,1220,1300,1220,1300,12
20
1200 PRINT "αριθος ημερων
nyta":ER=-1
1210 RETURN
1220 IF D>31 THEN 1200
1230 GOTO 1310
1240 IF Y/4<>INT(Y/4)THE
N 1270
1250 IF Y/400=INT(Y/400)
THEN 1280
1260 IF Y/100<>INT(Y/100
) THEN 1280
1270 IF D>28 THEN 1200
1280 IF D>29 THEN 1200
1290 GOTO 1310
1300 IF D>30 THEN 1200
1310 A=DM%(M-1)
1320 A=A+Y*365+INT(Y/4)+
D+1-INT(Y/100)+INT(Y/400
)
1330 IF INT(Y/4)<>Y/4 TH
EN 1380
1340 IF Y/400=INT(Y/400)
THEN 1360
1350 IF Y/100=INT(Y/100)
THEN 1370
1360 IF M>2 THEN 1380

```

```

1370 A=A-1
1380 RETURN
1390 REM ** ημερα της 68
ομαδας
1410 IF M>2 THEN 1430
1420 M=M+12:Y=Y-1
1430 N=D+2*M+INT(.6*(M+1
)))+Y+INT(Y/4)-INT(Y/100)
+INT(Y/400)+2
1440 N=N MOD 7
1450 PRINT DOW$(N)
1460 RETURN
1470 DIM DOW$(7),DM%(11)
1480 DATA "κυριακη","δευ
τερα","τριτη","τεταρτη",
"πεμπτη","παρασκευη","σα
ββατο"
1490 DATA 0,31,59,90,120
,151,181,212,243,273,304
,334
1500 FOR N=0 TO 6:READ D
OW$(N):NEXT
1510 FOR N=0 TO 11:READ
DM%(N):NEXT
1520 RETURN
1530 IN$=""
1540 X#=INKEY$
1550 IN#=IN#+X$
1560 PRINTX$:
1570 IF X#<>CHR$(13) THE
N 1540
1580 N=0
1590 FOR I=1 TO 3
1600 GOSUB 1640
1610 DT(I)=NB
1620 NEXT
1630 RETURN
1640 NB=0
1650 N=N+1
1660 IF N>LEN(IN#) THEN
PRINT"αθος"
1670 X#=MID$(IN#,N,1)
1680 IF ASC(X#)=>ASC("0"
) AND ASC(X#)<=ASC("9")
THEN NB=NB*10+VAL(X#):ON
ERROR GOTO 10000:GOTO 1
650
1690 IF NB=0 THEN GOTO 1
650 ELSE RETURN
10000 RUN

```

Με τα πλήκτρα 1, 2, 3, 4 κατευθύνουμε το πλοίο αντίστοιχα δεξιά, αριστερά, κάτω και πάνω. Σκοπός του παιχνιδιού είναι να το κατευθύνουμε από μια τυχαία αρχική θέση στο αριστερό χώρισμα της οθόνης. Το συνολικό μήκος του προγράμματος είναι 696 BYTES. Το πρόγραμμα παραχωρήθηκε ευγενικά από τον κ. Γιάννη Λυκούδη.

## ΤΟ ΠΛΟΙΟ



```

9 RANDOMIZE
10 CLS
20 LINE (0,0)-(119,0),PS
ET
30 LINE (119,0)-(119,31)
,PSET
40 LINE (0,31)-(0,0),PSE
T
50 LINE (119,31)-(0,31),
PSET
110 LINE (N,28)-(N,31),P
SET
120 LINE (N+6,28)-(N+6,3
1),PSET
1005 H=RND*114:U=RND*31
1010 A$=INKEY$
1015 IF VAL(A$)=0 THEN 1
020
1016 IF VAL(A$)>4 THEN 1
010
1017 X=VAL(A$)
1020 IF X>4 THEN 1010
1025 GOSUB 4000
1030 ON X GOSUB 2020,201
0,2030,2040,2050
1032 IF H<3 OR H>117 OR
U<4 OR U>31 GOTO 3000
1035 GOSUB 5000

```

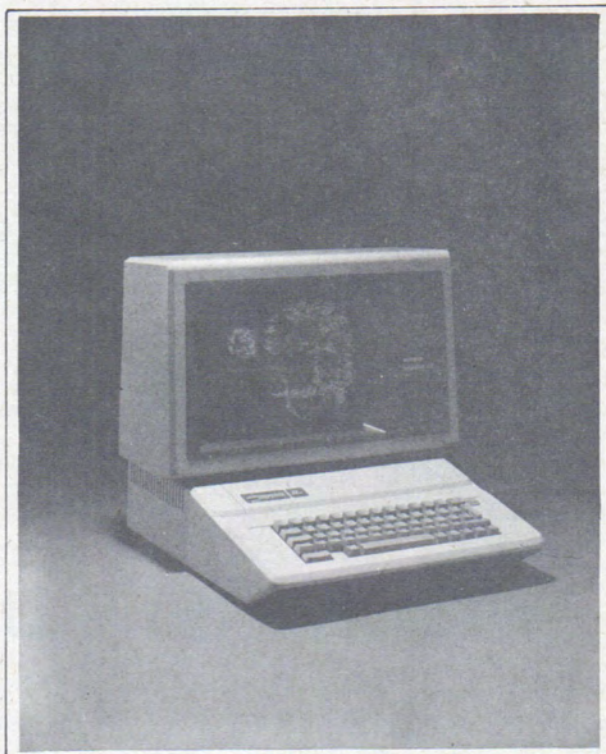
```

1040 GOTO 1010
2010 H=H+1:RETURN
2020 H=H-1:RETURN
2030 U=U+1:RETURN
2040 U=U-1:RETURN
2050 RETURN
3000 IF H>N+2 ANDH<N+3
THEN PRINT"BRAVO":GOTO 3
030
3010 PRINT"YOU LOST!!!!"
3030 FOR I=0 TO 30:SOUND
I,2:NEXT I:GOTO 10
3234 RAN
4000 PRESET(H,U-3)
4010 PRESET(H-1,U-2):PRE
SET (H+1,U-2)
4020 PRESET(H-2,U-1):PRE
SET (H+2,U-1)
4030 PRESET(H,U)
4040 RETURN
4678 RUN
5000 PSET (H,U-3)
5010 PSET (H-1,U-2):PSET
(H+1,U-2)
5020 PSET (H-2,U-1):PSET
(H+2,U-1)
5030 PSET (H,U)
5040 RETURN

```



Προγράμματα  
για τον:



---

apple IIe

---

ΕΠΙΜΕΛΕΙΑ:  
Γιάννης Σγουροβασιλάκης

# MODIFIABLE DATABASE

Το Modifiable Database είναι ένα ειδικό πακέτο δημιουργίας κι εκμετάλλευσης μιας "micro Τράπεζας Πληροφοριών" που μπορεί να χρησιμοποιηθεί εύκολα από τον κάθε χειριστή. Μ' αυτό το πακέτο μπορούμε να δημιουργήσουμε ένα ικανοποιητικό φάσμα εφαρμογών όπως για παράδειγμα παρακολούθηση αλληλογραφίας, ενημέρωση αρχείων βιβλιοθήκης, διατήρηση αρχείου πελατών, λογαριασμών, συνταγών, ανάλυση πωλήσεων, προσωπικό αρχείο κλπ. Βασικά όπως και οι κατασκευαστές του υποστηρίζουν η ποικιλία των εφαρμογών που μπορεί ν' αναπτύξει το πακέτο περιορίζεται μόνο από τις προσωπικές απαιτήσεις και την έκταση της φαντασίας μας.

Το πακέτο "τρέχει" σε οποιοδήποτε σύστημα Apple IIe με τα βασικά χαρακτηριστικά δηλαδή κεντρική μνήμη 64K RAM, οθόνη-πληκτρολόγιο, ένα σύστημα δισκέτας και φυσικά την APPLESOFTBASIC που βρίσκεται ενσωματωμένη στη ROM του συστήματος.

Για την ταξινόμηση των αρχείων και για την επεξεργασία κι αναζήτηση των αποθηκευμένων πληροφοριών χρησιμοποιούνται ταχύτατες ρουτίνες σε γλώσσα μηχανής. Το πακέτο βρίσκεται "φορτωμένο" σε μια δισκέτα των 5 $\frac{1}{4}$ " κι αποτελείται από δυο μέρη. Το πρώτο καθορίζει την δομή της τράπεζας πληροφοριών ενώ το δεύτερο φροντίζει για την εκμετάλλευσή της. Για να "τρέξουμε" το πακέτο "φορτώνουμε" τη δισκέτα στο σύστημα disk II (disk drive του Apple IIe) κι αυτόματα ενεργοποιείται το πρόγραμμα δημιουργίας της Τράπεζας Πληροφοριών. Αρχικά εμφανίζεται πάνω στην οθόνη ένα ειδικό ερωτηματολόγιο (SET UP QUESTIONS) που περιλαμβάνει ερωτήσεις για το όνομα του χειριστή το όνομα που επιθυμούμε να δώσουμε στη Τράπεζα Πληροφο-

ριών που δημιουργούμε, το όνομα του προγράμματος δημιουργίας για τη δισκέτα, σε ποιο I/O port βρίσκεται το interface του εκτυπωτή, τον αριθμό των χρησιμοποιούμενων συστημάτων δισκέτας, τον αριθμό των χαρακτήρων της κάθε εγγραφής μας (record), τον αριθμό των χαρακτήρων που υπάρχουν σε κάθε εγγραφή, τον αριθμό και τα ονόματα των πεδίων της εγγραφής κλπ. Ο χειριστής μπορεί να δει και να μεταβάλει οποιαδήποτε από τις παραπάνω πληροφορίες πριν από την τελική τους καταχώρηση. Αφού δοθούν οι παραπάνω πληροφορίες ο χειριστής καλείται από ένα κατάλληλο μήνυμα που εμφανίζεται στην οθόνη να βάλει μια "άγραφη" δισκέτα στο δεύτερο disk drive. Τότε το πρόγραμμα ενεργοποιείται κι αποθηκεύει το αρχείο με τα νέα δεδομένα στην κενή δισκέτα. Έτσι τώρα όταν καλείται το πρόγραμμα τρέχει με τα

νέα δεδομένα. Φυσικά μπορούν να δημιουργηθούν παρόμοια αρχεία δεδομένων για ένα μεγάλο πλήθος εφαρμογών. Κάθε όμως αρχείο πρέπει ν' αποθηκεύεται σε μια ξεχωριστή δισκέτα. Ο αριθμός και το μήκος των πεδίων των διαφόρων εγγραφών του αρχείου παίζει ένα καθοριστικό ρόλο για την κάθε εφαρμογή. Θα πρέπει να επιλέγουμε αριθμό πεδίων και μήκη που πραγματικά χρειαζόμαστε στη κάθε εγγραφή για να εκμεταλλευόμαστε έτσι όσο το δυνατόν καλύτερα το διαθέσιμο χώρο μνήμης. Αν για παράδειγμα ορίσουμε ένα πολύ μεγάλο record μεγέθους 250 χαρακτήρων χωρίς ουσιαστικά να εκμεταλλευόμαστε τότε όλο το record περιορίζουμε σημαντικά το συνολικό αριθμό records που μπορούμε να έχουμε στη διάθεσή μας που είναι γι' αυτή την περίπτωση 90. Αν πάλι έχουμε μια εφαρμογή που επεξεργάζεται εγγραφές (records) των 10 χαρακτή-

90. Αν πάλι έχουμε μια εφαρμογή που επεξεργάζεται εγγραφές (records) των 10 χαρακτή-

ρων τότε ο συνολικός αριθμός των εγγραφών που θα έχουμε στη διάθεσή μας είναι 2268. Το πρόγραμμα δημιουργίας της Τράπεζας Πληροφοριών επιτρέπει την ύπαρξη 1-22 πεδίων για κάθε εγγραφή με μήκος πεδίων από 1-49 χαρακτήρες. Οι παραπάνω περιορισμοί εφαρμόστηκαν για την καλύτερη εμφάνιση των δεδομένων πάνω στην οθόνη και για την ευκολότερη μετατροπή τους. Αφού τελειώσουμε το στάδιο καθορισμού και δημιουργίας της Τράπεζας Πληροφοριών "τρέχουμε" το πρόγραμμα εκμετάλλευσης (MODIFIED PROGRAM) το οποίο εμφανίζει αρχικά το κυρίως menu με τις παρακάτω δυνατές επιλογές:

1. START A NEW FILE
2. ENTER RECORDS
3. LIST / MODIFY
4. SEARCH
5. SORT
6. PRINT
7. DISK ACCESS
8. TERMINATE

Αν διαλέξουμε την πρώτη επιλογή #1 τότε μπορούμε να ορίσουμε ένα νέο αρχείο δεδομένων. Όλα τα προηγούμενα δεδομένα που τυχόν υπάρχουν στη μνήμη σβήνονται. Για λόγους ασφάλειας όταν πατήσουμε το #1 που ενεργοποιεί την πρώτη επιλογή εμφανίζεται στην οθόνη ένα προειδοποιητικό μήνυμα για να μας επιστήσει την προσοχή ότι τα προηγούμενα αρχεία δεδομένων πρόκει-

ται να διαγραφούν και να προλάβει πιθανή αφηρημάδα μας. Με τη δεύτερη επιλογή (#2) ENTER RECORDS μας δίνεται η δυνατότητα να εισάγουμε νέες εγγραφές. Με την επιλογή LIST / MODIFY (#3) μπορούμε να εμφανίσουμε το αρχείο στην οθόνη αρχίζοντας από οποιαδήποτε εγγραφή επιθυμούμε ή ακόμα να τροποποιήσουμε οποιαδήποτε εγγραφή του αρχείου. Όταν λοιπόν ο χειριστής πατήσει #2 καλείται να δώσει τον αριθμό ενός record που θα χρησιμοποιηθεί σαν αρχικό. Τότε το συγκεκριμένο record εμφανίζεται πάνω στην οθόνη μαζί μ'ένα βοηθητικό menu που περιλαμβάνει τις παρακάτω δυνατές επιλογές.

RET: Δηλαδή πάτημα του πλήκτρου επιστροφής (RETURN) για να εμφανιστεί στην οθόνη, η επόμενη εγγραφή.

ESC: Αν πατήσουμε ESCAPE τότε επιστρέφουμε στο κυρίως menu.

"E" (EDIT): Για να κά-  
νουμε αλλαγές σε πεδιά.

"D" (DELETE): Για να διαγράψουμε τη συγκεκριμένη εγγραφή από το αρχείο μας.

"P": Για να τυπώσουμε μια ή περισσότερες εγγραφές ή ακόμα κι ολόκληρο τ'αρχείο.

"I" (SCROLL). Η εντολή αυτή χρησιμοποιείται μόνο στη περίπτωση που το record περιέχει παραπάνω από 8 πεδία

οπότε μ'αυτή την εντολή εμφανίζονται στην οθόνη τα επόμενα 8 πεδία της εγγραφής.

Η επιλογή SEARCH (#4) του κυρίως menu μας επιτρέπει την αναζήτηση μιας οποιασδήποτε εγγραφής ή όλων των εγγραφών που περιέχουν ένα συγκεκριμένο πεδίο ή υποπεδίο του αρχείου μας ή απλά μιας συγκεκριμένης εγγραφής ή ακόμα και όλων των εγγραφών του αρχείου. Πριν γίνει οτιδήποτε ο χειριστής ερωτάται αν οι εγγραφές που αναζητούνται κι εμφανίζονται πάνω στην οθόνη πρόκειται να τροποποιηθούν ή απλά να τυπωθούν. Με το πάτημα του #5 ενεργοποιείται η πέμπτη επιλογή με την οποία γίνεται ταξινόμηση μόνο του "τρέχοντος" αρχείου δεδομένων κι όχι όλων των αρχείων δεδομένων που υπάρχουν στη δισκέτα. Η ταξινόμηση μπορεί να γίνει ως προς 1 μέχρι 3 διαφορετικά πεδία. Με την PRINT (#6) μπορούμε να εκτυπώσουμε οποιοδήποτε record του αρχείου ή ακόμα κι όλο το αρχείο. Η επιλογή DISK ACCESS (#7) μας επιτρέπει να "φορτώσουμε" ένα νέο αρχείο δεδομένων πάνω στη δισκέτα να αποθηκεύσουμε ή να διαγράψουμε αρχεία από τη δισκέτα. Με την ίδια επιλογή μπορούμε ακόμα να εμφανίσουμε στην οθόνη ένα κατάλογο με τα περιεχόμενα

των δυο δισκετών. Τέλος η επιλογή TERMINATE (#8) χρησιμοποιείται για να τερματίσει το πρόγραμμα εκμετάλλευσης της Τράπεζας Πληροφοριών. Αν αυτό γίνει χωρίς να έχουμε αποθηκεύσει τις τυχόν αλλαγές που έχουμε κάνει μας εμφανίζεται πάνω στην οθόνη ένα προειδοποιητικό μήνυμα. Αν τώρα θέλουμε να "ξανατρέξουμε" το πρόγραμμα δεν έχουμε παρά να τυπώσουμε "GOTO 19600" και να πατήσουμε το πλήκτρο επιστροφής. Αν σ'οποιαδήποτε φάση θέλουμε να διακόψουμε μια συγκεκριμένη επιλογή πατάμε το πλήκτρο ESC (ESCAPE).

Θέλουμε να τονίσουμε στο σημείο αυτό ότι η ταξινόμηση του αρχείου γίνεται με αξιοθαύμαστη ταχύτητα. Αυτό οφείλεται στο γεγονός ότι οι ρουτίνες που χρησιμοποιούνται για την ταξινόμηση είναι γραμμένες κατευθείαν σε γλώσσα μηχανής απ'την μια μεριά κι απ'την άλλη επειδή το πακέτο MODIFIABLE DATABASE επεξεργάζεται τα αρχεία δεδομένων ON-LINE δηλαδή υπάρχουν στην άμεση διάθεση του χειριστή 22.685 bytes για την καταχώρηση των δεδομένων του.

Αναφέρουμε τελευταία ότι το πακέτο συνοδεύεται από ένα ευανάγνωστο επεξηγηματικό εγχειρίδιο που βοηθάει τον χειριστή στη καλύτερη χρησιμοποίηση κι αξιοποίηση του πακέτου.

## MODIFIABLE DATABASE

### Πρόγραμμα Δημιουργίας

```

LOAD SETUP PROGRAM
LIST
110 PRINT CHR$(4)MAXFILES 3*
115 CHR$(20)GOTO 999
120 GOTO 250
121 IF C$ < " " THEN I = ASC(C$): IF I < 27 THEN INVERSE: PRINT CHR$(I + 64): NORMAL
122 PRINT B$;: RETURN
125 I = ASC(C$): B$ = C$: IF I < 27 THEN 24 = CHR$(I + 64): B$ = B$ + I;: LG% = LEN(B$): LG% = LG% - (LG% > 1): B$ = RIGHT$(B$, LG%)
126 RETURN
127 PRINT "IF THIS IS INCORRECT, PRESS ESC TO EDIT": RETURN
130 GET A$: PRINT A$: B$ = A$: IF B$ < " " THEN 130
140 RETURN
150 POKE 35,24: INVERSE: UTAB 21: HTAB 1: CALL - 958: HTAB 10: PRINT 25
155 IF U1 > N% THEN V1: PRINT "CHARACTERS LEFT": HTAB 4: PRINT "RECORD TO CONTAIN INDEX FILES": RETURN
155 V1 = 1 + P% * 16: IF V1 < 19 THEN RETURN
156 IF U1 > N% THEN V1: PRINT "RECORD TO CONTAIN INDEX FILES": RETURN
157 POKE 24,24: B$ = C$: IF B$ < " " THEN 157
160 PRINT M: NEXT M: FOR M = 1 TO 18: HTAB 9: PRINT CHR$(M): NEXT M
161 PRINT M: NEXT M: HTAB 3: FOR M = 1 TO 18: HTAB 2 + M: HTAB 11: PRINT AX(M): HTAB 25: PRINT H$(M): NEXT M: RETURN

```

```

158 FOR M = 1: HOME: FOR N = 19 TO 41: HTAB 3: PRINT CHR$(M + 55): HTAB 11: PRINT AX(M): HTAB 25: PRINT H$(M): NEXT M: RETURN
160 A$ = "": J1% = "": J1% = 1
170 HTAB X1% + J1%: GET A$
180 IF A$ = CHR$(27) AND B$ < A$ THEN 250
200 IF A$ = CHR$(8) OR A$ = CHR$(27) THEN RETURN
210 IF A$ = CHR$(8) THEN J1% = J1% - 1: PRINT " ": IF J1% < 2 THEN J1% = 1: A$ = "": GOTO 170
220 IF A$ = CHR$(8) THEN A$ = LEFT$(A$, J1% - 1): POKE 36, PEEK(36) - 2: PRINT " ": POKE 36, PEEK(36) - 2: GOTO 170
230 IF J1% > FLX THEN J1% = FLX + 1: PRINT " ": GOTO 170
235 IF ASC(A$) < 27 THEN INVERSE: PRINT CHR$(ASC(A$) + 64): NORMAL
240 PRINT A$: A$ = A$ + A$: J1% = J1% + 1: GOTO 170
250 DIM AX(100), H$(100): D$ = CHR$(4)
260 TEXT: HOME: UTAB 10: HTAB 8: PRINT "THE MODIFIABLE DATABASE": PRINT: HTAB 5: PRINT "BY ARR SMI, CLARY & CHRIS ANDSON": GOSUB 950: UTAB 25: PRINT "COPYRIGHT 1980 BY SYNERGISTIC SOFTWARE": GOSUB 950
270 UTAB 17: HTAB 1: CALL - 958: PRINT "NOTE: THIS DISK IS A MASTER DISK WHICH SHOULD BE USED ONLY TO SETUP WORKING DISKS WHICH HAVE BEEN MODIFIED TO SUIT YOUR NEEDS. THIS PROGRAM WILL PERFORM THE NECESSARY MODIFICATIONS. YOU WILL NEED A BLANK DISKETTE WHICH HAS BEEN INITIALIZED WITH AN A PLESET GREETING PROGRAM ON WHICH TO STORE YOUR NEW PROGRAM. IF YOU DO NOT HAVE SUCH A DISKETTE READY OR IF YOU HAVE NOT READ THE DOCUMENTATION ACCOMPANYING THIS DISKETTE, YOU SHOULD A SORT THE PROGRAM AT THIS TIME BY PRESSING ESC."
280 PRINT "PRESS A$G$ RETURN WHEN READY TO CONTINUE."
290 GET A$: IF A$ = CHR$(27) THEN 250
310 HOME: UTAB 5: PRINT "THIS DATA BASE PROGRAM ALLOWS THE USER TO MODIFY THE PROGRAM'S BASIC CHARACTERISTICS TO FIT HIS PARTICULAR NEEDS. YOU WILL SPECIFY THE NUMBER AND SIZE OF ALL FIELDS, THEIR HEADINGS, AND

```



```
3400 RM% = 1 : & P,N% + 1:N% = N% + 1 : GOSUB 4800 : GOTO 3300
3500 REM
3510 HOME : FLX = FOX(1) : PRINT "ENTER DATA FOR:" PRINT H%(1) : FIELD "I
IX = 1 : GOSUB 1900:IX = 0
3520 & L,F%(1) = AS
3600 RETURN
4700 GOSUB 15600
4710 & G,JJ : PRINT
4712 POKE 32,15 : POKE 33,25 : POKE 34,1 : VTAB 2 : HTAB 16 : FOR II = PCX * 8
TO PCX * 8 + J : VTAB 3 + 2 * II - 16 * PCX : HTAB 16 : A1% = F%(1) : GOSUB
2295 : NEXT II : TEXT : POKE 34,20 : IF MDX = 3 THEN RETURN
4800 HOME : VTAB 17 : HTAB 25 : PRINT "PRINTING INDEX" : HTAB 25 : PRINT "ESC
RTN TO MENU" : HTAB 25 : PRINT "E=EDIT" : HTAB 25 : PRINT "D=DELETE" :
HTAB 25 : PRINT "P=PRINT" : IF NFX > 8 THEN HTAB 25 : PRINT "S=SCR
OLL"
4810 IF SH% THEN VTAB 21 : HTAB 1 : PRINT "SEARCH IN PROCESS"
4825 GOSUB 3040 : VTAB 21 : IF AS = "/" THEN PCX = PCX + 1 : GOTO 4700
5000 IF AS = "E" THEN TZ% = JJ : GOSUB 16000:JJ = TZ% : & P,JJ:RM% = 1 : GOTO
4800
5020 IF AS < "0" THEN 5250
5050 PRINT "DELETE RECORD (Y/N)?" : GOSUB 3100 : IF AS < "Y" THEN VTAB
19 : CALL - 950 : GOTO 4800
5100 IF JJ < N% THEN FOR I = JJ TO N% : I & G,1 : I & P,1 : NEXT I
5150 N% = N% - 1:RM% = N% / 250 : POKE NRX + 1,1:RM%:RM% = N% - 250 * RM% : POKE
NRX,RM%:RM% = PEEK (RFX + 1) * 256 + PEEK (RFX):RM% = RM% - (RFX -
0) : IX = RM% / 256 : POKE RFX + 1,IX:IX = RM% - IX * 256 : POKE RFX,IX:J
J = JJ - 1:RM% = 1 : RETURN
5250 IF AS = "P" THEN GOSUB 12550 : GOSUB 12250 : PRINT D%:PRNO : TEXT : POKE
34,20 : GOTO 4800
5300 IF ASC (AS) < 13 THEN 4800
5400 RETURN
5500 GOSUB 2950
5600 PCX = 0 : GOSUB 4700:JJ = JJ + 1 : IF JJ < N% + 1 THEN 5000
5910 GOTO 19500
PRINT
PRINT "DISPLAY RESULTS"
2)...PRINT RESULTS
WHICH?MX = 2 : GOSUB 3100 : IF IX = 2 THEN SP1X = 1
6030 VTAB 11 : CALL - 950 : PRINT "
1)...CURRENT FILE ONLY
2)...ALL FILES DISK 1
3)...ALL FILES DISK 2
4)...ALL FILES BOTH DISKS
SELECT SEARCH RANGE"
6040 MX = 4 : GOSUB 3100:RM% = IX : IF RM% < 1 AND RM% THEN VTAB 21 : GOSUB
2990
6045 IF RM% = 1 THEN GOSUB 2600
6050 VTAB 11 : CALL - 950 : PRINT "
1)...SEARCH FOR A SUBSTRING
2)...SEARCH FOR ONE FIELD
3)...SEARCH FOR TWO FIELDS
WHICH SEARCH OPTION?MX = 3
6100 GOSUB 3100:SRX = IX:RCX = RFX + 1
6110 GOSUB 15500:SRX = 1 : VTAB 21 : PRINT "SEARCH WHICH FIELD?" : GOSUB 30
50:KK = 0
6200 HOME : PRINT "ENTER SEARCH KEY" : PRINT "FOR " : H%(1) : " FIELD." : FLX =
FOX(1) : POKE 33,25 : GOSUB 1400 : POKE 33,40:BS = AS : IF KK THEN II =
FOX(BS) : A = 1 : F%(KK) = AS:CS = F%(KK) : GOTO 11000
6210 IF SRX < 3 THEN 11000
6215 HOME : PRINT "SECOND FIELD" : MX = 11:Z% = AS : GOSUB 3050:KK = 11 : GOTO
6200
6300 GOSUB 2900 : PRINT "
SELECT 1, 2, OR 3 FIELDS TO SORT BY" : MX = 3 : GOSUB 3100 : GOSUB 15590
6350 FOR FL = 1 TO IX
6360 VTAB 21 + FL : HTAB 1 + FL : PRINT "SELECT SORT FIELD " : IFL : GOSUB 3
050 : VTAB 21 + FL : HTAB 25 + FL : PRINT H%(1)
6370 BRFL = 1 : IF BRFL < 3 THEN 11000:FL = 2 : IFL = NEXT FL
7000 & X,F%(B%(1)),F%(B%(2)),F%(B%(3))
8400 HOME : PRINT : PRINT " DO YOU WANT A PRINTER LISTING (Y/N)?" : GOSUB
3040
8415 IF AS < "Y" THEN 19500
8425 TEXT : HOME : GOTO 11800
11000 BS = BS : ON RM% GOTO 11010,11020,11030,11030,11020
11010 GOSUB 11000 : GOSUB 2900 : GOTO 19500
11020 DRX = 1 : GOSUB 2900
11030 IF RM% > 2 THEN DRX = 2 : GOSUB 2980
11035 GOSUB 2900 : GOTO 19500
11090 BS = BS : POKE RFX + 1,0 : POKE RFX,0
11100 & L,F%(1) = BS
11200 SX% = 256 * PEEK (RFX + 1) + PEEK (RFX) : & S,F%(1),SX% + 1: SX% =
256 * PEEK (RFX + 1) + PEEK (RFX)
11300 IF SX% < 1 OR SX% > N% THEN 11250
11320 JJ = SRX : IF SRX > 2 THEN & G,0,0 : IF BS < "P" THEN 11400
11330 MX = 11 : IF SP1X THEN GOSUB 12030 : PRINT D%:PRNO
11340 IF SP1X = 0 THEN GOSUB 4700 : GOSUB 2970
11345 II = MX : IF SX% < N% THEN 11100
11350 IF SRX = 1 THEN BS = " " : BS : IF LEN (BS) <= FOX(1) THEN POKE
RFX,0 : POKE RFX + 1,0 : GOTO 11100
11400 PRINT D%:PRNO : TEXT : HOME : VTAB 10 : PRINT " NO MORE RECORDS FOR
ND IN THIS FILE" : RETURN
11600 REM
11810 PRINT "
1)...PRINT ONE RECORD
2)...PRINT STARTING AT...
3)...PRINT ALL RECORDS THIS FILE
4)...PRINT ALL RECORDS DISK 1
5)...PRINT ALL RECORDS DISK 2
6)...PRINT ALL RECORDS BOTH DISKS
WHICH?"
11850 MX = 6 : GOSUB 3100:RM% = IX
11860 IF RM% < 4 THEN GOSUB 2600
11900 JJ = 1 : IF RM% < 3 THEN VTAB 14 : CALL - 950 : PRINT : GOSUB 2950
11880 IF RM% > 3 AND RM% THEN VTAB 14 : GOSUB 2970
11900 VTAB 14 : CALL - 950 : VTAB 20 : HTAB 7 : PRINT "PRESS ESC. TO HALT PR
INTOUT:DRX = 1 : ON RM% GOTO 11210,11220,11220,11240,11250,11240
11910 GOSUB 12000 : GOTO 11925
11920 GOSUB 12030 : IF JJ < N% THEN JJ = JJ + 1 : GOTO 11920
11925 GOSUB 12250 : GOTO 19500
11940 N% = 4 : GOSUB 2950 : IF RM% < 4 THEN 11925
11950 DRX = 2:RM% = 4 : GOTO 11940
12030 IF PEEK (- 16384) = 155 OR FX THEN POKE - 16384,0 : PRINT D%:PRNO
0 : POKE 34,19 : POKE 33,40 : HOME : FLX = 1 : JJ = GOSUB 1300 : JJ = 1 : VTAB
14 : CALL - 950 : VTAB 20 : HTAB 7 : PRINT "PRESS ESC. TO HALT PRINTOUT"
```

```
12055 GOSUB 950
12070 & B,JJ : PRINT "
...RECORD NUMBER " : JJ : *****
12100 FOR II = 0 TO NFX - 1 : PRINT H%(1) : SPC (10 - LEN (H%(1))) : "
```

```
11 IF LEN (F%(1)) < 25 THEN PRINT F%(1) : GOTO 12200
12180 PRINT "LEFTS " : F%(1),24 : PRINT "SPC 15" : PRINT "RIGHTS " : F%(1), LEN
(F%(1)) - 24
12200 NEXT II : PRINT CHR (10) : RETURN
12250 PRINT
*****
: RETURN
15400 IF RM% THEN TEXT : HOME : VTAB 10 : PRINT " SAVE WORKING FILE FI
RST (Y/N)?" : GET AS : IF AS < "Y" THEN 19500
15480 VTAB 25 : HTAB 1 : PRINT "END (Y/N)?" : GET AS : IF AS = "Y" THEN END
15500 GOTO 19500
15590 GOSUB 15600 : GOSUB 15800 : IF NFX > 8 AND MDX < 3 THEN VTAB 21 : HTAB
27 : PRINT "S= SCROLL"
15595 RETURN
15600 TEXT : POKE 35,20 : HOME : POKE 34,2 : HOME : INVERSE
15605 IF PCX < 0 : NFX = 1 THEN PCX = 0
15609 J = NFX - 1 : PCX = 0 : IF J > 7 THEN J = 7
15610 FOR I = PCX * 8 TO J + PCX * 8 : HTAB 5 : PRINT "SPC (10) : HTAB 5 : PRINT
H%(1) : HTAB 16 : CALL - 868 : PRINT : PRINT : NEXT I
15700 POKE 35,24 : POKE 34,20 : NORMAL : VTAB 13 : HTAB 13 : IF JJ THEN PRINT
"RECORD " : JJ : IF PCX THEN PRINT "CONT"
15750 RETURN
15800 VTAB 3 : HTAB 11 : J = NFX - PCX * 8 : IF J > 8 THEN J = 8
15805 FOR I = PCX * 0 + 1 TO PCX * 0 + J : IF I < 10 THEN PRINT I :
15810 IF I < 9 THEN PRINT CHR (1 + 55)
15820 PRINT " : " : PRINT : NEXT I : CALL - 868 : RETURN
16000 MX = 2 : GOSUB 15800 : GOSUB 2860
*****
16290 I = INT ((1 / 5) * I) : PCX THEN PCX = 1 : GOSUB 18590
16400 CSX = 1 : MDX = 1 : GOSUB 3500:CSX = 0 : MDX = 0
16450 FOR I = 3 TO 17 STEP 2 : VTAB 1 : HTAB 1 : PRINT "SPC (4) : NEXT I
16500 RETURN
18800 BS = "THE MODIFIABLE DATA BASE" : IN% = "NOBODY"
18800 TEXT : HOME : VTAB 8 : HTAB (20 - LEN (BS) / 2) : PRINT D% : VTAB 11 :
HTAB 6 : PRINT "BY R.C. CLARDY & C.P. ANDERSON" : PRINT : HTAB 14 - LEN
(BS) : PRINT "REVISED BY J.M. GOSUB 2900:BS = CHR (4) : VTAB 23
: PRINT "COPYRIGHT 1980 BY SYNCRISTIC SOFTWARE"
19000 PRINT D%:BLOAD:BS,AS:9400 : POKE 1013,76 : POKE 1014,0 : POKE 1015
,148 : POKE NRX,0 : POKE NRX + 1,0:BS = D% + "OPENFILES,L20" : I% = D% +
"WRITEFILES,R" : BS = D% + "CLOSE" : BS = D% + "READFILES,R"
19200 RM% = 228:RLX = 100:SNX = 2:INFX = 10 : DIM F%(20),F%(20),H%(20)
19210 & F,F% = F%(0),10:F%(1),10:F%(2),10:F%(3),10:F%(4),10:F%(5),10:F%(6
),10:F%(7),10:F%(8),10:F%(9),10
19220 FOR I = 0 TO 9:INFX = 10 : NEXT I
19250 FOR I = 0 TO 9:H%(1) = "FIELD " : CHR (1 + 49) : NEXT I
19500 PRINT D% : PRINT BS : PRINT D%:PRNO
19550 ONERR GOTO 19570
19580 FOR I : GOTO 19540
19590 POKE 21,0 : ONERR GOTO 45000
19600 TEXT : JJ = 0 : SRX = 0 : FX = 0 : SP1X = 0 : SX% = 0 : J1X = 0 : PCX = 0 : CSX =
0 : MDX = 0 : N% = 256 * PEEK (NRX + 1) + PEEK (NRX) : POKE RFX,0 : POKE
RFX + 1,0
19700 HOME : HTAB 10 : PRINT "MAIN MENU
1)...START A NEW FILE
2)...ENTER RECORD
3)...LIST/MODIFY
4)...SEARCH
5)...SORT
6)...PRINT
7)...DISK ACCESS
8)...TERMINATE"
19900 VTAB 23 : PRINT " WHICH?" : HTAB 15 : INVERSE : PRINT N% : NORMAL : PRINT
" OF " : RX% : " RECORDS USED" : IF I% < 0 THEN HTAB 24 - LEN (I%) : PRINT
"CURRENT FILE IS " : INVERSE : PRINT I%
19905 IF RM% THEN VTAB 21 : HTAB 1 : INVERSE : PRINT "
RECORDS HAVE B
LEN MODIFIED.
19907 MX = 8
19910 NORMAL : GOSUB 3100 : VTAB 2 + IX + 3 : CALL - 950 : IF IX = 3 OR IX =
8 THEN GOSUB 2600
20100 RM% = 0 : ON IX GOTO 3280,3300,5600,6000,6300,11800,30000,15400
30000 MDX = 2 : PRINT " DISK DRIVE 1 OR 2 (DEFAULT=1)?" : GOSUB 3040:DRX =
IX : IF DRX < 2 THEN DRX = 1
30015 VTAB 17 : CALL - 868 : PRINT " 1)...LOAD
2)...SAVE
3)...DELETE
4)...CATALOG
SELECT WHICH?MX = 4 : GOSUB 3100
30017 IF IX < 4 THEN CALL - 936 : PRINT D%:CATALOG,D%:DRX : PRINT : INVERSE
: HTAB 10 : PRINT " PRESS RETURN " : GET AS : NORMAL : GOTO 19500
30020 VTAB 17 : CALL - 950 : VTAB 17 : INVERSE : PRINT " READY TO LOAD
" : IF IX = 3 THEN VTAB 17 : HTAB 13 : PRINT "DELETE"
30021 IF IX = 2 THEN VTAB 17 : HTAB 13 : PRINT "SAVE"
30022 PRINT " IF IX = 1 AND RM% THEN GOSUB 2970:IX = 1 : VTAB 10 : CALL
950 : PRINT
30023 NORMAL : PRINT "INPUT FILE NAME (RETURN FOR CATALOG) : IF I% < 0
THEN PRINT "PRESS " : FOR FILE " : I%
30030 PRINT "FLX = 1" : GOSUB 1400 : IF AS = " " THEN HOME : PRINT D%:CATAL
OG,D%:DRX : PRINT : GOTO 30025
30035 IF IX < 3 THEN I% = AS
30040 ON IX GOTO 30100,30200,30300
30100 VTAB 14 : CALL - 950 : VTAB 10 : HTAB 14 - LEN (AS) / 2 : INVERSE : PRINT
"LOADING " : AS : " : NORMAL
30150 PRINT D%:BLOAD" : AS : " : D%:DRX : POKE NRX, PEEK (37886) : POKE NRX + 1, PEEK
(37887):RM% = 0 : GOTO 19400
30200 VTAB 14 : CALL - 950 : VTAB 10 : HTAB 14 - LEN (AS) / 2 : INVERSE : PRINT
"SAVING " : AS : " : NORMAL
30250 GOSUB 40000 : GOSUB 40005 : IF J THEN GOSUB 40010
30254 IF FX = 0 THEN PRINT W% : J - 1 : THEN I% : PRINT W% : 0 : PRINT J + 1 -
30255 GOTO 19500
30300 PRINT : PRINT " DELETE " : AS : " FILE (Y/N)?" : BS = AS : GOSUB 3040 : IF
AS < "Y" THEN 19400
30320 AS = BS : VTAB 16 : CALL - 950 : VTAB 18 : HTAB 14 - LEN (AS) / 2 : INVERSE
: PRINT "DELETING " : AS : " : NORMAL
30380 HTAB 11 : CALL - 868 : PRINT "DELETE " : AS : " : D%:DRX : GOSUB 40005 : IF
J THEN GOSUB 40010 : IF FX THEN FOR I = FX TO J - 1 : PRINT R% : I + 1 :
INPUT BS : PRINT W% : I : PRINT BS : NEXT I : PRINT W% : 0 : PRINT J - 1
30351 GOTO 19500
40000 POKE 37886, PEEK (NRX) : POKE 37887, PEEK (NRX + 1) : PRINT D%:BSAVE
" : AS : 37885 - N% * RLX : " : L% : N% * RLX + 3 : D%:DRX : RM% = 0 : RETURN
40005 PRINT D% : PRINT R% : 0 : INPUT J : PRINT BS : RETURN
40010 FX = 0 : PRINT D% : FOR I = 1 TO J : PRINT R% : INPUT BS : IF BS = AS THEN
FX = 1
40020 NEXT I : RETURN
43000 IF PEEK (222) < 42 AND PEEK (222) < 5 THEN 43005
43001 PRINT BS : PRINT D% : PRINT W% : 0 : PRINT D% : POKE 21,0:FX = 0 : JJ = 0 : IF
N% > 2 THEN 30254
43005 TEXT : HOME : VTAB 10 : PRINT "ERROR CODE NUMBER " : PEEK (222) : POKE
21,0 : HTAB 5 : PRINT "PRESS RETURN" : GET AS : GOTO 19500
```

# APPLE PROMS

Με το πρόγραμμα APPLE PROMS, που είναι κι' αυτό γραμμένο σε APPLESOFT BASIC, μπορούμε να ενημερώνουμε μια μικρή αποθήκη. Με την εκτέλεση της εντολής RUN εμφανίζεται πάνω στην οθόνη το βασικό Menu του προγράμματος με τρεις δυνατές επιλογές. Εμφάνιση των ειδών της αποθήκης και των ποσοτήτων από κάθε αποθηκευμένο είδος, ενημέρωση ενός ή περισσοτέρων ειδών ή ακόμα κι όλων, ή τέλος τον τερματισμό του προγράμματος. Και το πρόγραμμα αυτό μπορεί επίσης να χρησιμοποιηθεί για οικιακή χρήση ή για χρήση γραφείου ή ακόμα σε μια μικρή επιχείρηση.

```
JPR#0
JLIST

5 PRINT "NOMONI,0,C": HOME
10 TEXT :D$ = CHR$ (4)
20 LET OP$ = D$ + "OPEN "
30 LET CL$ = D$ + "CLOSE "
40 LET RD$ = D$ + "READ "
50 LET WR$ = D$ + "WRITE "
60 LET FL$ = "APPLE PROMS"
70 PRINT OP$;FL$;";L40"
74 GOSUB 9000
80 GOSUB 390
90 ON Q GOTO 100,180,480
100 GOSUB 330
110 FOR R = R1 TO R2
120 PRINT RD$;FL$;";R";R
130 INPUT N$,BL,BW,ST
140 PRINT " ";R; TAB( 8);N$; TAB( 24);BL; TAB( 32);ST
150 NEXT R
160 PRINT D$
170 GOTO 310
180 GOSUB 330
190 LET T = 7: FOR R = R1 TO R2:T = T + 1
200 PRINT RD$;FL$;";R";R: INPUT N$,BL,BW,ST: PRINT D$
210 VTAB (T): PRINT " ";R; TAB( 8);N$;: HTAB (7): INPUT Q$
220 IF LEN (Q$) > 15 THEN 200
230 IF LEN (Q$) < > 0 THEN N$ = Q$
250 VTAB (T): HTAB (24): PRINT BL;: HTAB (23): INPUT Q$: IF LEN (Q$) < > 0 THEN BL = VAL (Q$)
270 VTAB (T): HTAB (32): PRINT ST;: HTAB (31): INPUT Q$: IF LEN (Q$) > 0 THEN ST = VAL (Q$)
280 VTAB (T): PRINT " ";R; TAB( 8);N$; TAB( 24);BL; TAB( 32);ST;" "
290 PRINT WR$;FL$;";R";R: PRINT N$;";";BL;";";BW";";ST
294 CALL 768: PRINT
300 PRINT D$: NEXT R
310 VTAB (23): PRINT "PRESS THE RETURN KEY TO CONTINUE.": GET Q$
320 GOTO 80
330 PRINT : INPUT "PART NUMBER 1-9 (0=ALL) ";Q$
340 PRINT Q$:Q = VAL (Q$): IF (Q < 1 OR Q > 9) AND Q$ < > "0" THEN PRINT CHR$ (7);: GOTO 330
350 LET R1 = Q:R2 = Q: IF Q = 0 THEN R1 = 1:R2 = 9
360 HOME : VTAB (5)
370 PRINT "PART# NAME SIZE IN STOCK ----"
-----
380 RETURN
390 HOME : PRINT TAB( 12);FL$: VTAB (10)
400 PRINT "COMMAND","NUMBER"
410 PRINT "-----","-----"
420 PRINT "LIST"," 1"
430 PRINT "CHANGE"," 2"
440 PRINT "EXIT"," 3"
450 PRINT : INPUT "CHOOSE NUMBER (1 - 3) ";Q$:Q = VAL (Q$)
460 IF Q > 0 AND Q < 4 THEN RETURN
470 VTAB (15): PRINT CHR$ (7);: GOTO 450
480 PRINT CL$;FL$
490 HOME : END
500 REM DEMONSTRATION OF RANDOM ACCESS
9000 FOR I = 768 TO 775: READ J: POKE I,J: NEXT
9010 RETURN
9100 DATA 169,0,32,237,253,76,142,253
```

# ANIMALS

Πρόκειται για ένα έξυπνο διασκεδαστικό "πνευματικό" παιχνίδι του Apple IIe. Με την αρχή του παιχνιδιού εμφανίζονται στην οθόνη αναλυτικές οδηγίες που βοηθούν τον παίχτη να μάθει γρήγορα και εύκολα το παιχνίδι. Το πρόγραμμα καλεί τον παίχτη να βάλει ένα είδος ζώου στο μυαλό του και ο υπολογιστής με κατάλληλες ερωτήσεις προσπαθεί να το βρει. Αν αποτύχει παραδέχεται σαν καλός αντίπαλος την ήττα του, ζητά διευκρινίσεις για το είδος που απέτυχε να βρει και καταχωρεί το γνωστό πια για αυτόν είδος ζώου στην μνήμη του και συνεχίζουμε μ'άλλο είδος ζώου. Αν ο

υπολογιστής βρει το είδος του ζώου που έχει βάλει ο παίχτης στο μυαλό του αφήνει ένα επιφωνημα χαράς (δηλαδή ένα χαρακτηριστικό ήχο) και προκαλεί τον παίχτη για να του θέσει μια πιο δύσκολη ερώτηση. Έτσι μετά από πολλές ερωτήσεις ο υπολογιστής έχει στην μνήμη του αρκετά είδη ζώων που τον βοηθούν για ένα καλύτερο παιχνίδι. Αν θέλουμε όμως μπορούμε να τα σβήσουμε όλα και να ξαναρχίσουμε απ'την αρχή παιδεύοντας τον για να βρει το ζώο που έχουμε βάλει στο μυαλό μας. Οι ελάχιστες απαιτήσεις του παιχνιδιού είναι 16K RAM και ένα σύστημα δισκέτας του Apple IIe.

```

]PR#0
>LIST
100 CUR=1: REM HERE IS WHERE PROGRAM IS PLAYED
110 PRINT OP#
120 PRINT RF#;CUR
130 INPUT NEWS: IF NEWS(1,1)="" THEN 500: INPUT RTPTR: INPUT WRNGPTR
140 REM NOW HAVE QUESTION TO ASK. RTPTR POINTS AT RECORD TO GO TO IF HE ANSWERS YES, WRNGPTR IF HE SAYS NO.
150 PRINT CL#
160 PRINT : PRINT NEWS: INPUT "?",A#:PREV=CUR: IF NOT LEN(A#) THEN 160: A#=A$(1,1): IF A##"Y" AND A##"N" THEN 160
170 IF A#="" THEN CUR=RTPTR: IF A#="" THEN CUR=WRNGPTR: GOTO 110
500 REM HAVE REACHED AN ANIMAL ENTRY. IS IT THE RIGHT ONE?
510 PRINT CL#;ANS="AEIOU": PRINT "IS THE ANIMAL YOU'RE THINKING OF A": FOR U#L#1 TO 5
520 IF AN$(U#L,U#L)=NEWS(2,2) THEN PRINT "N": NEXT U#: PRINT : PRINT NEWS(2): INPUT "?",ANS: IF NOT LEN(ANS) THEN 510
530 ANS=ANS(1,1): IF ANS="" AND ANS="" THEN 510: IF ANS="" THEN 800
540 REM I DON'T KNOW WHAT ANIMAL IT IS, ASK HIM
550 PRINT : PRINT "ALL RIGHT, I GIVE UP. WHAT ANIMAL WERE": PRINT "YOU THINKING OF?"
560 ANIM#="": INPUT ANIM$(2)
570 PRINT : PRINT "PLEASE TYPE A QUESTION THAT WOULD": PRINT "DISTINGUISH BETWEEN A":NEWS(2): PRINT "AND A":ANIM$(2)
580 INPUT ":",Q#: IF Q#( LEN(Q#))="" THEN Q#=Q$(1, LEN(Q#)-1)
590 PRINT : PRINT "WHAT WOULD THE CORRECT ANSWER FOR A": PRINT ANIM$(2): " BE?": INPUT ANS;ANS=ANS(1,1): IF ANS="" AND ANS="" THEN 590
600 PRINT OP#
610 PRINT RF#;0
620 REM UPDATE HEADER RECORD.
630 INPUT CNT,LAST: PRINT WF#;0
640 CNT=CNT+1:LAST=LAST+2: PRINT CNT:",";LAST: PRINT RF#;PREV
650 INPUT NEWS: INPUT OLD#; INPUT OLD#; PRINT WF#;PREV
660 PRINT NEWS: IF A#="" THEN 670: PRINT LAST-1: PRINT OLD#: GOTO 680
670 PRINT OLD#: PRINT LAST-1
680 PRINT WF#;LAST-1
820 PRINT "I GOT IT RIGHT!!!"
700 PRINT LAST: PRINT CUR
710 PRINT WF#;LAST
720 PRINT ANIM#: PRINT CL#
730 PRINT "I NOW KNOW":CNT;" ANIMALS!": GOTO 830
800 REM I GOT IT RIGHT!!!
810 FOR L#1 TO 10: PRINT : NEXT L
820 PRINT "I GOT IT RIGHT!!!"
930 PRINT : INPUT "WANT TO PLAY AGAIN? ",A#: IF LEN(A#) THEN IF A$(1,1)="" THEN END : GOTO 100
1020 PRINT "NORM I,O,C"
1030 TEXT : CALL -936: DIM NEWS(70),ANIM$(70),Q$(70),A$(70),RF$(20),WF$(20),ANS(10),CL$(20),OP$(40)
1040 RF#="READ ANIMALSFILE,R":WF#="WRITE ANIMALSFILE,R":CL#="CLOSE"OP#="OPEN ANIMALSFILE,LB0": PRINT OP#
1041 REM THE ABOVE STRINGS WERE SET SO THAT WE ONLY NEED TO PRINT A STRING RATHER THAN TYPE THE WHOLE OPERATION. EXAMPLE:
1042 REM RATHER THAN PRINT D#;READ ANIMALSFILE,R WE PRINT RF#. NOTE THAT THE "R" IS FOR RANDOM ACCESS INTO FILES.
1043 PRINT "WARNING: YOUR DISKETTE MAY NOT BE": PRINT "WRITE PROTECTED IN ORDER TO PLAY THIS": PRINT "GAME!"
1050 UTAB 5: PRINT " **** ANIMAL LEARNING GAME ****": PRINT : PRINT "HAVE YOU EVER PLAYED BEFORE?": INPUT A#: IF LEN(A#) THEN IF A$(1,1)="" THEN 100
1060 PRINT : PRINT "INSTRUCTIONS:": PRINT : PRINT "YOU WILL THINK OF AN ANIMAL AND I WILL": PRINT "TRY TO GUESS WHAT ANIMAL YOU ARE"
1070 PRINT "THINKING OF. I WILL DO THIS BY ASKING: PRINT "YOU A SERIES OF QUESTIONS ABOUT YOUR"
1080 PRINT "ANIMAL, TO WHICH YOU RESPOND EITHER": PRINT "YES OR NO. (ACTUALLY, A SIMPLE 'Y' OR": PRINT "'N' RESPONSE IS SUFFICIENT)"
1090 PRINT "IF I DO NOT GUESS WHAT YOUR ANIMAL IS,": PRINT "I WILL ASK YOU A FEW QUESTIONS SO THAT"
1100 PRINT "NEXT TIME I WILL KNOW WHAT YOUR ANIMAL": PRINT "IS.": PRINT
1110 PRINT "DO YOU WANT ME TO START LEARNING, OR": PRINT "HAVE YOU TAUGHT ME ANIMALS BEFORE?"
1111 PRINT "TYPE 'NEWFILE' FOR ME TO FORGET WHAT I": PRINT "HAVE LEARNED UP TO NOW OR IF NO ONE HAS"
1112 INPUT "TAUGHT ME ANYTHING YET: ",A#: IF A##"NEWFILE" THEN 100
1120 PRINT WF#;0
1130 PRINT "2,4"
1140 PRINT WF#;1
1150 PRINT "DOES IT LIVE IN THE WATER": PRINT 2: PRINT 3
1160 PRINT WF#;2
1170 PRINT "FROG"
1180 PRINT WF#;3
1190 PRINT "MOOSE"
1200 GOTO 100

```

# ADDRESS

## (ΔΙΕΥΘΥΝΣΙΟΓΡΑΦΟΣ)

Το πρόγραμμα ADDRESS είναι ένα από το σύνολο των προγραμμάτων της δισκέτας που συνοδεύει το εγχειρίδιο της Applesoft Basic για τον Iie. Πρόκειται για ένα κλασι-

κό πρόγραμμα Διευθυνσιογράφου που μας επιτρέπει την καταχώρηση όλων των απαραίτητων διευθύνσεων μας σ' ένα αρχείο και την εμφάνιση πάνω στην οθόνη οποιασδήποτε διεύθυνσης επιθυμούμε σ' άμεσο χρόνο.

Με το τρέξιμο του προγράμματος εμφανίζεται ένα βασικό menu πάνω στην οθόνη που περιλαμβάνει μια ερώτηση για το τι ακριβώς θέλουμε να κάνουμε που συνοδεύεται από τις δυνατές επιλογές του προγράμματος, για παράδειγμα, να εισάγουμε μια νέα διεύθυνση ήδη καταχωρημένη διεύθυνση ή τέλος αφού έχουμε εκτελέσει τις επιθυμητές εργασίες να τερματίσουμε το πρόγραμμα. Ένα μήνυμα πάνω στην οθόνη: TYPE A NUMBER ADDRESS (RETURN) μας καλεί να τυπώσουμε τον αριθμό της επιλογής μας. Έτσι αν διαλέξουμε την πρώτη εκλογή έχουμε την δυνατότητα να καταχωρήσουμε μια νέα διεύθυνση. Εμφανίζονται στην οθόνη οι λέξεις:

NAME (όνομα)  
ADDRESS (διεύθυνση)  
CITY (πόλη)  
STATE (χώρα)  
ZIP CODE (T.T)  
PHONE (τηλέφωνο)

Αφού συμπληρώσουμε τ' ανάλογα στοιχεία μπορούμε να χρησιμοποιήσουμε τη δεύτερη επιλογή για να ελέγξουμε την οποιαδήποτε καταχώρησή μας ή να συνεχίσουμε την καταχώρηση διευθύνσεων πατώντας το πλήκτρο επιστροφής ή να διαλέξουμε την τρίτη επιλογή και να τερματίσουμε την εκτέλεση του προγράμματος. Το πρόγραμμα είναι κατάλληλο για κάθε γραφείο ή επιχείρηση και υπάρχει η δυνατότητα τροποποίησης του ώστε να κρατά λεπτομερέστερα στοιχεία.

```

JLIST
10 D$ = CHR$(4)
20 F$ = "BLACK.BOOK"
30 HOME
40 PRINT "WHAT WOULD YOU LIKE TO DO? "
50 PRINT " 1 ENTER A NEW ADDRESS"
60 PRINT " 2 LOOK UP AN ADDRESS"
70 PRINT " 3 END"
80 INPUT "TYPE A NUMBER AND PRESS RETURN ";C$
90 C = VAL(C$)
100 IF C < 1 OR C > 3 THEN GOTO 30
110 IF C = 1 THEN GOSUB 1000
120 IF C = 2 THEN GOSUB 2000
130 IF C = 3 THEN END
140 GOTO 30
1000 REM
1010 INPUT "NAME: ";N$
1020 INPUT "ADDRESS: ";A$
1030 INPUT "CITY: ";C$
1040 INPUT "STATE: ";S$
1050 INPUT "ZIP CODE: ";Z$
1060 INPUT "PHONE: ";P$
1070 PRINT D$;"OPEN ";F$;"",L200"
1080 PRINT D$;"READ ";F$;"",R0"
1090 INPUT TR:TR = TR + 1
1100 PRINT D$;"WRITE";F$;"",R";TR
1110 PRINT N$: PRINT A$: PRINT C$
1120 PRINT S$: PRINT Z$: PRINT P$
1130 PRINT D$;"WRITE ";F$;"",R0"
1140 PRINT TR
1150 PRINT D$;"CLOSE ";F$
1160 RETURN
2000 PRINT D$;"OPEN ";F$;"",L200"
2010 PRINT D$;"READ ";F$;"",R0"
2020 INPUT TR
2030 IF TR = 0 THEN GOTO 2210
2040 HOME
2050 PRINT "WHOSE ADDRESS DO YOU WANT? "
2060 FOR I = 1 TO TR
2070 PRINT D$;"READ ";F$;"",R";I
2080 INPUT N$
2090 PRINT I,N$
2100 NEXT I
2110 PRINT D$
2120 INPUT "TYPE A NUMBER AND PRESS RETURN ";R$
2130 R = VAL(R$)
2140 IF R < 1 OR R > TR THEN GOTO 2120
2150 HOME
2160 PRINT D$;"READ ";F$;"",R";R
2170 INPUT N$,A$,C$,S$,Z$,P$
2180 PRINT N$: PRINT A$
2190 PRINT C$,S$
2200 PRINT Z$: PRINT P$
2210 PRINT D$;"CLOSE ";F$
2220 VTAB 23: HTAB 10
2230 INPUT "PRESS RETURN TO CONTINUE";T$
2240 RETURN

```



# MUSIC SYNTHESIZER

Πρόκειται για ένα ευχάριστο, διασκεδαστικό πρόγραμμα γραμμένο σε UCSD-PASCAL που σας προσκαλεί να δοκιμάσετε τις συνθετικές σας ικανότητες, γράφοντας μια δική σας σύνθεση, και στη συνέχεια να διευθύνετε την "ορχήστρα" του APPLE IIe. Για να γράψετε μια επιτυχή σύνθεση θα πρέπει προσεκτικά να ορίσετε το μέτρο που θα χρησιμοποιήσετε, τον αριθμό των διαστημάτων, τις νότες κάθε διαστήματος και το τέμπο που προτιμάται.

Η σύνθεση της μουσικής γίνεται με την βοήθεια του προγράμματος SCORE ενώ το πρόγραμμα αναλαμβάνει την ακουστική απόλαυση της σύνθεσης σας.

Όταν εκτελείται το πρώτο πρόγραμμα εμφανίζεται αρχικά στην οθόνη το μήνυμα:

ENTER A SONG:

Με το πρώτο χτύπημα του πλήκτρου επιστρο-

φής εμφανίζεται το μήνυμα:

ENTER NUMBER OF BARS

που σας καλεί να καθορίσετε τον αριθμό των μουσικών διαστημάτων που θα χρησιμοποιήσετε. Ο μέγιστος επιτρεπτός αριθμός διαστημάτων περιορίζεται στον αριθμό 20 από τη σταθερά MAXB του προγράμματος. Στη συνέχεια εμφανίζεται το μήνυμα ENTER TEMPO οπότε διαλέγετε το τέμπο που θα χρησιμοποιήσετε από ένα φάσμα 255 διαφορετικών ρυθμών. Έτσι μπορούμε να συνθέσουμε από ένα μοντέρνο κομμάτι σε ξέφρενο ρυθμό μέχρι μια σιγανή αλλά επιβλητική κλασική σύνθεση.

Στη συνέχεια εμφανίζεται το μήνυμα

ENTER NUMBER OF NOTES IN THIS MEASURE

οπότε καθορίζετε πόσες νότες θα περιλάβετε στο κάθε διάστημα.

Ανάλογα με τον αριθμό από νότες που θα ορί-

σετε θα εμφανιστούν στην οθόνη αντίστοιχα μηνύματα:

ENTER NOTE  
ENTER KIND OF NOTE

στα οποία δίπλα συμπληρώνουμε τις νότες της μουσικής μας και το είδος της κάθε νότας αν δηλαδή πρόκειται για 1/2 (HALF), 1/8 (EIGHTH), 1/4 (QTR) κλπ. Αφού "γεμίσετε" όλα τα διαστήματα με τις νότες σας το μόνο που απομένει είναι να "τρέξετε" το δεύτερο πρόγραμμα PLAY και να υψώσετε τα χέρια σας για να είστε έτοιμος να διευθύνετε την πρώτη σας μουσική δημιουργία που εκτελεί η ορχήστρα του Apple IIe. Δίνουμε παρακάτω μερικά από τα δεδομένα που χρησιμοποίησε ο συνεργάτης μας κατά το TEST RUN του προγράμματος MUSIC SYNTHESIZER.

## ΔΕΔΟΜΕΝΑ

ENTER A SONG:  
ENTER NUMBER OF BARS:4  
ENTER TEMPO:125

ENTER NUMBER OF NOTES IN THIS MEASURE:3  
ENTER NOTE:SD1  
ENTER KIND OF NOTE: HALF  
ENTER NOTE:SD1  
ENTER KIND OF NOTE: HALF  
ENTER NOTE:SD1  
ENTER KIND OF NOTE: WHOLE  
ENTER NUMBER OF NOTES IN THIS MEASURE:1  
ENTER NOTE:SF2  
ENTER KIND OF NOTE: WHOLE  
ENTER NUMBER OF NOTES IN THIS MEASURE:2  
ENTER NOTE:SF2  
ENTER KIND OF NOTE: HALF  
ENTER NOTE:NA1  
ENTER KIND OF NOTE: THIRD  
ENTER NUMBER OF NOTES IN THIS MEASURE:1  
ENTER NOTE:ND1  
ENTER KIND OF NOTE: WHOLE

Το αποτέλεσμα αυτών των δεδομένων ήταν ένας αργός, βαρύς, αλλά εντυπωσιακός σκοπός.

## LISTEN TO THE MUSIC

```

program PLAY ;

uses APPLESTUFF ;

const
  MAXN = 16 ;
  MAXB = 20 ;

type
  MEASURES = record
    N : 1 .. MAXN ;
    BLASTS : array[ 1..MAXN ] of 0..50 ;
    LEN : array[ 1..MAXN ] of 0..255 ;
  end ;
  
```

```

SONGS      = record
    TEMPO  : 0 .. 255 ;
    BARS   : 1 .. MAXB ;
    SCORE  : array[ 1..MAXB ] of MEASURES ;
end ;

var
B, M      : 1 .. MAXB ;
T         : 0 .. 255 ;
J         : 1 .. MAXN ;

DISK      : file of SONGS ;
COUNT    : integer ;

begin
    RESET ( DISK, 'SONGS.DATA' ) ;
    T := DISK^.TEMPO ;
    B := DISK^.BARS ;

    for M := 1 to B do
        with DISK^.SCORE[ M ] do
            for J := 1 to N do
                begin
                    COUNT := LEN[ J ] ;
                    case COUNT of
                        1 : COUNT := T ;
                        2 : COUNT := T div 2 ;
                        3 : COUNT := T div 3 ;
                        4 : COUNT := T div 4 ;
                        5 : COUNT := T div 8 ;
                        6 : COUNT := ( LEN[ J-1 ] div 2 ) * T ;
                    end ;
                    NOTE( BLASTS[ J ], COUNT ) ;
                end ;
            end ;
        end ;
    end ;

    CLOSE( DISK ) ;
end.

```

Figure 9.6. PLAY

#### MAKING MUSIC

```

WRITELN ( 'ERROR IN INPUT !!' ) ;
until FALSE ;
end ; (* SOUNDS *)

function KINDS : integer ;

var
S : string[ 5 ] ;
I : BEATS ;
begin
    repeat
        WRITE ( 'ENTER KIND OF NOTE : ' ) ;
        READLN ( S ) ;
        for I := ZERO to DOT do
            begin
                KINDS := ORD( I ) ;
                if S = BT[ I ] then exit( KINDS ) ;
            end ;
        until FALSE ;
        WRITELN ( 'ERROR IN INPUT !!' ) ;
    until FALSE ;
end ; (* KINDS *)

begin
NT [ RST ] := 'RST' ;    BT [ ZERO ] := 'ZERO' ;
NT [ NF1 ] := 'NF1' ;   BT [ WHOLE ] := 'WHOLE' ;
NT [ SF1 ] := 'SF1' ;   BT [ HALF ] := 'HALF' ;
NT [ NG1 ] := 'NG1' ;   BT [ THIRD ] := 'THIRD' ;
NT [ SG1 ] := 'SG1' ;   BT [ QTR ] := 'QTR' ;
NT [ NA1 ] := 'NA1' ;   BT [ EIGHT ] := 'EIGHT' ;
NT [ SA1 ] := 'SA1' ;   BT [ DOT ] := 'DOT' ;
NT [ NB1 ] := 'NB1' ;
NT [ NC1 ] := 'NC1' ;
NT [ SC1 ] := 'SC1' ;
NT [ ND1 ] := 'ND1' ;
NT [ SD1 ] := 'SD1' ;
NT [ NE1 ] := 'NE1' ;
NT [ NF2 ] := 'NF2' ;
NT [ SF2 ] := 'SF2' ;
NT [ NG2 ] := 'NG2' ;
NT [ SC2 ] := 'SC2' ;

```

```

WRITELN ( 'ENTER A SONG : ' ) ;
WRITE ( 'ENTER NUMBER OF BARS = ' ) ; READLN ( B ) ;
WRITE ( 'ENTER TEMPO ( 1..255 ) = ' ) ; READLN ( T ) ;

REWRITE( DISK, 'SONGS.DATA' ) ;
DISK^.TEMPO := T ;
DISK^.BARS := B ;

for M := 1 to B do
    with DISK^.SCORE[ M ] do
        begin
            N := HOWMANY ;
            for J := 1 to N do
                begin
                    BLASTS[ J ] := SOUNDS ;
                    LEN[ J ] := KINDS ;
                end ;
            end ; (* with *)
        end ;
    end ;

PUT( DISK ) ;
CLOSE( DISK, LOCK ) ;
end.

```

Figure 9.5. (Continued)

```

program SCORE ;

uses APPLESTUFF ;

const
MAXN = 16 ;
MAXB = 20 ;

type
NOTES = ( RST, NF1, SF1, NG1, SG1, NA1, SA1,
          NB1, NC1, SC1, ND1, SD1, NE1, NF2,
          SF2, NG2, SG2 ) ;

BEATS = ( ZERO, WHOLE, HALF, THIRD, QTR, EIGHT, DOT )

MEASURES = record
    N : 1 .. MAXN ;
    BLASTS : array[ 1..MAXN ] of 0..50 ;
    LEN : array[ 1..MAXN ] of 0..255 ;
end ;

SONGS = record
    TEMPO : 0 .. 255 ;
    BARS : 1 .. MAXB ;
    SCORE : array[ 1..MAXB ] of MEASURES ;
end ;

var
B, M : 1 .. MAXB ;
T : 0 .. 255 ;
J : 1 .. MAXN ;
DISK : file of SONGS ;
NT : array[ NOTES ] of string[ 3 ] ;
BT : array[ BEATS ] of string[ 5 ] ;

function HOWMANY : integer ;

var
N : 1 .. MAXN ;
begin
    WRITE ( 'enter number of notes in this measure = ' ) ;
    READLN ( N ) ;
    HOWMANY := N ;
end ; (* HOWMANY *)

function SOUNDS : integer ;

var
S : string[ 3 ] ;
I : NOTES ;
begin
    repeat
        WRITE ( 'ENTER NOTE : ' ) ;
        READLN ( S ) ;
        for I := RST to SG2 do
            begin
                SOUNDS := ORD( I ) ;
                if S = NT[ I ] then exit( SOUNDS ) ;
            end ;
        end ;
    until FALSE ;
end ;

```

Figure 9.5. SCORE

Προγράμματα για τόν:

# OSBORNE 1

ΕΠΙΜΕΛΕΙΑ:

Παναγιώτης Γιωτόπουλος



# ΔΙΑΧΕΙΡΗΣΗ ΣΕΙΡΙΑΚΩΝ ΑΡΧΕΙΩΝ ΣΤΗΝ MBASIC

Η δημιουργία αρχείων και η διαχείριση των στοιχείων που περιέχουν, δηλαδή η δυνατότητα προσθήκης νέων στοιχείων, η δυνατότητα αλλαγής των υπάρχοντων στοιχείων και η διαγραφή τους, είναι βασικές προϋποθέσεις για την ουσιαστική αξιοποίηση των προσωπικών MICROS. Το πρόγραμμα που σας παρουσιάζουμε δημιουργεί και διαχειρίζεται αρχεία σε μορφή SEQUENTIAL, με δυνατότητα 4 ή 7 πεδίων για κάθε RECORD. Προσθέτοντας, όμως, τις κατάλληλες υπορουτίνες στο τέλος του, μπορούμε να αρχειοθετήσουμε στοιχεία-RECORDS με οποιοδήποτε επιθυμητό αριθμό πεδίων.

Ας αναφέρουμε μερικές πολύ συνηθισμένες περιπτώσεις, όπου χρειάζονται αρχεία. Μπορούμε να διαφυλάξουμε εκεί πληροφορίες, όπως π.χ. το σύνολο των προσωπικών μας φίλων ή συγγενών, τα μέλη ενός συλλόγου, το σύνολο των πελατών της επιχείρησής μας, ή ακόμα, το σύνολο των προϊόντων που παράγουμε ή εμπορευόμαστε.

Για να ακολουθήσουμε από κοντά πως δουλεύει το πρόγραμμα, ας φανταστούμε τη δημιουργία ενός αρχείου πελατών, με 7 πληροφοριακά στοιχεία-πεδία για κάθε πελάτη. Μπο-

ρούσαν νάναι και μόνο 4, όπως ήδη είπαμε.

Έτσι για κάθε πελάτη-RECORD ενημερώνουμε το αρχείο με τα εξής πεδία.

- 1-Τον κωδικό του αριθμό
- 2-Το όνομα του πελάτη
- 3-Τον κωδικό του κλάδου ή τομέα δραστηριότητας στον οποίο ανήκει ο πελάτης
- 4-Την διεύθυνση, δηλαδή οδό και αριθμό
- 5-Την περιοχή, δηλαδή συνοικία, πόλη
- 6-Το τηλέφωνό του
- 7-Διάφορες χρήσιμες πληροφορίες

Όταν δουλεύουμε με αρχεία, είναι ανάγκη να τηρούμε σε ιδιαίτερη κατάσταση τις ονομασίες τους, όπως τις αναγνωρίζει ο υπολογιστής μας, τη δομή τους - πόσα πεδία περιέχει κάθε RECORD, κλπ. Επίσης, κατά την διαχείριση των αρχείων χρειάζεται, ενδιάμεσα, να δημιουργήσουμε μερικά προσωρινά αρχεία-TEMPFIL, στα οποία γίνονται οι επιθυμητές αλλαγές, προσθήκες κλπ.

Αυτά θυμίζουμε στο χειριστή με τις εντολές 4 και 6 του προγράμματος. Στη συνέχεια, το πρόγραμμα ζητά να μάθει το όνομα του αρχείου, και το πλήθος των πεδίων ανά RECORD-εδώ 7-.

Με τον πίνακα-ARRAY DIM D\$(Z), που ακολουθεί ορίζουμε

τους παραπάνω χαρακτηρισμούς-ονόματα των 7 πεδίων. Αυτά τα στοιχεία θα μας ζητάει το πρόγραμμα να συμπληρώσουμε για κάθε RECORD. Το MENU επιλογής, στη συνέχεια, μας καλεί να ορίσουμε τι θέλουμε να κάνουμε με το αρχείο. Να το δημιουργήσουν, να προσθέσουμε σ'αυτό νέα RECORDS, ν'αλλάξουμε τα ήδη υπάρχοντα, να διαγράψουμε στοιχεία, να τ'ανακαλέσουμε στην οθόνη, ή, τέλος, να τελειώσουμε μ'αυτή την ..ιστορία!. Κατά τη δημιουργία του αρχείου μας, που προηγείται βέβαια, το ανοίγουμε για να γράψουμε σ'αυτό τα στοιχεία, που θέλουμε. Με το STOP τελειώνουν οι εγγραφές και κλείνει το αρχείο. Για τους υπόλοιπους χειρισμούς-προσθήκης, αλλαγής διαγραφής στοιχείων, μεσολαβεί μια αντιγραφή των στοιχείων που κάθε φορά επιθυμούμε εκ του βασικού αρχείου μας, σε ένα προσωρινό αρχείο-TEMPFIL, που δημιουργούμε μέσω ενός διαφορετικού BUFFER, π.χ. 2.Επακολουθεί στο πρόγραμμα η διαγραφή του πρώτου αρχείου και η μετονομασία του TEMPFIL με το όνομα του αρχικού μας αρχείου, που έχει ήδη υποστεί την επεξεργασία που θέλαμε.

Πάντοτε, μετά την κάθε επιλογή, γυρίζου-

με στο αρχικό MENU για συνέχεια. Τα αρχεία, μετά την κάθε φάση επεξεργασίας τους, πρέπει να κλείνουν με την εντολή CLOSE, πράγμα που βεβαιώνουμε με σχετική αναφορά μας στο πρόγραμμα.

Εγγραφές στα αρχεία ή διαβάσματα απ'αυτά κάνουμε με τις κατάλληλες εντολές της MBASIC, χρησιμοποιώντας τις τελευταίες υπορουτίνες και δουλεύοντας, στη προκειμένη περίπτωση, μ'αυτή που αναφέρεται στα 7 πεδία. Με την ανάκληση (6), επιβεβαιώνουμε κάθε φορά ποια και πόσα RECORDS, έχουμε στο αρχείο. Ας μη ξεχνάμε ότι, τα αρχεία που δημιουργούμε δεν είναι αυτοσκοπός. Είναι μέσα - πρώτη ύλη θα λέγαμε - που αξιοποιούμε στη συνέχεια με πλήθος άλλων προγραμμάτων, για να πάρουμε χρήσιμες στην εργασία μας πληροφορίες.

Άλλα προγράμματα που θα "τρέξουμε" στον OSBORNE 1 προϋποθέτουν την ύπαρξη αρχείων. Απλά, επεξεργάζονται τα στοιχεία με τα οποία τα έχουμε ενημερώσει, προς όφελός μας. Για οποιαδήποτε πρόσθετη πληροφορία σχετική με το πρόγραμμα απευθυνθείτε στον κ. Παν. Γιωτόπουλο, τηλ. 5245-811 - πρώτες ώρες.

```

1 REM ΓΙΟΤΟΠΟΥΛΟΣ ΠΑΝΑΓΙΩΤΗΣ
2 REM ΣΕΠΤΕΜΒΡΙΟΣ 1983
3 PRINT CHR$(26)
4 PRINT " ΣΥΜΒΟΥΛΕΥΣΙΣ ΠΙΝΑΚΑ ΜΕ ΤΙΣ ΟΝΟΜΑΣΙΕΣ"
6 PRINT " ΚΑΙ ΤΗΝ ΔΟΜΗ ΤΩΝ ΑΡΧΕΙΩΝ ΣΑΣ"
8 LINE INPUT "ΔΩΣΕ RETURN ΓΙΑ ΣΥΝΕΧΕΙΑ";U$
10 PRINT CHR$(26)
20 PRINT "ΤΟ ΟΝΟΜΑ ΚΑΘΕ ΑΡΧΕΙΟΥ ΚΑΘΟΡΙΖΕΤΑΙ ΑΠΟ ΤΟ ΠΡΟΓΡΑΜΜΑ"
30 PRINT TAB(5);"ΔΩΣΕ ΟΝΟΜΑ ΑΡΧΕΙΟΥ ΣΕ ΜΟΡΦΗ Β;ΟΝΟΜΑ"
35 INPUT F$
36 PRINT "-----"
80 PRINT TAB(10);"ΔΩΣΕ ΠΛΗΘΟΣ ΠΕΔΙΩΝ ΚΑΘΕ RECORD 4 Η 7 "
90 INPUT Z
110 DIM D$(Z)
120 FOR I=1 TO Z
126 PRINT "-----"
130 PRINT "ΟΝΟΜΑ ΠΕΔΙΟΥ";" ";I
140 INPUT D$(I)
150 NEXT I
160 PRINT CHR$(26)
170 REM ΕΠΕΞΕΡΓΑΣΙΑ ΑΡΧΕΙΩΝ
175 JOO=0
180 PRINT TAB(5);"ΕΠΙΛΟΓΗ ΕΙΔΟΥΣ ΕΠΕΞΕΡΓΑΣΙΑΣ ΤΟΥ ΑΡΧΕΙΟΥ";" ";F$
190 PRINT "-----"
200 PRINT " 1-ΓΙΑ ΔΗΜΙΟΥΡΓΙΑ ΑΡΧΕΙΟΥ"
210 PRINT " 2-ΓΙΑ ΠΡΟΣΘΗΚΕΣ ΝΕΩΝ ΣΤΟΙΧΕΙΩΝ"
220 PRINT " 3-ΓΙΑ ΑΛΛΑΓΕΣ ΣΤΟΙΧΕΙΩΝ"
230 PRINT " 4-ΓΙΑ ΔΙΑΓΡΑΦΗ ΣΤΟΙΧΕΙΩΝ"
240 PRINT " 5-ΓΙΑ ΤΕΛΟΣ ΕΠΕΞΕΡΓΑΣΙΑΣ ΤΟΥ ΑΡΧΕΙΟΥ"
245 PRINT " 6-ΑΝΑΚΛΗΣΗ ΣΤΟΙΧΕΙΩΝ"
250 PRINT "-----"
260 PRINT "ΔΩΣΕ ΑΡΙΘΜΟ ΕΠΙΛΟΓΗΣ" : INPUT X
262 IF X<=0 OR X>=7 THEN 160
270 ON X GOTO 290,450,740,1070,1300,1380
280 REM ΔΗΜΙΟΥΡΓΙΑ ΑΡΧΕΙΟΥ
290 PRINT CHR$(26)
300 PRINT "ΔΗΜΙΟΥΡΓΙΑ ΑΡΧΕΙΟΥ";" ";F$
310 DIM W$(Z)
320 DIM S$(20)
330 OPEN "O",1,F$
340 FOR I=1 TO Z
350 PRINT "ΔΩΣΕ";" ";D$(I)
356 PRINT "-----"
360 INPUT W$(I)
370 LET S$(I)=W$(I)
380 NEXT I
390 GOSUB 2010
400 LINE INPUT "ΑΝ ΤΕΛΕΙΩΣΕΣ ΔΩΣΕ-STOP-ΑΛΛΟΙΩΣ RETURN";A$
410 IF A$="STOP" THEN 430
420 GOTO 340
430 CLOSE 1
435 PRINT "ΕΚΛΕΙΣΕ ΤΟ ΑΡΧΕΙΟ" : FOR U=1 TO 500:NEXT U
440 GOTO 160
446 REM ΠΡΟΣΘΗΚΕΣ ΝΕΩΝ RECORDS
450 PRINT CHR$(26)
460 OPEN "I",1,F$
470 OPEN "O",2,"B;ΤΕΜΠΦΙΑ"
480 IF EOF(1) THEN 530
490 GOSUB 3000

```

```

500 GOSUB 4000
510 GOTO 480
530 PRINT TAB(5);"ΑΡΧΙΣΕ ΤΙΣ ΠΡΟΣΘΗΚΕΣ ΤΩΝ ΝΕΩΝ RECORDS"
535 PRINT TAB(5);"-----"
540 FOR I=1 TO Z
550 PRINT "ΔΩΣΕ";" ";D$(I)
560 PRINT "-----"
570 INPUT W$(I)
580 LET S$(I)=W$(I)
590 NEXT I
600 GOSUB 4000
610 LINE INPUT "ΓΙΑ ΤΕΛΟΣ ΠΡΟΣΘΗΚΩΝ ΔΩΣΕ-STOP";A$
620 IF A$="STOP" THEN 640
630 GOTO 540
640 CLOSE 1,2
650 KILL F$
660 NAME "B;ΤΕΜΠΦΙΑ" AS F$
730 GOTO 160
740 REM ΑΛΛΑΓΕΙΣ ΣΤΑ RECORDS
750 PRINT CHR$(26)
770 OPEN "I",1,F$
780 OPEN "O",2,"B;ΤΕΜΠΦΙΑ"
790 PRINT TAB(10);"ΠΡΟΣΟΧΗ-ΤΟΥΣ ΚΩΔΙΚΟΥΣ ΤΩΝ RECORDS ΓΙΑ ΑΛΛΑΓΗ"
800 PRINT TAB(10);"ΝΑ ΤΟΥΣ ΕΙΣΑΓΕΤΕ ΣΕ ΑΥΘΟΥΣΙΑ ΣΕΙΡΑ"
810 PRINT "ΔΩΣΕ ΚΩΔΙΚΟ RECORD ΓΙΑ ΑΛΛΑΓΗ":INPUT M$
830 IF EOF(1) THEN 970
840 GOSUB 3000
850 IF S$(1)=M$ THEN 880
860 GOSUB 4000
870 GOTO 830
880 PRINT TAB(5);"ΑΛΛΑΣΤΕ ΤΑ ΣΤΟΙΧΕΙΑ ΤΟΥ RECORD";" ";M$
890 FOR I=1 TO Z
900 PRINT "ΔΩΣΕ ΝΕΟ";" ";D$(I)
910 INPUT W$(I)
920 LET S$(I)=W$(I)
930 NEXT I
940 GOSUB 4000
950 LINE INPUT "ΘΕΛΕΙΣ ΑΛΛΗ ΑΛΛΑΓΗ-Υ Η Ν-";A$:IF LEFT$(A$,1)="Y" THEN 790
960 GOTO 830
970 CLOSE 1,2
980 KILL F$
990 NAME "B;ΤΕΜΠΦΙΑ" AS F$
1050 PRINT "ΕΓΙΝΑΝ ΑΛΛΑΓΕΙΣ ΚΑΙ ΕΚΛΕΙΣΑΝ ΤΑ ΑΡΧΕΙΑ":FOR E=1 TO 500:NEXT E
1060 GOTO 160
1070 REM ΔΙΑΓΡΑΦΗ RECORDS
1080 PRINT CHR$(26)
1100 OPEN "I",1,F$
1110 OPEN "O",2,"B;ΤΕΜΠΦΙΑ"
1120 LINE INPUT "ΔΩΣΕ ΚΩΔΙΚΟ RECORD ΠΡΟΣ ΔΙΑΓΡΑΦΗ";P$
1130 IF EOF(1) THEN 1200
1140 GOSUB 3000
1150 IF S$(1)=P$ THEN 1130
1160 GOSUB 4000
1170 GOTO 1130
1200 CLOSE 1,2
1210 KILL F$
1220 NAME "B;ΤΕΜΠΦΙΑ" AS F$
1280 PRINT "ΕΓΙΝΑΝ ΔΙΑΓΡΑΦΕΙΣ ΚΑΙ ΕΚΛΕΙΣΑΝ ΑΡΧΕΙΑ":FOR E=1 TO 500:NEXT E

```

```

1290 GOTO 160
1300 REM ΤΕΛΟΣ ΕΠΕΞΕΡΓΑΣΙΑΣ ΑΡΧΕΙΟΥ
1305 PRINT CHR$(26)
1350 PRINT "ΤΑ ΕΝΗΜΕΡΩΜΕΝΑ ΑΡΧΕΙΑ ΣΑΣ ΜΠΟΡΕΙΤΕ ΤΩΡΑ ΝΑ ΤΑ"
1360 PRINT "ΧΡΗΣΙΜΟΠΟΙΗΣΕΤΕ ΣΤΑ ΠΡΟΓΡΑΜΜΑΤΑ ΣΑΣ-ΚΑΛΗ ΔΟΥΛΕΙΑ"
1370 GOTO 5100
1380 PRINT CHR$(26)
1390 IF Z<>4 THEN 1540
1400 OPEN "I",1,F$
1410 IF EOF(1) THEN 1500
1420 J00=J00+1
1430 GOSUB 3000
1440 PRINT S$(1)
1450 PRINT S$(2)
1460 PRINT S$(3)
1470 PRINT S$(4)
1480 PRINT
1490 GOTO 1410
1500 PRINT "ΣΥΝΟΛΟ RECORD ΑΡΧΕΙΟΥ"; " ";F$;"=";J00
1510 CLOSE 1
1520 PRINT "ΔΩΣΕ RETURN ΓΙΑ ΣΥΝΕΧΕΙΑ":INPUT R$
1530 GOTO 160
1540 OPEN "I",1,F$
1550 IF EOF(1) THEN 1680
1560 J00=J00+1
1570 GOSUB 3000
1580 PRINT TAB(10);J00
1590 PRINT S$(1)
1600 PRINT S$(2)
1610 PRINT S$(3)
1620 PRINT S$(4)
1630 PRINT S$(5)
1640 PRINT S$(6)
1650 PRINT S$(7)
1670 GOTO 1550
1680 PRINT "ΣΥΝΟΛΙΚΟΣ ΑΡΙΘΜΟΣ RECORDS ΤΟΥ ΑΡΧΕΙΟΥ"; " ";F$;"=";J00
1685 CLOSE 1
1690 PRINT "ΔΩΣΕ RETURN ΓΙΑ ΣΥΝΕΧΕΙΑ":INPUT R$
1700 GOTO 160
2000 REM SUBROUTINE ΓΙΑ PRINT 1 Η WRITE 1
2010 IF Z<>4 THEN 2020
2012 PRINT 1,S$(1);",";S$(2);",";S$(3);",";S$(4)
2014 RETURN
2020 PRINT 1,S$(1);",";S$(2);",";S$(3);",";S$(4);",";S$(5);",";S$(6);",";S$(7)
2030 RETURN
3000 REM SUBROUTINE ΓΙΑ INPUT 1
3010 IF Z<>4 THEN 3040
3020 INPUT 1,S$(1),S$(2),S$(3),S$(4)
3030 RETURN
3040 INPUT 1,S$(1),S$(2),S$(3),S$(4),S$(5),S$(6),S$(7)
3050 RETURN
4000 REM SUBROUTINE ΓΙΑ PRINT 2
4010 IF Z<>4 THEN 4040
4020 PRINT 2,S$(1);",";S$(2);",";S$(3);",";S$(4)
4030 RETURN
4040 WRITE 2,S$(1),S$(2),S$(3),S$(4),S$(5),S$(6),S$(7)
4050 RETURN
5100 END

```

# ΔΥΟ ΠΡΟΓΡΑΜΜΑΤΑ ΑΞΙΟΠΟΙΗΣΗΣ ΤΩΝ ΣΤΟΙΧΕΙΩΝ ΤΟΥ FILESTAR 4/7

Τα δυο προγράμματα που παραθέτω στη συνέχεια αναφέρονται στην παρακολούθηση των πελατών μιας μικρής επιχείρησης και στην ταξινόμησή τους ανάλογα με τις ανάγκες της. Μπορούν να είναι χρήσιμα σε ένα τμήμα παραγγελιών ή στην διεύθυνση πωλήσεων, αντίστοιχα.

Ας αρχίσουμε από το CARDEX πελάτη. Το πρόγραμμα ζητάει να δώσουμε τον κωδικό του πελάτη. Αυτόν φροντίζουμε να του γνωρίζουμε πάντοτε από μια αλφαβητική ταξινόμηση των πελατών μας, που παίρνουμε με άλλο πρόγραμμα. Ανοίγουμε και "διαβάζουμε" το αρχείο που δημιουργήσαμε προηγουμένα, θεωρώντας, πάντοτε ότι είναι 7 τα πεδία κάθε RECORD πελάτη. Μετά την τακτο-

ποίηση του κωδικού του, εμφανίζεται στην οθόνη του OSBORNE 1 η καρτέλλα του πελάτη που ζητήσαμε, με όλα τα στοιχεία του. Με αντίστοιχες ονομασίες των πεδίων και των εντολών μπορούμε εύκολα να δημιουργήσουμε CARDEX με στοιχεία προσώπων, προϊόντων κλπ.

Το πρόγραμμα FIND-IT λειτουργεί στην ίδια λογική, ταξινομώντας τους πελάτες επιχείρησής μας κατά κλάδο ή τομέα δραστηριότητας, κατά περιοχή, κατά κλάδο και περιοχή, ενώ τέλος δίνει Διευθυνσιογράφο πελατών κατά κλάδο. Οι επιλογές αυτές μπαίνουν σ'ένα αρχικό MENU, ενώ με την εντολή 140 το πρόγραμμα εκτελείται σε διάφορες φάσεις, δίνοντας στην

οθόνη τα στοιχεία που ζητάμε. Είναι φανερό η χρησιμότητα του προγράμματος για οποιαδήποτε διεύθυνση πωλήσεων. Η δραστηριότητα του τμήματος είναι κατανεμημένη συνήθως σε διάφορους κλάδους ή τομείς της αγοράς. Ο κάθε πωλητής πρέπει να γνωρίζει το σύνολο των πελατών στους οποίους απευθύνεται.

Κάποια στιγμή προγραμματίζονται ταξίδια για προσωπικές επαφές με την αγορά. Η διεύθυνση πωλήσεων, αλλά και ο ίδιος ο πωλητής πρέπει να ξέρει τους πελάτες τους οποίους θα επισκεφθεί στη περιοχή ή ακόμα τους πελάτες της περιοχής που ανήκουν στον κλάδο ή τομέα που παρακολουθεί. (Επιλογές του MENU 2 και 3.)

Ας σημειωθεί εδώ ότι, βολικά και φορητά συστήματα όπως ο OSBORNE 1 αποδεικνύονται ιδιαίτερα χρήσιμα. Όλα βρίσκονται στους δίσκους\* και το βαλιτσάκι! Η επικοινωνία όμως με τους πελάτες δεν είναι μόνο προσωπική (επισκευές). Επιβάλλεται, κατά καιρούς, να είναι και γραπτή. Σ' αυτό βοηθά ο διευθυνσιογράφος που ετοιμάζει ετικέτες διευθύνσεων για την αποστολή εγκυκλίων, εντύπων, κλπ. στους πελάτες ενός ορισμένου κλάδου ή τομέα.

Και όπως γνωρίζουμε, με τους MICROS δεν ετοιμάζουμε μόνο ετικέτες, αλλά και τα ίδια τα κείμενα χρησιμοποιώντας έτοιμα προγράμματα επεξεργασίας κειμένων (WORDSTAR).

```
5 REM ΓΙΟΤΟΠΟΥΛΟΣ ΠΑΝΑΓΙΩΤΗΣ
8 REM ΠΡΟΓΡΑΜΜΑ CARDEX
10 PRINT CHR$(26)
20 PRINT "ΔΟΣΕ ΚΩΔΙΚΟ ΟΝΟΜΑΤΟΣ ΠΕΛΑΤΗ"
30 INPUT N$
40 OPEN "I",1,"B;ΠΕΛΑΤΕΣ"
50 IF EOF(1) THEN 320
```



```

60 INPUT 1,A$,B$,C$,D$,E$,F$,G$
70 IF A$=N$ THEN 110
80 GOTO 50
90 REM DISPLAY CARDEX
110 PRINT CHR$(26)
120 PRINT "-----"
130 PRINT "      ΚΑΡΤΕΛΛΑ      ΠΕΛΑΤΗ      "
140 PRINT "-----"
150 PRINT "ΚΩΔΙΚΟΣ"TAB(26);A$
160 PRINT
170 PRINT "ΠΕΛΑΤΗΣ";TAB(26);B$
180 PRINT
190 PRINT "ΚΩΔ. ΚΛΑΔΟΥ";TAB(26);C$
200 PRINT
210 PRINT "ΔΙΕΥΘΥΝΣΗ";TAB(26);D$
220 PRINT
230 PRINT "ΠΕΡΙΟΧΗ";TAB(26);E$
240 PRINT
250 PRINT "ΤΗΛΕΦΩΝΟ";TAB(26);F$
260 PRINT
270 PRINT "ΠΛΗΡΟΦΟΡΙΕΣ";TAB(15);G$
280 PRINT
290 PRINT "-----"
300 LINE INPUT "ΔΩΣΕ RETURN ΓΙΑ ΣΥΝΕΧΕΙΑ";R$
310 GOTO 50
320 CLOSE 1
330 PRINT "ΘΕΛΕΙΣ ΑΛΛΟ CARDEX ΠΕΛΑΤΗ-Υ Η Ν"
340 INPUT W$
350 IF LEFT$(W$,1)="Y" THEN 10
360 PRINT "ΤΕΛΟΣ ΕΡΓΑΣΙΑΣ"
370 END

```

```

1 REM ΓΙΟΤΟΠΟΥΛΟΣ ΠΑΝΑΓΙΟΤΗΣ
2 REM ΠΡΟΓΡΑΜΜΑ FIND IT
10 PRINT CHR$(26)
20 PRINT "ΤΟ ΠΡΟΓΡΑΜΜΑ ΑΥΤΟ ΠΑΡΟΥΣΙΑΖΕΙ ΤΟΥΣ ΠΕΛΑΤΕΣ"
30 PRINT "      ΚΑΤΑ ΚΑΤΗΓΟΡΙΕΣ"
40 PRINT "-----"
50 PRINT "      ΠΙΝΑΚΑΣ ΕΠΙΛΟΓΗΣ-MENU      "
60 PRINT "-----"
70 PRINT TAB(5);"1-ΠΕΛΑΤΕΣ ΚΑΤΑ ΚΛΑΔΟ ΔΡΑΣΤΗΡΙΟΤΗΤΑΣ"
80 PRINT TAB(5);"2-ΠΕΛΑΤΕΣ ΚΑΤΑ ΠΕΡΙΟΧΗ"

```

```

90 PRINT TAB(5);"3-ΠΕΛΑΤΕΣ ΚΑΤΑ ΚΛΑΔΟ ΚΑΙ ΠΕΡΙΟΧΗ"
100 PRINT TAB(5);"4-ΠΕΛΑΤΕΣ ΣΕ ΔΙΕΥΘΥΝΣΙΟΓΡΑΦΟ ΚΑΤΑ ΚΛΑΔΟ"
105 PRINT TAB(5);"5-ΤΕΛΟΣ ΕΡΓΑΣΙΑΣ"
110 PRINT "-----"
120 PRINT TAB(20);"ΔΩΣΕ ΑΡΙΘΜΟ ΕΠΙΛΟΓΗΣ"
130 INPUT N
140 ON N GOTO 150,310,460,620,840
150 PRINT CHR$(26)
160 PRINT "ΔΩΣΕ ΚΩΔΙΚΟ ΚΛΑΔΟΥ ΠΕΛΑΤΩΝ"
170 INPUT T$
180 PRINT "ΔΩΣΕ ΟΝΟΜΑ ΚΛΑΔΟΥ" : INPUT Y$
190 PRINT CHR$(26)
200 PRINT "ΠΕΛΑΤΕΣ ΤΟΥ ΚΛΑΔΟΥ";" ";Y$
210 PRINT "-----"
220 OPEN "I",1,"B;ΠΕΛΑΤΕΣ"
230 IF EOF(1) THEN 290
240 INPUT 1,A$,B$,C$,D$,E$,F$,G$
250 IF C$=T$ THEN 270
260 GOTO 230
270 PRINT A$,B$,F$
280 GOTO 230
290 PRINT "-----"
300 CLOSE 1
305 PRINT "ΔΩΣΕ RETURN ΓΙΑ ΣΥΝΕΧΕΙΑ":INPUT O$
308 GOTO 10
310 PRINT CHR$(26)
320 PRINT "ΔΩΣΕ ΟΝΟΜΑ ΠΕΡΙΟΧΗΣ"
330 INPUT X$
340 PRINT CHR$(26)
350 PRINT "ΠΕΛΑΤΕΣ ΣΤΗΝ ΠΕΡΙΟΧΗ";" ";X$
360 PRINT "-----"
370 OPEN "I",1,"B;ΠΕΛΑΤΕΣ"
380 IF EOF(1) THEN 440
390 INPUT 1,A$,B$,C$,D$,E$,F$,G$
400 IF E$=X$ THEN 420
410 GOTO 380
420 PRINT A$,B$,F$
430 GOTO 380
440 PRINT "-----"
450 CLOSE 1
455 PRINT "ΔΩΣΕ RETURN ΓΙΑ ΣΥΝΕΧΕΙΑ":INPUT O$
458 GOTO 10

```

```

460 PRINT CHR$(26)
470 PRINT "ΔΩΣΕ ΚΩΔ ΚΛΑΔΟΥ ΠΕΛΑΤΩΝ" : INPUT M$
480 PRINT "ΔΩΣΕ ΟΝΟΜΑ ΚΛΑΔΟΥ" : INPUT V$
490 PRINT "ΔΩΣΕ ΟΝΟΜΑ ΠΕΡΙΟΧΗΣ" : INPUT N$
500 PRINT CHR$(26)
510 PRINT "ΠΕΛΑΤΕΣ ΚΛΑΔΟΥ"; " "; V$; " "; "ΣΤΗΝ ΠΕΡΙΟΧΗ"; " "; N$
520 PRINT "-----"
530 OPEN "I", 1, "B; ΠΕΛΑΤΕΣ"
540 IF EOF(1) THEN 600
550 INPUT 1, A$, B$, C$, D$, E$, F$, G$
560 IF C$=M$ AND E$=N$ THEN 580
570 GOTO 540
580 PRINT A$, B$, F$
590 GOTO 540
600 PRINT "-----"
605 PRINT "ΔΩΣΕ RETURN ΓΙΑ ΣΥΝΕΧΕΙΑ" : INPUT O$
606 CLOSE 1
608 GOTO 10
620 PRINT CHR$(26)
630 PRINT "ΔΩΣΕ ΚΩΔ ΚΛΑΔΟΥ ΓΙΑ ΔΙΕΥΘΥΝΣΙΟΓΡΑΦΟ": INPUT R$
640 PRINT "ΔΩΣΕ ΟΝΟΜΑ ΚΛΑΔΟΥ": INPUT K$
650 PRINT CHR$(26)
660 LPRINT "ΔΙΕΥΘΥΝΣΕΙΣ-ΕΤΙΚΕΤΤΕΣ ΠΕΛΑΤΩΝ ΚΛΑΔΟΥ"; " "; K$
670 LPRINT "-----"
680 OPEN "I", 1, "B; ΠΕΛΑΤΕΣ"
690 IF EOF(1) THEN 800
700 INPUT 1, A$, B$, C$, D$, E$, F$, G$
710 IF C$=R$ THEN 730
720 GOTO 690
730 LPRINT TAB(3); "ΚΥΡΙΟΥΣ"
740 LPRINT B$
750 LPRINT C$
760 LPRINT D$
770 LPRINT E$
780 LPRINT : PRINT
790 GOTO 690
800 PRINT "-----"
810 CLOSE 1
840 PRINT "ΤΕΛΟΣ ΕΡΓΑΣΙΑΣ"
850 END

```

# ΓΡΑΦΙΚΗ ΑΠΕΙΚΟΝΙΣΗ ΠΩΛΗΣΕΩΝ (SALES—GR)

Το μικρό αυτό πρόγραμμα έχει την δυνατότητα ν' απεικονίζει στην επιθυμητή κλίμακα την εξέλιξη των πωλήσεων μιας εταιρείας, μακροπρόθεσμα.

Στις εντολές DATA καταχωρούμε το ύψος των πωλήσεων κάθε μήνα σε kg, τεμάχια ή ΔΡΧ. Φυσικά, η τελευταία εντολή DATA δίνει τις πωλήσεις των μηνών του

τρέχοντος έτους.

Αυτό που έχει σημασία, για να γνωρίζουμε την εξέλιξη των πωλήσεων μακροχρόνια, είναι να παρουσιάζονται οι πωλήσεις αθροιστικά κάθε μήνα. Δηλαδή, σε κάθε μήνα να εμφανίζονται οι πωλήσεις που έγιναν απ' την αρχή του έτους, μέχρι τότε. Γι' αυτόν το λόγο υπάρχει στην εντολή 110 ο αθροιστής T.

Επίσης, το πρόγραμμα φροντίζει να εμφανίζει και αριθμητικά το ύψος των πωλήσεων "μέχρι τότε" στο σημείο ακριβώς που τελειώνει η κάθε μπάρα της γραφικής παράστασης.

Το μήκος, όμως, της κάθε γραμμής-μπάρας μπορούμε εύκολα να το καθορίζουμε ανάλογα με το διαθέσιμο

χώρο οθόνης ή χαρτιού (όταν χρησιμοποιούμε την LPINT για εκτύπωση με την εντολή 120.

Σε κάθε περίπτωση, ενώνοντας τα τελικά σημεία των γραμμών στους μήνες των διαδοχικών ετών (βλέπε σχέδιο από τον εκτυπωτή EPSON 82FT), μπορούμε να έχουμε οπτικά, αλλά και αριθμητικά την εξέλιξη των πωλήσεων.

```
1 REM ΓΙΩΤΟΠΟΥΛΟΣ ΠΑΝΑΓΙΩΤΗΣ
2 REM ΠΡΟΓΡΑΜΜΑ SALES GRAF
10 LPRINT "ΓΡΑΦΙΚΗ ΠΑΡΑΣΤΑΣΗ ΕΞΕΛΙΞΗΣ ΠΩΛΗΣΕΩΝ"
20 LPRINT "ΤΟΥ ΠΡΟΙΟΝΤΟΣ"
30 LPRINT :LPRINT
40 LPRINT TAB(10);"ΚΛΙΜΑΚΑ 1;500 "
50 LPRINT
60 LPRINT TAB(5);STRING$(45,CHR$(45))
70 LPRINT TAB(5);"I"
80 FOR U=1 TO 12
90 READ S
100 IF S=0 THEN 230
110 T=T+S
120 R=INT(T/100)
130 LPRINT U;TAB(5);"I";STRING$(R,CHR$(47));T
150 NEXT U
160 LPRINT TAB(5);"I"
170 LPRINT TAB(5);"I"
180 T=0
190 GOTO 80
200 DATA 200,300,700,500,100,50,150,200,300,400,300,500
210 DATA 800,1000,200,700,800,500,200,300,210,250,255,400
220 DATA 1200,1500,300,500,300,150,200,400,0
230 END
```

ΓΡΑΦΙΚΗ ΠΑΡΑΣΤΑΣΗ ΕΞΕΛΙΞΗΣ ΠΩΛΗΣΕΩΝ  
ΤΟΥ ΠΡΟΪΟΝΤΟΣ

I  
1 I\*\* 200  
2 I\*\*\*\*\* 500  
3 I\*\*\*\*\* 1200  
4 I\*\*\*\*\* 1700  
5 I\*\*\*\*\* 1800  
6 I\*\*\*\*\* 1850  
7 I\*\*\*\*\* 2000  
8 I\*\*\*\*\* 2200  
9 I\*\*\*\*\* 2500  
10 I\*\*\*\*\* 2900  
11 I\*\*\*\*\* 3200  
12 I\*\*\*\*\* 3700

I  
I  
1 I\*\*\*\*\* 800  
2 I\*\*\*\*\* 1800  
3 I\*\*\*\*\* 2000  
4 I\*\*\*\*\* 2700  
5 I\*\*\*\*\* 3500  
6 I\*\*\*\*\* 4000  
7 I\*\*\*\*\* 4200  
8 I\*\*\*\*\* 4500  
9 I\*\*\*\*\* 4710  
10 I\*\*\*\*\* 4960  
11 I\*\*\*\*\* 5215  
12 I\*\*\*\*\* 5615

I  
I  
1 I\*\*\*\*\* 1200  
2 I\*\*\*\*\* 2700  
3 I\*\*\*\*\* 3000  
4 I\*\*\*\*\* 3500  
5 I\*\*\*\*\* 3800  
6 I\*\*\*\*\* 3950  
7 I\*\*\*\*\* 4150  
8 I\*\*\*\*\* 4550

# ΕΛΕΓΧΟΣ ΠΡΟΜΗΘΕΙΩΝ (SUPPLIER III)

Με το πρόγραμμα αυτό δίνει μια εύκολη και ταχύτατη ενημέρωση για όσους διευθυντές προμηθειών χρησιμοποιούν μικρούπολογιστή για την παρακολούθηση των συνθηκών αγοράς των προϊόντων που προμηθεύονται. Βασίζεται στην ταξινόμηση-σορτάρισμα τριών κλειδιών 1, 2, 3, που καθένα τους αντιστοιχεί σε μια συγκεκριμένη πληροφορία DATA.

Κατά την προμήθεια προϊόντων, στην πράξη, τρία βασικά στοιχεία ενδιαφέρουν. Πρώτα, το όνομα της πηγής προμήθειας δηλαδή του προμηθευτή. Δεύτερον, το όνομα του προϊόντος που προμηθευόμαστε. Τρίτο, η ημερομηνία που πήραμε την προσφορά από τον προμηθευτή και η τιμή. Έτσι, στο DATA BLOCK

του προγράμματος εγγράφουμε τις πληροφορίες συμπληρώνοντας τα τρία "κλειδιά" σε κάθε εγγραφή. Κάθε κλειδί πρέπει υποχρεωτικά ν' αρχίζει με το "όνομα προμηθευτή", "ημερομηνία προσφοράς-τιμή", αλλά είναι δυνατή η συμπλήρωση και άλλων χρήσιμων στοιχείων μετά τα παραπάνω στοιχεία, όπως π.χ. τηλέφωνο μετά το όνομα του προμηθευτή, οίκος παραγωγής μετά το όνομα του προϊόντος κλπ.

Με την εντολή 130 ζητάμε τον αριθμό του κλειδιού με το οποίο να γίνει η ταξινόμηση. Αν δώσουμε 1 για όνομα προμηθευτή στον πίνακα που εμφανίζεται στην οθόνη, η πρώτη στήλη, με τα ονόματα των προμηθευτών, είναι σορταρισμένη κατ'αλφαβητική

σειρά. Έτσι, απέναντι από το όνομα του προμηθευτή που εύκολα βρίσκουμε (πατώνοντας την κατάσταση που κάνει SCRICKUNG με το πλήκτρο CONTROL C) έχουμε το προϊόν ή τα προϊόντα που μας πρόσφερε στο παρελθόν (2η στήλη) και την ημερομηνία προσφοράς καθώς και την τιμή (3η στήλη).

Με το κλειδί 2 έχουμε την κατάσταση με τα προϊόντα τώρα σε αλφαβητική σειρά, οπότε για συγκεκριμένο προϊόν, παίρνουμε τους προμηθευτές και τις προσφορές τους.

Τέλος, με το κλειδί 3 έχουμε την τρίτη στήλη των ημερομηνιών προσφοράς σορταρισμένες κατ'αυξάνουσα σειρά (π.χ. από 010183 έως 123183), οπότε για συγκεκρι-

μένη ημερομηνία βλέπουμε ποιοι και για τι μας έκαναν προσφορές. Ας σημειώσουμε ότι ο τρόπος εγγραφής της ημερομηνίας προσφοράς στις εντολές DATA γίνεται σε μορφή ΧΧΥΥΖΖ, όπου ΧΧ είναι ο αριθμός του μήνα από 01 έως 12, ΥΥ είναι η μέρα από 01 έως 31 και ΖΖ ο χαρακτηριστικός αριθμός για το έτος, π.χ. 83.

Το πρόγραμμα χρησιμοποιεί (εντολές 160-330) μια ρουτίνα σορταρίσματος που είναι αυτοτελής και χρήσιμη και για άλλες εφαρμογές, με κατάλληλη προσαρμογή. Έτσι, π.χ. μπορεί να σορτάρει στοιχεία ενός αρχείου, αρκεί να το "αυξήσουμε" και να το "διαβάσουμε" με τις κατάλληλες εντολές της MBA SIC.

```
1 REM ΓΙΩΤΟΠΟΥΛΟΣ Π -- ΠΡΟΓΡΑΜΜΑ SUPPLIER III
2 PRINT CHR$(26)
3 PRINT TAB(12);"ΤΟ ΠΡΟΓΡΑΜΜΑ ΑΥΤΟ ΤΑΞΙΝΟΜΕΙ"
4 PRINT TAB(13);"ΚΑΤΑ ΣΕΙΡΑ, ΕΝΑ ΑΠΟ ΤΑ ΤΡΙΑ"
5 PRINT TAB(13);"ΣΤΟΙΧΕΙΑ-ΚΛΕΙΔΙΑ ΤΟΥ ΑΡΧΕΙΟΥ"
6 PRINT:PRINT
```

```

7 PRINT TAB(12); "ΠΡΟΜΗΘΕΥΤΗΣ=1, ΠΡΟΙΟΝ-ΟΙΚΟΣ=2"
8 PRINT TAB(14); "ΗΜΕΡΟΜΗΝΙΑ-ΠΡΟΣΦ-ΤΙΜΗ=3"
9 PRINT TAB(11); STRING$(30, CHR$(150))
10 PRINT:PRINT
20 PRINT "ΕΝΑΡΞΗ ΠΡΟΓΡΑΜΜΑΤΟΣ"
30 DIM A$(10,4), P(10) :N=0
40 FOR R=1 TO 10
50 FOR F=1 TO 3
60 READ A$(R,F)
70 IF A$(R,F)="R" THEN 110
80 NEXT F
90 N=N+1
100 NEXT R
110 FOR I=1 TO N :P(I)=I :NEXT I
120 REM "SELECT FIELD ON WHICH TO DO SORT"
130 PRINT TAB(8); "ΔΡΣΕ ΣΕ ΠΟΙΟ ΚΛΕΙΔΙ-ΣΤΟΙΧΕΙΟ"
132 PRINT TAB(8); "ΘΕΛΕΙΣ ΝΑ ΓΙΝΕΙ Η ΤΑΞΙΝΟΜΗΣΗ-1,2,3";:INPUT F
140 IF F<1 OR F>3 THEN 130
150 IF N=1 THEN 300
160 REM "SORTING ROUTINE"
170 FOR JOO=2 TO N
180 I=JOO-1:T=P(JOO)
190 IF I<=0 THEN 240
200 IF A$(T,F)>=A$(P(I),F) THEN 240
210 P(I+1)=P(I)
220 I=I-1
230 GOTO 190
240 P(I+1)=T
250 NEXT JOO
260 REM "PRINT RECORDS SORTED BY SPECIFIED FIELD"
300 PRINT "ΠΛΗΡΟΦΟΡΙΕΣ ΚΑΤΑ ΣΕΙΡΑ ΣΤΟΙΧΕΙΟΥΔ";F;"/////":PRINT
310 FOR R=1 TO N
320 FOR F=1 TO 3:PRINT TAB(13*(F-1));A$(P(R),F);:NEXT F:PRINT
325 FOR U=1 TO 500 :NEXT U
330 NEXT R
340 STOP
350 REM "STRING DATA FOR RECORDS"
360 DATA "ΜΟΣΧΟΛΙΟΣ", "ΤΡΙΠΟΛΥΦΘΣΦΑΤ-ΓΑΛ", "210383C*300D0Λ"
370 DATA "ΛΟΥΒΗΣ", "ΑΛΜΠΟΥΜΙΝΗ-ΣΕΥΜΟΥΡ", "210983C-650DPX"
380 DATA "ΜΟΣΧΟΛΙΟΣ", "ΠΕΡΥΝΤΡΟΛ*35-ΔΕΓΟΥΣΑ", "120583C-75DPX"
390 DATA "ΡΟΥΜΠΟΥΛΑΚΗΣ", "ΜΕΝΒΟΛ-ΚΙΝΑΣ", "310283C-1800DPX"
400 DATA "ΓΚΟΛΣΜΙΤ", "ΤΕΓΚΙΝ ΣΠΕΣΙΑΛ", "310283C-600DPX"
410 DATA "R", "R", "R"
420 END

```

## Computer generated ποιήματα

Ο πρώτος διαγωνισμός του περιοδικού μας, το *PIXELMANIA ONE*, είναι αφιερωμένος στα έλληνικά computer generated ποιήματα. Το *PIXELMANIA* θα πλαισιώνεται πάντα από ένα απλό και γενικοποιημένο πρόγραμμα σε standard Basic σχετικό με το θέμα του διαγωνισμού. Θα υπάρχουν ακόμα επεξηγήσεις, παραδείγματα και appendixes/συμπληρώματα με επιπλέον στοιχεία και βοηθητικά προγράμματα όπως επίσης και νύξεις για περαιτέρω εξέλιξη του βασικού προγράμματος.

Επιμέλεια ύλης: Έκτωρ Χαραλάμπους

### Εισαγωγικό σημείωμα

Ενα micro δεν είναι βέβαια από μόνο του σε θέση να αρχίσει να γράφει ποιήματα, να ζωγραφίζει πίνακες ή και να συλλαμβάνει ιδέες. Ένας user όμως, έχει τη δυνατότητα να φτιάξει ένα πρόγραμμα για το micro του ικανό να κάνει τυχαίες επιλογές λέξεων και να τις τοποθετεί τελικά πάνω σε ένα σταθερό συντακτικό/γραμματικό σκελετό. Εννοείται βέβαια ότι αυτός ο γραμματικός σκελετός μπορεί και να διαφέρει από στίχο σε στίχο, και επίσης με την προσθήκη ειδικών Conditional statements μπορεί να ελέγχει την ομοιοκαταληξία, τον αριθμό συλλαβών του κάθε στίχου ή ότι άλλο επιθυμεί.

Τις περισσότερες μάλιστα φορές το

αποτέλεσμα είναι ουσιαστικά μια αρκετά παραδεκτή προσπάθεια δημιουργίας ενός ποιήματος. Συχνά όμως, το micro τυπώνει μια χαζοχαρούμενη αλλά διασκεδαστική και ευρηματική παρωδία ποιήματος. Το λεξιλόγιο και η όλη δομή του ποιήματος, έχουν φυσικά τον τελικό λόγο όσον αφορά την ποιότητα ή το στυλ του αποτελέσματος, η λίγο περισσότερο προσεγμένη δομή (semantic schema) δίνει στα "ποιήματα" πολύ μεγαλύτερη συνοχή.

Το πρόγραμμα που σας παρουσιάζουμε παρακάτω είναι αρκετά απλό ώστε να είναι κατανοητό και user-friendly στο ευρύ κοινό ενώ παράλληλα δεν προβάλλει τον ισοχυρισμό του ότι έχει artificial intelligence.

### Ποιήματα στο micro σας

Διαλέξαμε σαν θέμα του διαγωνισμού μας τη δημιουργία ποιημάτων τύπου haiku, με microcomputers, στα ελληνικά. Τα haiku είναι μια εξειδικευμένη μορφή ιαπωνικής ποίησης όπου το κάθε ποίημα έχει τρεις μόνο στίχους που περιέχουν συνολικά δεκαεπτά συλλαβές. Το σχετικά μικρό τους μέγεθος, η απλότητα στη δομή, και η δημοτικότητα τους όχι μόνο στην τωρινή Ιαπωνία αλλά και στους κύκλους αυτών που ασχολούνται με computer generated texts μας δίνει τη δυνατότητα να σας παρουσιάσουμε αρκετά παραδείγματα ποιημάτων haiku και ένα βασικό πρόγραμμα σε standard basic ικανό να τα συνθέτει.

## Ποιήματα Haiku από υπολογιστές

Τα παραδείγματα που ακολουθούν, είναι το αποτέλεσμα μιας on-line αλληλεπίδρασης ανθρώπου-υπολογιστή στο Cambridge Research Unit, και αναφέρονται στη χρήση υπολογιστών για τη σύνθεση ποιημάτων. Το πρόγραμμα είχε γραφτεί σε γλώσσα TRAC<sup>1</sup>.

Ένα "simulation" αυτού του προγράμματος σε standard basic, το listing του οποίου δεν υπάρχει στο βιβλιογραφικό υλικό που έχουμε στα χέρια μας, θα βρείτε λίγο παρακάτω. Το πρόγραμμα σε γλώσσα TRAC είναι ένα frame με slots μέσα στα οποία ο χειριστής τοποθετεί διάφορες λέξεις. Στα παραδείγματα, "1 Poem" και "2 Poem", ο χειριστής διαλέγει τις λέξεις που θέλει εκείνος, στα άλλα δυο όμως παραδείγματα που τιτλοφορούνται: "3 Poem", η επιλογή λέξεων ρυθμίζεται από διάφορους περιορισμούς σχετικούς πάντα με τις κατευθύνσεις των δεικτών, (και τους διάφορους αριθμούς που ακολουθούν τους δείκτες και αναφέρονται στα υπόλοιπα slot του πίνακα 1) που υπάρχουν πάνω από κάθε slot του λεξιλογίου/αρχείου με τα συνώνυμα και τα συγγενικά που έχουν διαλεχτεί για να

#### 1 Poem

eons deep in the ice  
I paint all time in a whorl  
bang the sludge has cracked

#### 2 Poem

eons deep in the ice  
I see gelled time in a whorl  
pffftt the sludge has cracked

#### 3 Poem

all green in the leaves  
I smell dark pools in the trees  
crash the moon has fled

#### 3 Poem

all white in the buds  
I flash snow peaks in the spring  
bang the sun has fogged

Slot 1 (→4) (→5)	Slot 2 (→5) (→6)	Slot 3 (→5)	Slot 4 (→6) (→7)	Slot 5 (→8)	Slot 6 (→2)	Slot 7	Slot 8 (→5)	Slot 9 (→5) (→8)
White	Buds	See	Snow	Trees	Spring	Bang	Sun	Flit
Blue	Twigs	Trace	Tall	Peaks	Full	Hush	Moon	Fled
Red	Leaves	Glimpse	Pale	Hills	Cold	Swish	Star	Dimmed
Black	Hills	Flash	Dark	Streams	Heat	Pffftt	Cloud	Cracked
Grey	Peaks	Smell	Faint	Birds	Sun	Whizz	Storm	Passed
Green	Snow	Taste	White	Specks	Shade	Flick	Streak	Shrunk
Brown	Ice	Hear	Clear	Arcs	Dawn	Shoo	Tree	Smashed
Bright	Sun	Seize	Red	Grass	Dusk	Grrr	Flower	Blown
Pure	Rain		Blue	Stems	Day	Whirr	Bud	Sprung
Curved	Cloud		Green	Sheep	Night	Look	Leaf	Crashed
Crowned	Sky		Grey	Cows	Mist	Crash	Child	Gone
Starred	Dawn		Black	Deer	Trees		Crane	Fogged
	Dusk		Round	Stars	Woods		Bird	Burst
	Mist		Square	Clouds	Hills		Plane	
	Fog		Straight	Flowers	Pools		Moth	
	Spring		Curved	Buds				
	Heat		Slim	Leaves				
	Cold		Fat	Trees				
			Burst	Pools				
			Thin	Drops				
			Bright	Stones				
				Bells				
				Trails				

1. Σήμερα η γλώσσα TRAC είναι σχετικά άγνωστη στο πλατύ κοινό, τα πειράματα όμως με computer generated poems, που άρχισαν τη δεκαετία του 60, συνεχίζονται, είναι όμως γραμμένα σε πιο σύγχρονες γλώσσες ηλεκτρονικών υπολογιστών και περισσότερο σύνθετα.



# Κερδίστε ένα Jupiter Ace

χρησιμοποιηθούν σαν δομικό υλικό της σειράς των ποιημάτων.

Το πρόγραμμα που ακολουθεί είναι γραμμένο σε σχετικά προσηχειακή μορφή, ούτως ώστε, εκτός από ότι είναι περισσότερο κατανοητό, να έχετε ευχέρεια και τα περιθώρια να κάνετε διάφορες επεμβάσεις ή και βελτιώσεις. Στην ουσία όμως, είναι μια απλοποιημένη "αναπαραγωγή" του αρχικού προγράμματος των Margaret Masterman και Robin Mckinnon Wood σε Standard Basic.

Το λεξιλόγιο που χρησιμοποιήσαμε στο πρόγραμμα λόγω microλακωνισμού δεν περιέχει όλες τις λέξεις που βρίσκονται στα slot, αν θέλετε μπορείτε να τις προσθέσετε όλες ανεξαιρέτως αφού συμβουλευθείτε το σχετικό πίνακα (listin 1).

Χρησιμοποιώντας αυτούς του δείκτες, μπορεί να φτιάξει κανείς ένα semantic schema (σημειολογικό διάγραμμα, σχεδιάγραμμα σημαντικής) του haiku (εικ. 1) που δείχνει ότι το semantic center (σημειολογικό κέντρο) του ποιήματος - με τα πέντε βέλη που ηγαίνουν προς αυτό και το ένα που αρχίζει από αυτό - είναι τοποθετημένο στο slot υπ' αριθμόν 5. Αυτά τα ποιήματα, και τα σχετικά προγράμματα, έχουν γραφτεί από την Margaret Masterman και τον Robin Mckinnon, Wood. Ο τύπος αυτών των ποιημάτων ονομάζεται Computerized japanese haiku και παρόλο το "Japanese" του τίτλου μπορούν να γραφτούν σε οποιαδήποτε γλώσσα του κόσμου, από ελληνικά μέχρι αφγανικά, αλβανικά και ούτω καθεξής, εννοείται φυσικά ότι χρησιμοποιεί και την ανάλογη γραμματική και λεξιλόγιο. Τίποτα δεν αποκλείει επίσης και την περίπτωση να υπάρχουν ήδη στα ελληνικά ποιήματα, γραμμένα από ποιητές, και τα οποία θα μπορούσαν να θεωρηθούν ...haiku αλά ελληνικά.

## LISTING 1

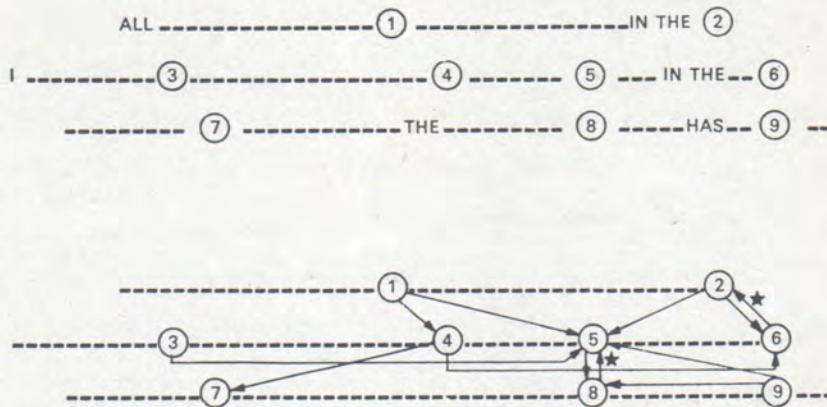
```

1 REM HAIKU
30 REM CONDENSED VERSION
40 REM OF THE ORIGINAL PROGRAM
50 LET l=INT (RND*65535)
60 LET r=INT (RND*26+65)
90 GO TO INT (RND*8+100)
99 REM SLOT 1 (linked with SLO
T 4 and SLO 5)
100 LET a$="white": LET a=2: GO
TO INT (RND*8+200)
101 LET a$="blue": LET a=1: GO
TO INT (RND*8+200)
102 LET a$="red": LET a=1: GO T
O INT (RND*8+200)
103 LET a$="black": LET a=1: GO
TO INT (RND*8+200)
104 LET a$="grey": LET a=1: GO
TO INT (RND*8+200)
105 LET a$="green": LET a=1: GO
TO INT (RND*8+200)
106 LET a$="brown": LET a=1: GO
TO INT (RND*8+200)
107 LET a$="bright": LET a=1: G
O TO INT (RND*8+200)
109 REM SLOT 2 (linked with SLO
T 5 and SLO 6)
200 LET b$="buds": LET b=1: GO
TO INT (RND*8+300)
201 LET b$="twigs": LET b=1: GO
TO INT (RND*8+300)
202 LET b$="leaves": LET b=2: G
O TO INT (RND*8+300)
203 LET b$="hills": LET b=1: GO
TO INT (RND*8+300)
204 LET b$="peaks": LET b=1: GO
TO INT (RND*8+300)
205 LET b$="snow": LET b=1: GO
TO INT (RND*8+300)
206 LET b$="ice": LET b=2: GO T
O INT (RND*8+300)
207 LET b$="sun": LET b=1: GO T
O INT (RND*8+300)
209 REM SLOT 3 (linked with SLO
T 5)
300 LET c$="see": LET c=1: GO T
O INT (RND*8+400)
301 LET c$="trace": LET c=2: GO
TO INT (RND*8+400)
302 LET c$="glimpse": LET c=2:
GO TO INT (RND*8+400)
303 LET c$="flash": LET c=1: GO
TO INT (RND*8+400)
304 LET c$="smell": LET c=1: GO
TO INT (RND*8+400)
305 LET c$="taste": LET c=2: GO
TO INT (RND*8+400)
306 LET c$="hear": LET c=1: GO
TO INT (RND*8+400)
307 LET c$="size": LET c=2: GO
TO INT (RND*8+400)

```

## 3 POEM

### ΣΗΜΑΣΕΙΟΛΟΓΙΚΟ ΣΧΕΔΙΑΓΡΑΜΜΑ SEMANTIC SCHEMA



\* Ο άστερίσκος υποδηλώνει μία δυαδική σύνδεση. Για να είναι το όλο σύστημα computable, πρέπει να επιλεγεί ένα από τα βέλη με άστερίσκο.

EIKONA 1

```

399 REM SLOT 4 (linked with SLO
T 6 and SLO 7)
400 LET d$="snow": LET d=1: GO
TO INT (RND*8+500)
401 LET d$="fall": LET d=1: GO
TO INT (RND*8+500)
402 LET d$="pale": LET d=2: GO
TO INT (RND*8+500)
403 LET d$="dark": LET d=1: GO
TO INT (RND*8+500)
404 LET d$="faint": LET d=1: GO
TO INT (RND*8+500)
405 LET d$="white": LET d=2: GO
TO INT (RND*8+500)
406 LET d$="clear": LET d=1: GO
TO INT (RND*8+500)
407 LET d$="arc": LET d=1: GO T
O INT (RND*8+500)
499 REM SLOT 5 (linked with SLO
T 6)
500 LET e$="fress": LET e=1: GO
TO INT (RND*8+600)
501 LET e$="peaks": LET e=1: GO
TO INT (RND*8+600)
502 LET e$="hills": LET e=1: GO
TO INT (RND*8+600)
503 LET e$="streams": LET e=1:
GO TO INT (RND*8+600)
504 LET e$="birds": LET e=1: GO
TO INT (RND*8+600)
505 LET e$="specks": LET e=1: G
O TO INT (RND*8+600)
506 LET e$="arc": LET e=1: GO
TO INT (RND*8+600)
507 LET e$="grass": LET e=1: GO
TO INT (RND*8+600)
599 REM SLOT 6 (linked with SLO
T 7)
600 LET f$="spring": LET f=1: G
O TO INT (RND*8+700)
601 LET f$="full": LET f=1: GO
TO INT (RND*8+700)
602 LET f$="cold": LET f=1: GO
TO INT (RND*8+700)
603 LET f$="heat": LET f=1: GO
TO INT (RND*8+700)
604 LET f$="sun": LET f=1: GO T
O INT (RND*8+700)
605 LET f$="shade": LET f=2: GO
TO INT (RND*8+700)
606 LET f$="dawn": LET f=1: GO
TO INT (RND*8+700)
607 LET f$="dusk": LET f=1: GO
TO INT (RND*8+700)
699 REM SLOT 7
700 LET g$="bang": LET g=1: GO
TO INT (RND*8+800)
701 LET g$="hush": LET g=1: GO
TO INT (RND*8+800)
702 LET g$="swish": LET g=1: GO
TO INT (RND*8+800)
703 LET g$="pffft": LET g=1: G
O TO INT (RND*8+800)
704 LET g$="whizz": LET g=1: GO
TO INT (RND*8+800)
705 LET g$="flick": LET g=1: GO
TO INT (RND*8+800)
706 LET g$="shoo": LET g=1: GO
TO INT (RND*8+800)
707 LET g$="grf": LET g=1: GO
TO INT (RND*8+800)
799 REM SLOT 8 (linked with SLO
T 5)
800 LET h$="sun": LET h=1: GO T
O INT (RND*8+900)
801 LET h$="moon": LET h=1: GO
TO INT (RND*8+900)
802 LET h$="star": LET h=1: GO
TO INT (RND*8+900)
803 LET h$="cloud": LET h=1: GO
TO INT (RND*8+900)
804 LET h$="storm": LET h=1: GO
TO INT (RND*8+900)
805 LET h$="streak": LET h=1: G
O TO INT (RND*8+900)
806 LET h$="tree": LET h=1: GO
TO INT (RND*8+900)
807 LET h$="flower": LET h=2: G
O TO INT (RND*8+900)
899 REM SLOT 9 (linked with SLO
T 5 and SLO 6)
900 LET j$="flit": LET j=1: GO
TO 4000
901 LET j$="fled": LET j=1: GO
TO 4000
902 LET j$="dimmed": LET j=2: G
O TO 4000
903 LET j$="cracked": LET j=2:
GO TO 4000
904 LET j$="passed": LET j=2: G
O TO 4000
905 LET j$="shrunk": LET j=1: G
O TO 4000
906 LET j$="smashed": LET j=2:
GO TO 4000
907 LET j$="blown": LET j=1: GO
TO 4000
4000 REM SEMANTIC SCHEMA
4100 REM r$ is the first line of
the haiku, t$ the second and u$
is the third line.
5000 LET r$="all "+a$+" in the "
+b$: LET t$="I "+c$+" +d$+" +e
$+" in the "+f$: LET u$=g$+" the
"+h$+" has "+j$
5100 REM CONDITIONAL STATEMENTS
5200 REM FOR A FIRST LINE WITH 5
SYLLABLES, A SECOND LINE WITH 7
AND A THIRD LINE WITH 5 SYLLAB
LE 3. THUS THE TOTAL NUMBER OF SYLL
ABLES IN A HAIKU POEM IS SEVENTE
EN.
5250 REM THE SLOTS 1-9 ARE SUPPO
SED TO CONTAIN MONOSYLLABLE WORD
S. IN CASE YOU ARE IN AGREEMENT W
ITH THIS REMOVE THE VERY NEXT CO
NDITIONAL STATEMENT. IN THE OPPOS
ITE CASE, ACTIVATE IT BY LIFTING
THE REM.
5300 REM IF a+b<2 OR c+d+e+f<4
OR g+h+j<3 THEN GO TO 90
5400 REM FOR A 32 CHARACTER MAXI
MUM LINE LENGTH,
5500 REM IF LEN t$>32 THEN RANDO
MIZE: RANDOMIZE: GO TO 90
5600 REM READ AGAIN THE SUBTITLE
UNDER THE SEMANTIC SCHEMA FOR C
LARIFICATION. YOUR INTERPRETATION
MAY BE DIFFERENT THAN THAT ASSU
MED BY THE FOLLOWING CONDITIONAL
STATEMENT.
5700 IF b$=f$ AND e$=h$ THEN RAN
DOMIZE: GO TO 90
5800 PRINT ""
5900 PRINT ""
6000 LET m$="POEM": LET s$=STR$
VAL "1 "+c$+" +CHR$ r+m$: PRINT
"
6100 PRINT r$: t$: u$: LET l=1+
6200 REM PRINT: "FREE MEMORY" =
: 65536-USR 7962
6300 GO TO 90
6400 STOP

```

# PIXELMANIA ONE

## Ανάλυση του προγράμματος

50-60: Οι γραμμές αυτές δίνουν έναν random alphanumeric κωδικό αριθμό στο κάθε haiku που συνθέτει το πρόγραμμα. Στη γραμμή 6100 το LET 1=1+1 δίνει αύξοντες κωδικούς αριθμούς σε κάθε νέο RUN του προγράμματος στα διαφορετικά και πολυπληθή haiku που παράγει... Η γραμμή 90 διαλέγει τυχαία μια από τις λέξεις του Slot 1, της aS, πηγαίνοντας σε μια από τις γραμμές 100-107 ή INT (RND\*8+100) και οι οποίες περιέχουν το λεξιλόγιο του Slot 1. Η γραμμή INT (RND\*8+100) αφού ορίσει το aS, μια από τις λέξεις του Slot 1 και τον αριθμό των συλλαβών της λέξης, LET a=1 ή LET a=2, με το GO TO INT (RND\*8+200) που περιέχει στο τέλος μεταφέρει την όλη διαδικασία ξανά σε μια από τις γραμμές 200-207 ή INT (RND\*8+200) και η οποία με τη σειρά της ορίζει το bS, μια από τις λέξεις του Slot 2, και τον αριθμό των συλλαβών... και ούτω καθεξής. Τελικά αυτή η τυχαία "σάρωση" ορίζει και μια λέξη από κάθε ένα από τα εννέα Slot.

5000: Αυτή η γραμμή ορίζει και "στήνει" βάσει των προδιαγραφών του semantic schema/σημασιολογικού σχεδιαγράμματος της εικόνας 1 τις τρεις γραμμές/strings rS, tS και uS που είναι και οι τρεις στίχοι του ποιήματος. Ακολουθούν γραμμές με conditional statements που συνοδεύονται πάντα από REM statements σχετικές με τη λειτουργικότητά τους.

5300: Εδώ γίνεται ένας έλεγχος σχετικά με το αν η πρώτη γραμμή του

ποιήματος έχει 5 συλλαβές, αν η δεύτερη έχει 7 και αν η τρίτη έχει πάλι 5 συλλαβές. Αυτός ο περιορισμός είναι μεν αναγκαίος για το φτιάξιμο ποιημάτων haiku, στην προκειμένη όμως περίπτωση, όπου τα Slot εννοείτε ότι περιέχουν μονοσύλλαβες λέξεις, και για αυτό ειδικά το σημασιολογικό σχεδιάγραμμα η γραμμή αυτή δεν είναι απαραίτητη. Μπορεί φυσικά και να αμφισβάλλει κανείς για το κατά πόσο οι λέξεις: dimmed, cracked κλπ. είναι μονοσύλλαβες ή όχι, αλλά αυτό είναι άλλο θέμα.

5500: Όσα ποιήματα έχουν στο δεύτερο στίχο περισσότερους από 32 χαρακτήρες απορρίπτονται, καταχρηστικά βέβαια, από αυτό το conditional statement. Ένα REM στην αρχή της γραμμής το κρατά ουδετεροποιημένο.

5700: Η γραμμή αυτή ρυθμίζει ίσως αυτό που υπαινίσσεται ο αρκετά ενδιαφέρον υπότιτλος του σημασιολογικού σχεδιαγράμματος και ο οποίος λέει χαρακτηριστικά τα εξής: "για να είναι το όλο σύστημα computable, πρέπει να επιλεγεί μόνο ένα από τα βέλη με αστερίσκο". Αν έχετε διαφορετικές απόψεις, αλλάξτε τη γραμμή αυτή σύμφωνα με το δικό σας σκεπτικό.

6000: Εδώ καθορίζεται το STRS που τιτλοφορεί το ποιήμα. Η γραμμή 6100 δίνει αύξοντα αριθμό στο αριθμητικό μέρος αυτού του alphanumeric STRS/τίτλου, εφόσον το RUN σας παράγει ποιήματα εν σειρά, και τυπώνει ταυτόχρονα το ποιήμα. 6200: Με τη γραμμή αυτή μπορείτε

να τσεκάρετε το πόση μνήμη έχετε ελεύθερη, ανάλογα με το ποιο ακριβώς micro έχετε πρέπει φυσικά να κάνετε αλλαγές αφού συμβουλευθείτε το manual.

6300: Από εδώ με ένα GO TO 90 πηγαίνουμε στην αρχή του προγράμματος οπότε και δημιουργείται άλλο ένα ποιήμα και ούτω καθεξής...

Όταν κάνετε RUN, θα παρατηρήσετε ότι το πρόγραμμα είναι σε θέση να συνθέτει μια φαινομενικά ατελείωτη σειρά από ποιήματα. Άλλα από αυτά, λίγα πάντως, είναι παραδεκτά ενώ πάρα πολλά είναι απαράδεκτα, για διάφορους λόγους, ή τουλάχιστον σουρεαλιστικά. Αν μάλιστα γεμίσετε τα διάφορα slot με όλες τις λέξεις του πίνακα 1 και βρείτε την ιδανικότερη διατύπωση, για το conditional statement της γραμμής 5700, το simulation του original προγράμματος θα είναι ακόμα πιο κοντά στο real thing.

Τελικά όμως, ρωτήστε τον εαυτό σας: γιατί από όλες αυτές τις εκατοντάδες, αν όχι χιλιάδες, ποιήματα που συνθέτει το πρόγραμμα οι μελετητές δημοσίευσαν τελικά στη σχετική αρθρογραφία δυο και μόνο δυο ποιήματα, τα "3 Poem" που είδαμε προηγουμένως. Είναι άραγε θέμα προσθήκης και άλλων conditional statements έτσι ώστε το πρόγραμμα να τυπώνει τελικά δυο μόνο ποιήματα; είναι θέμα user-choice από το πλήθος των ποιημάτων; ή συμβαίνει τελικά κάτι άλλο; στο Appendix II/Συμπλήρωμα II θα βρείτε ορισμένες αναφορές σχετικά με αυτό το ερώτημα.

## Οι όροι του διαγωνισμού PIXELMANIA ONE

Ο πρώτος διαγωνισμός του Pixel αρχίζει με θέμα τα ελληνικά computer generated haiku, όχι για λόγους ξενομανίας ή ότι άλλο, το σκεπτικό μας είναι περισσότερο πρακτικό από όσο ίσως δείχνει: οι άλλες μορφές ποιημάτων που δημιουργούνται με computer generated texts είναι πολύπλοκότερες και είναι κάτι που θα προτιμούσαμε να δούμε μαζί σας λίγο αργότερα.

Τα Appendices/Συμπλήρωματα που βρίσκονται στο τέλος του άρθρου και αμέσως μετά από τους όρους του διαγωνισμού, δίνουν διάφορα επιπλέον στοιχεία σχετικά με τα haiku. Αρχικά στο Appendix I ο Ernest Wood, διάσημος μελετητής του Zen, κάνει μια σύντομη αλλά διαφωτιστική ανφορά στα man made haiku δίνοντας ταυτόχρονα πολλά παραδείγματα. Στο Appendix II, ο Douglas Hofstadter, ο πασίγνωστος Computer Scientist και αρθρογράφος του Metamagical Themas στο Scientific American, εξετάζει με τη σειρά του και από την πλευρά του τα computer generated haiku με αποσπάσματα από το περίφημο βιβλίο του: GODEL, ESCHER, BACH: AN ETERNAL GOLDEN BRAID.

Μια παράλληλη μελέτη και των δυο όψεων του θέματος νομίζουμε ότι θα σας βοηθήσει αρκετά! Στο Appendix III, τελικά, θα βρείτε διάφορα listings, τα οποία μπορείτε να ενσωματώσετε αν θέλετε στο πρόγραμμά μας σχετικά με τη δημιουργία ελληνικών computer generated haiku, όπως προγράμματα που "γράφουν" ελληνικούς χαρακτήρες κ.ά. Αυτά τα Appendices/Συμπλήρωματα

έχουν φυσικά απώτερο σκοπό όχι μόνο την εξοικείωσή σας στα haiku, αλλά και την παρουσίαση αρκετών παραδειγμάτων, και απόψεων επάνω στο θέμα, ώστε να έχετε αρκετές προλαμβάνουσες παραστάσεις και δυνατότητες για να συμμετάσχετε στο διαγωνισμό μας.

### Κερδίζουν από ένα Point όσοι:

Χρησιμοποιώντας το παραπάνω πρόγραμμα με λίγες (ή και καθόλου) αλλαγές) με δικά τους (6 μέχρι 10) νέα slot (και τα οποία περιέχουν 10 τουλάχιστον ΕΛΛΗΝΙΚΕΣ λέξεις ανά slot) και με ένα δικό τους σταθερό σημασιολογικό σχεδιάγραμμα είναι σε θέση να συνθέτουν δεκάδες από ελληνικά computer generated haiku. Το πρόγραμμα αυτό πρέπει να σταλεί μαζί με τον πίνακα με τα slot των ελληνικών λέξεων, το σημασιολογικό σχεδιάγραμμα και δυο από τα καλύτερα ποιήματα που συνθέτει στα γραφεία του περιοδικού μας εντός δυο μηνών. Όσοι έχουν προβλήματα με τη μνήμη του micro τους, έχουν το δικαίωμα να χρησιμοποιήσουν slot με λιγότερες λέξεις.

### Κερδίζουν από τρία έως πέντε Points όσοι:

Βελτιώνουν ριζικά, εμφανώς και ευρηματικά το παραπάνω πρόγραμμα με χρήση DATA και νέων CONDITIONAL statements ή ότι άλλο θεωρούν αναγκαίο. Τα slot του προγράμματος αυτού πρέπει να είναι από 9 μέχρι 17 και να περιέχουν από 10 τουλάχιστον διαφορετικές ΕΛΛΗΝΙΚΕΣ λέξεις το καθένα, μπορούν όμως και να έχουν ορισμένες κοινές λέξεις μεταξύ τους. Το σημασιολογικό σχεδιάγραμμα πρέπει να είναι σταθερό. Το πρόγραμμα πρέπει να ομιλεί τη Standard Basic κατά τα COMMANDS κλπ. και τα ΕΛΛΗΝΙΚΑ κατά τις γραφές, έχοντας παράλληλα τη δυνατότητα, τη φαντασία και τη μνήμη να συνθέτει εκατοντάδες ελληνικά computer generated haiku, από τα οποία τα 2% τουλάχιστον πρέπει να είναι σχετικώς παραδεκτά.

Το πρόγραμμα αυτό, πρέπει να σταλεί μαζί με τα slot των ελληνικών λέξεων, το σημασιολογικό σχεδιάγραμμα και τέσσερα από τα καλύτερα σας ποιήματα στα γραφεία του περιοδικού μας εντός δυο μηνών. Μέχρι εδώ και εφόσον η όλη σας μελέτη είναι αξιόλογη και πλήρης κερδίζετε 3 points.

Αν το πρόγραμμά σας είναι κάτι το ιδιαίτερα εξαιρετικό, τότε κερδίζετε άλλο ένα ή δυο points εφόσον φυσικά πρόκειται περί πρωτοτύπου προγράμματος και όχι, ...περί κάποιου ανατύπου ή μετατροπής από τα ίδια περιοδικά που έχουμε βέβαια και εμείς στο αρχείο μας.

Όσοι έχουν προβλήματα με τη μνήμη του micro τους ασ κάνουν ορισμένες υποχωρήσεις έναντι των όρων εξηγώντας φυσικά τους λόγους.

# Κερδίστε ένα Jupiter Ace

## Αποστολή απαντήσεων

Οι απαντήσεις πρέπει να σταλούν στα γραφεία του περιοδικού μας στα γραφεία του περιοδικού μας εντός δυο μηνών και στη θέση του παραλήπτη να γράφουν:

ΠΕΡΙΟΔΙΚΟ PIXEL  
ΓΙΑ: PILEXMANIA ONE  
ΔΙΕΥΘΥΝΣΗ: ΒΕΡΑΝΖΕΡΟΥ 15,  
106 77 ΑΘΗΝΑ

Τα προγράμματά σας πρέπει να αρχίζουν με αρκετές REM statements και οι οποίες πρέπει να αναφέρουν εν σειρά τα εξής:

1 REM PILEXMANIA ONE - 2 REM ΑΡΙΘΜΟΣ ΤΩΝ POINTS ΠΟΥ ΔΙΕΚΔΙΚΕΙΤΕ - 3 REM ΟΝΟΜΑΤΕ- ΠΩΝΥΜΟ - 4 REM ΑΡΙΘΜΟΣ ΤΑΥΤΟΤΗΤΟΣ - 5 REM ΑΡΙΘΜΟΣ ΤΗΛΕΦΩΝΟΥ - 6 REM ΔΙΕΥΘΥΝΣΗ - 7 REM ΗΜΕΡΟΜΗΝΙΑ ΑΠΟΣΤΟΛΗΣ ΑΠΑΝΤΗΣΕΩΝ - 8 REM ΤΟ ΟΝΟΜΑ ΤΟΥ MICRO ΣΑΣ - REM LIST OF VARIABLES (σε όσα REM statements χρειάζεται).

Τά έπαθλα και η  
επιλογή τών τυχερών  
του PIXELMANIA  
ONE

Η επιλογή του τυχερού για το έπαθλο του διαγωνισμού PILEXMANIA ONE θα γίνεται βάσει των points-κλήρων που θα έχετε κερδίσει. Η συμμετοχή σας στην κλήρωση θα γίνεται βάσει αυτών των points-κλήρων. Τα έπαθλα έχουν ως εξής:  
ΠΡΩΤΟ ΒΡΑΒΕΙΟ: Ένας μικροϋπολογιστής JUPITER ACE, που προσφέρει η εταιρεία Παπασωτηρίου.

ΔΕΥΤΕΡΟ ΒΡΑΒΕΙΟ: Δέκα προγράμματα για τον LASER.

ΤΡΙΤΟ ΒΡΑΒΕΙΟ: Τρία χρόνια συνδρομή στο περιοδικό PIXEL.

## APPENDIX I / ΣΥΜΠΛΗΡΩΜΑ I

### Haiku φτιαγμένα από... «άνθρωπινους ποιητές»: Ernest Wood

Τα haiku είναι μια εξειδικευμένη μορφή Ιαπωνικής ποίησης σε τρεις στίχους, ο πρώτος με πέντε συλλαβές, ο δεύτερος με επτά, και ο τρίτος ξανά με πέντε συλλαβές. Η διάδοση των haiku στην Ιαπωνία είναι τόσο μεγάλη ώστε λέγεται ότι υπάρχουν τουλάχιστον 50 περιοδικά αφιερωμένα σ'αυτά. Τα haiku δεν γράφονται για διδακτικούς λόγους, ο απώτερος σκοπός είναι η βαθειά απόλαυση των συναίσθημάτων ή της γνώσης. Ορισμένα παραδείγματα έχουν ως εξής:

A morning glory.  
And so, today, may seem  
My own life story.

Δόξα πρωινή  
και έτσι, σήμερα, ίσως φανεί  
της ζωής μου η ιστορία.

The peony has fallen.  
A few scattered petals  
Lie on one another.

Έπεσε η παιωνία.  
Λίγα σκόρπια πέταλα  
Το ένα πάνω στ' άλλο

A butterfly  
Asleep, perched upon  
The temple bell.

Μια πεταλούδα  
Κοιμισμένη, πάνω  
Στου ναού την καμπάνα.

The thief  
Left it behind.  
The moon at the window.

Ο κλέφτης  
Το άφησε πίσω.  
Το φεγγάρι στο παραθύρι.

Αυτά τα παραδείγματα, όμως, δεν είναι εντελώς Ζενικά, καθώς περιέχουν πολύ σκεπτικό, που τείνει προς τη φιλοσοφία. Αυτό που ονομάζουμε Ζενικό haiku είναι εκείνο που έχει σαν αποτέλεσμα της υπερβατικής ενόρασης, έτσι ώστε, όπως το θέτει χαρακτηριστικά ο Blyth, "Το haiku δεν είναι ένα ποιήμα, δεν είναι λογοτεχνία, είναι το χέρι που κάνει νόημα: μια πόρτα μισάνοιχτη, ένας καθρέφτης που έχει τελείως σκουπιστεί. Είναι ο δρόμος επιστροφής στη φύση... (ακόμα και) στη φύση του πνεύματος". Τα παραδείγματα που ακολουθούν είναι γραμμένα από τον Basho Yesei ένα περίφημο συγγραφέα Ζενικών haiku:

The old pond.  
A frog jumps in.  
Plop.

Η παλιά λιμνούλα  
Ένας βάρταχος πηδά μέσα.  
Πλοπ.

On a withered boych  
A crow has stopped to perch  
And autumn derkens.

Σ' ένα ξερόκλαδο,  
Ένα κοράκι σταμάτησε για να κουνιάσει  
Και το φθινόπωρο σκοτινιάζει.

We gaze,  
Even at horses,  
This morn of snow.

Κοιτάμε  
Ακόμα και τάλογα,  
Αυτό το χιονισμένο πρωινό.

Λέγεται ότι για να συλλάβει κανείς αυτό που "κρύβει μέσα του" ή και για να νιώσει την επενέργεια ενός τέτοιου haiku, πρέπει να το επαναλάβει αρκετές φορές, ή και πάρα πολλές φορές, με ολοένα αυξανόμενη συγκέντρωση. Έτσι φτάνει κανείς στο επίπεδο της ενόρασης ή καλύτερα της διαίσθησης. Αυτό είναι συμφώνως με την αίσθηση του θαυμαστού.

Αυτή η αίσθηση του θαυμαστού, ή του ανοίγματος του πνεύματος, πέρα από τη σκέψη, ή αυτού που είναι στο πλατύ κοινό γνωστό σαν έκτη αίσθηση, (και όχι το sense-perception), προκαλείται τεχνητά, μέχρις ενός σημείου, από τη μορφή (σημειολογική υποδομή) του ποιήματος, έτσι ώστε να δημιουργεί μια παύση, ή και μια σειρά από παύσεις δεκτικότητας στη σκέψη. Ας υποθέσουμε ότι, αντί να κοιτάμε μέσα από ένα παράθυρο και να λέμε: "Βρέχει ξανά αυτό το πρωινό", κάποιος είπε:

It is raining,  
Again,  
This morning.

Βρέχει,  
Ξανά,  
Αυτό το πρωινό.

Αυτό δημιουργεί μια διαφορά, έτσι δεν είναι; Ο R.H. Blyth έχει γράψει τέσσερις τόμους σχετικά με τα haiku, γεμάτα από πλήθος πληροφοριών και οδηγιών προς τους haikoϊζομένους όπως επίσης και έναν τόμο που τιτλοφορείται: Zen in English Literature.

"Τάδε έφη" ο Ernest Wood στο βιβλίο του: Zen Dictionary

#### 20300 G POEM

All white in the hills  
I taste dark trees in the shade  
grrr the sun has fled

#### 20301 G POEM

All blue in the buds  
I seize white peaks in the cold  
flick the sun has flit

#### 20302 G POEM

All brown in the hills  
I flash red specks in the dusk  
swish the moon has cracked

Ένα δείγμα από τα αποτελέσματα που δίνει το πρόγραμμα του listing 1 και τα οποία μπορείτε να αντιπαραβάλετε με τα haiku που είναι φτιαγμένα από "άνθρωπινους ποιητές".

# PIXELMANIA ONE

## APPENDIX II / ΣΥΜΠΛΗΡΩΜΑ II

### Ποιήματα Haiku από ύπολογιστές: Douglas Hofstadter

Ο Douglas Hofstadter (Assistant Professor του Computer Science στο Indiana University) κάνει αρκετές αναφορές για τα Haiku και τα computer generated texts στο πολυσύμμαντο βιβλίο του: "GODEL, ESCHER,

BACH: A Metaphorical Fugue on Minds and Machines in the Spirit of Lewis Carroll. Το βιβλίο αυτό δεν απευθύνεται μόνο στους ειδικούς και εκτός από το ότι "είναι ένα από τα σημαντικότερα βιβλία που γράφτηκαν εδώ και αρκετές δεκαετίες"

(αυτό τα αναφέρει ο Martin Gardner) είναι στην ουσία ένα open university course με θέμα το Computer Science και επίκεντρο τα Μαθηματικά (Godel), τη σύγχρονη Τέχνη (Escher), και τη Μουσική (Bach).

### Διάλογος 'Αχιλλέα - Χελώνας για τὰ haiku.

Σε έναν από τους δεκάδες, αλληγορικά διδακτικούς διαλόγους, που παισιάζουν συχνά τα κεφάλαια του βιβλίου του και που έχουν κεντρικούς ήρωες τον Αχιλλέα και τη χελώνα (γνωστούς ήρωες του Ζήνωνα... και αρκετά αργότερα του Lewis Carroll) τα haiku πρωτοπαρουσιάζονται ως εξής: (Ο Αχιλλέας και η Χελώνα! μόλις έχουν αποτελειώσει ένα γευστικότατο κινέζικο γεύμα για δυο άτομα, σε ένα από τα καλύτερα κινέζικα ρεστωράν της περιοχής...).

Αχιλλέας: Μπορεί να ξέρεις περισσότε-

ρα για την κινεζική κουζίνα απ'ότι εγώ, κυρία Χελώνα, αλλά βάζω στοίχημα ότι ξέρω περισσότερα για την γιαπωνέζικη ποίηση απ'όσο εσύ. Έχεις ποτέ διαβάσει κανένα haiku;

Χελώνα: Πολύ φοβάμαι πως όχι. Τι είναι το haiku;

Αχιλλέας: Το haiku είναι ένα γιαπωνέζικο δεκαεπτασύλλαβο ποίημα ή μινιποίημα, μάλλον, μια μορφή επίκλησης με τον ίδιο, ίσως, τρόπο που είναι και ένα μυ-

ρωδάτο τριανταφυλλένιο πέταλο, ή μια λιμνούλα με κρίνους όταν ψιχαλίζει. Γενικά αποτελείται από ομάδες, πέντε, μετὰ επτά και τέλος πέντε συλλαβών.

Χελώνα: Τέτοια συμπιεσμένα ποιήματα με δεκαεπτά συλλαβές δεν είναι ποτέ δυνατόν να περιέχουν πολύ νόημα...

Αχιλλέας: Το νόημα βρίσκεται τόσο στο νου του αναγνώστη όσο και στο haiku.

Χελώνα: Χμμ... Αυτό είναι μια δήλωση - επίκληση...

### 'Από τὰ Computer Haiku σέ μία Recursive Transition Network Grammar

Η πολυπλοκότητα της έκφρασης της γνώσης, στον εγκέφαλο του ανθρώπου, μου έκανε για πρώτη φορά εντύπωση όταν δούλεψα πάνω σε ένα πρόγραμμα που δημιουργούσε αγγλικές προτάσεις "στα ξαφνικά". Το project αυτό μου "ήρθε" κατά ένα αρκετά ενδιαφέροντα τρόπο. Είχα ακούσει στο ραδιόφωνο ορισμένα από αυτά τα περίφημα "Computer Haiku". Κάτι, σχετικά με αυτά με επηρέασε πολύ βαθειά. Υπήρχαν πολλά στοιχεία χιούμορ και ταυτόχρονα μυστηρίου στο να κάνει ένα computer να δημιουργήσει κάτι που συνήθως θα είχε θεωρηθεί καλλιτεχνική δημιουργία. Διασκέδαζε πάρα πολύ με... τη χιουμοριστική πλευρά του θέματος και το "μυστήριο" μου είχε εξάψει υπερβολικά το ενδιαφέρον - ακόμα και της αντίφασης - σχετικά με τη χρήση του προγραμματισμού στις Καλές Τέχνες. Έτσι άρχισα να γράφω ένα πρόγραμμα πολύ περισσότερο αντιφατικό, μυστηριώδες και χιουμοριστικό από το πρόγραμμα για haiku.

Αρχικά απέβλεπα στη δημιουργία μιας εύπλαστης και "προοδευτικής" γραμματικής, έτσι ώστε να μην έχει κανείς την αίσθηση ότι το πρόγραμμα γέμιζε απλώς και μόνον τα κενά με ένα κάποιο πρότυπο. Εκείνη ακριβώς την εποχή έπεσε στα χέρια μου ένα άρθρο του Victor Yngve στο Scientific American που περιέγραφε μια απλή αλλά εύπλαστη γραμματική που μπορούσε να δημιουργεί μια μεγάλη ποικιλία από προτάσεις, σαν και αυτές που βρίσκει κανείς σε ορισμένα παιδικά βιβλία. Τροποποίησα μερικές από τις ιδέες που είχα συγκεντρώσει από το άρθρο και κατέληξα σε μια σειρά από κανόνες που μορφοποίησαν μια ως πούμε "προοδευτική" γραμματική. Σε αυτή τη γραμματική, η επιλογή των λέξεων μιας πρότασης καθορίζονταν από μια διαδικασία που άρχιζε με το διάλεγμα - στην τύχη - της ολικής δομής της πρότασης. Βαθμιαία η διαδικασία λήψης αποφάσεων περνούσε σιγά-σιγά δια μέσου των κατω-

τέρων επιπέδων της δομής αυτής μέχρι που έφθανε στο επίπεδο των λέξεων και στο επίπεδο των γραμμάτων. Πολλά έπρεπε να γίνουν πέρα από το επίπεδο των λέξεων, όπως οι κλίσεις των ρημάτων και ο πληθυντικός των ουσιαστικών έπρεπε επίσης να δημιουργούνται, στην αρχή κανονικά, οι διάφορες ανώμαλες μορφές ρημάτων και ουσιαστικών, και μετὰ αν ταίριαζαν με τα entries ενός πίνακα, τότε γινόταν μια αντικατάσταση με τις σωστές (ανώμαλες) μορφές. Κάθε λέξη υπώνυταν μόλις παρνούσε από αυτή την τελική μορφοποίηση. Το πρόγραμμα αυτό ήταν σαν την παροιμιώδη μαιμού μπροστά στη γραφομηχανή, αλλά λειτουργούσε ταυτόχρονα σε πολλά επίπεδα γλωσσολογικής δομής - και όχι μόνο στο επίπεδο των γραμμάτων.

Στις πρώτες φάσεις της ανάπτυξης αυτού του προγράμματος, χρησιμοποίησα ένα τελείως "χαζοχαρούμενο" λεξιλόγιο - επίτηδες, μια που απέβλεπα στη δημιουργία χιούμορ. Το πρόγραμμα συνέθεσε πάρα πολλές παράλογες προτάσεις, πολλές από τις οποίες είχαν πάρα πολύ συγκεχυμένη δομή, ενώ άλλες ήταν πολύ μικρές. Ορισμένα από τα σταχουολογήματα έχουν ως εξής: (εικόνα 2)

Το εμφάνει είναι έντονα σουρρεαλιστικό και ορισμένες φορές θυμίζει haiku - παραδείγματος χάριν, το τελευταίο παράδειγμα με τις τέσσερις διαδοχικές μικρές προτάσεις. Στην αρχή όλα φαίνονταν

### EIKONA 2

Τό άρσενικό μολύβι πού πρέπει νά γελάει άγαρμπα θά έκανε κουά-κουά. Πρέπει πρόγραμμα όχι πάντα νά μασουλάει κορίτσι στή μνήμη; Τό δεκαδικό bug πού φτύνει άγαρμπα ίσως κάνει τούμπες. Τό κέικ πού σίγουρα παίρνει έναν άπρόβλεπτο άντρα εντός σχέσεως μπορεί πάντα νά ρίχνει τραπουλόχαρτο.

Τό πρόγραμμα πρέπει νά τρέχει χαρούμενο.

Τό game κάνει κουά-κουά. 'Ο καθηγητής θά γράψει πίκλα. Τό bug κάνει τούμπες. 'Ο άντρας παίρνει τό κουτί πού γλυστρά.

1 Για διάφορους δικούς του λόγους, ο Hofstadter θεωρεί τη χελώνα άρσενικού γένους και την ονομάζει: Mr. Tortoise αφού πρώτα το δικαιολογήσει απολύτως.



# ΚΑΛΩΣ ΗΛΘΑΤΕ ΣΤΟΝ ΚΟΣΜΟ ΤΩΝ ΚΟΜΠΙΟΥΤΕΡ

Ἡ σελίδα αὐτή ἀποτελεῖ ἓνα πολὺ περιληπτικὸ ὁδηγὸ μὲ σκοπὸ νὰ σᾶς φέρει σὲ μιὰ πρώτη γνωριμία μὲ τὸ περίεργο (ἀλλὰ δυστυχῶς ἀπαραίτητο) λεξιλόγιό πού συνοδεύει τοὺς κομπιούτερ. Ὅπως θὰ διαπιστώσετε πολὺ σύντομα, οἱ βασικὲς ἔννοιες δὲν εἶναι ἰδιαίτερα πολὺπλοκες καὶ θὰ μπορέσετε εὐκόλα νὰ τῆς ξεπεράσετε. Ἡ ἴδια ἀκριβῶς σελίδα θὰ δημοσιεύεται σὲ κάθε τεύχος τοῦ περιοδικοῦ μὲ σκοπὸ ν' ἀποτελεῖ ἓνα ἄμεσο βοήθημα γιὰ τοὺς καινούργιους κάθε φορὰ φίλους τοῦ κομπιούτερ. Ἐλπίζουμε ὅτι θὰ βοηθήσει καὶ ἐσᾶς νὰ ξεπεράσετε τοὺς πρώτους φόβους καὶ δισταγμοὺς πού συχνά συνοδεύουν τοὺς νεοφώτιστους στὸ χῶρο αὐτό, καὶ σᾶς εὐχόμεσθε "καλῶς ἤλθατε στὸν κόσμον τῶν κομπιούτερ".



Θὰ ξεκινήσουμε μελετώντας τὴ λειτουργία τοῦ μικροκομπιούτερ καὶ στὴ συνέχεια θὰ ἐξετάσουμε τὰ ἐπιμέρους τμήματα πού ἀποτελοῦν τὸ μὴχάνημα.

Ἐνας μικροκομπιούτερ δέχεται ἀπὸ τὸ χειριστὴ του πληροφορίες, τῆς **ΕΠΕΞΕΡΓΑΖΕΤΑΙ** καὶ στὴ συνέχεια ἀποθηκεύει τ' ἀποτελέσματα τῆς ἐπεξεργασίας ἢ τὰ παρουσιάζει στὸ χειριστὴ. Οἱ πληροφορίες αὐτὲς ὀνομάζονται διεθνῶς **"DATA"** καὶ συμπεριλαμβάνουν ἀριθμοὺς, γράμματα, καὶ εἰδικὰ σύμβολα πού μποροῦν νὰ διαβαστοῦν ἀπὸ τὸν ἄνθρωπο. Παρὰ τὸ γεγονὸς ὅτι ὁ κομπιούτερ δέχεται τὰ στοιχεῖα αὐτὰ (καὶ παρουσιάζει τ' ἀποτελέσματα στὸ χειριστὴ του) σὲ κάποια "ἄνθρωπινη" γλῶσσα, ἡ ἐπεξεργασία καὶ ἡ ἀποθήκευσή τους ἀπὸ τὸ μὴχάνημα γίνεται ἀφοῦ μεταφραστοῦν σ' ἓνα εἶδος ἠλεκτρονικοῦ κώδικα πού ὀνομάζεται **ΔΥΑΔΙΚΟΣ**, ἐπειδὴ ἀποτελεῖται ἀπὸ τοὺς συνδυασμοὺς δύο μόνο στοιχείων, τοῦ μηδέν καὶ τῆς μονάδας. Ἐτσι, στοὺς περισσώτερους κομπιούτερ, ὁποιοσδήποτε χαρακτήρας ἢ σύμβολο μπορεῖ νὰ παρασταθεῖ ἀπὸ 8 "δυσδικὰ ψηφία" (BITS) τὸ σύνολο τῶν ὁποίων ἀποτελεῖ ἓνα **BYTE**. Ἐνα τέτοιο

byte μπορεῖ συνεπῶς νὰ παίρνει τιμὲς πού ξεκινοῦν ἀπὸ 00000000 καὶ φτάνουν σὲ 11111111 (δημιουργώντας ἔτσι ἓνα σύνολο 256 διαφορετικῶν συνδυασμῶν). Γιὰ νὰ μὴ ὑπάρχει σύγχυση (ἐπειδὴ καθένας θὰ μπορούσε ν' ἀποδίδει σὲ κάθε συνδυασμὸ ὁποιοδήποτε γράμμα, ἀριθμὸ ἢ σύμβολο ἤθελε), ὑπάρχουν σήμερα ὀρισμένα ἀναγνωρισμένα συστήματα κωδικοποίησης ὅπως π.χ. τὸ ASCII (American Standard Code for Information Interchange). Σὰν ἓνα παράδειγμα τοῦ κώδικα αὐτοῦ ἀναφέρουμε ὅτι ὁ "ἄνθρωπινος" ἀριθμὸς 5 ἀντιπροσωπεύεται στὸ ASCII ἀπὸ τὸ byte 00110101-ἓνας συνδυασμὸς ἐξαιρετικὰ πολὺπλοκος γιὰ τὸν ἄνθρωπο ἀλλὰ πολὺ ἀπλὸς γιὰ τὸν κομπιούτερ (πού εἶναι σὲ θέση νὰ ἀναγνωρίσει τῆς διαφορὲς μεταξύ τοῦ μηδέν καὶ τῆς μονάδας σὰν διαφορὲς ἠλεκτρικοῦ δυναμικοῦ).

Ὁ κομπιούτερ ἐπεξεργάζεται τὰ στοιχεῖα πού τοῦ παρέχει ὁ χειριστὴς ἐκτελώντας διάφορες ἀριθμητικὲς πράξεις ἢ συγκρίνοντάς τα μὲ ἄλλες πληροφορίες. Ἡ τελευταία αὐτὴ λειτουργία τοῦ ἐπιτρέπει νὰ "παίρνει ἀποφάσεις" καὶ νὰ δίνει ἔτσι τὴν ἐντύπωση ἑνὸς εὐφυοῦς μηχανή-

ματος. Γιὰ νὰ πραγματοποιήσει ὅμως τὴ λειτουργία αὐτὴ, ὁ κομπιούτερ πρέπει νὰ ἔχει τροφοδοτηθεῖ μ' ἓνα "σύνολο ὁδηγιῶν" πού νὰ τοῦ ὑποδεικνύουν τί θὰ κάνει. Ἡ λίστα αὐτὴ τῶν ὁδηγιῶν (πού μέσα στὸ μὴχάνημα ἀποθηκεύονται στὴ μνήμη μὲ τὴ μορφή πολλῶν byte) ὀνομάζεται **ΠΡΟΓΡΑΜΜΑ** καὶ μπορεῖ νὰ δοθεῖ στὸν κομπιούτερ εἴτε στὴ "μητρικὴ του γλῶσσα" (**ΚΩΔΙΚΑΣ ΜΗΧΑΝΗΣ**), εἴτε σὲ μιὰ πιὸ "ἄνθρωπινη" μορφή, πού θὰ πρέπει στὴ συνέχεια ὁ κομπιούτερ νὰ μεταφράσει στὸ δυαδικὸ σύστημα. Ὅποιαδήποτε τέτοια γλῶσσα πού βρίσκεται πιὸ κοντὰ στὸν ἄνθρωπο παρὰ στὴ μηχανὴ ὀνομάζεται **"ΥΨΗΛΟΥ ΕΠΙΠΕΔΟΥ"** καὶ ἀπαιτεῖ λιγότερη προσπάθεια ἀπὸ τὸν προγραμματιστὴ. (Ἀπὸ τὴν ἄλλη μεριά ὅμως, ἐπειδὴ ὁ κομπιούτερ θὰ πρέπει νὰ μεταφράζει τὰ στοιχεῖα αὐτὰ στὸ δυαδικὸ σύστημα, ἓνα πρόγραμμα γραμμμένο σὲ ὑψηλοῦ ἐπιπέδου γλῶσσα εἶναι πάντα πιὸ ἀργὸ ἀπ' ὅσο ἓνα γραμμμένο σὲ κώδικα μηχανῆς).

Σήμερα ὑπάρχουν πολλὲς γλῶσσες ὑψηλοῦ ἐπιπέδου ὅπως ἡ FORTRAN, ἡ COBOL, ἡ PASCAL, ἡ BASIC, ἡ PILOT, ἡ FORTH καὶ ἄλλες, καθεμιὰ ἀπὸ τῆς ὁποῖες πα-

ρουσιάζει ορισμένα ιδιαίτερα χαρακτηριστικά. Μιά από τις δημοφιλέστερες γλώσσες στους μικροκομπιούτερ είναι η BASIC, μία αρκετά απλή γλώσσα που επιτρέπει στον καθένα να μάθει προγραμματισμό εύκολα και γρήγορα. Οι εντολές του προγράμματος δίνονται από το πληκτρολόγιο και στη συνέχεια καταχωρούνται στη μνήμη του κομπιούτερ. Όταν πάρει την εντολή να εκτελέσει το πρόγραμμα, το μηχάνημα χρησιμοποιεί ένα "μεταφραστή" που μεταφράζει τη γλώσσα ύψηλου επιπέδου σε κώδικα μηχανής, και μεταβιβάζει το πρόγραμμα στον επεξεργαστή για τή περαιτέρω.

Περνώντας τώρα στο σύνολο των διαφόρων τμημάτων που απαρτίζουν την ύλική υπόσταση του κομπιούτερ και που φέρονται κάτω από τη γενική ονομασία "HARDWARE" (σε αντίθεση με τή διάφορα προγράμματα που ονομάζονται "SOFTWARE"), βλέπουμε ότι η καρδιά του μικροκομπιούτερ είναι ή ΚΕΝΤΡΙΚΗ ΜΟΝΑΔΑ ΕΠΕΞΕΡΓΑΣΙΑΣ (Central Processing Unit - CPU). Η CPU αποτελείται από ένα μικροεπεξεργαστή που είναι υπεύθυνος για όλους τούς υπολογισμούς που πραγματοποιεί ή κομπιούτερ και που χρειάζεται μια μνήμη πάνω στην οποία να καταχωρούνται τή διάφορα στοιχεία καθώς και τή πρόγραμμα του κομπιούτερ. Γενικά οι μικροκομπιούτερ διαθέτουν δύο ειδών μνήμη, RAM (Random Access Memory - Μνήμη Τυχαίας Προσπέλασης) και ROM (Read Only Memory). Η RAM αποτελεί ένα σημειωματάριο πάνω στο οποίο ή CPU μπορεί να γράφει και να σβύνει ανάλογα με τή ανάγκη του κομπιούτερ. Έπειδή όμως τή περιεχόμενα τής μνήμης RAM χάνονται κάθε φορά που ή κομπιούτερ παύει να τροφοδοτείται με ρεύμα, είναι απαραίτητη ή ύπαρξη μιας μόνιμης μνήμης πάνω στην οποία ή κατασκευαστής του κομπιούτερ καταχωρεί μια σειρά στοιχείων απαραίτητων για τή λειτουργία του μηχανήματος (όπως π.χ. τή "μεταφραστή" από γλώσσα ύψηλου επιπέδου σε δυαδικό κώδικα). Έπειδή ή CPU μπορεί μόνο να διαβάσει τή περιεχόμενα τής ROM χωρίς να μπορεί να τή μεταβάλλει, γι' αυτό ή τελευταία παρομοιάζεται (σε αντίδιαστολή με τή RAM) μ' ένα βιβλίο. Υπάρχουν σήμερα στο εμπόριο ειδικές προγραμμα-

τιζόμενες μνήμες ROM (PROM) καθώς και EPROM, PROM δηλαδή που μπορούν να "σβυστούν" και να ξαναγραφούν. Τέτοια πρόσητα εξαρτήματα αυξάνουν σημαντικά τή δυνατότητες ενός κομπιούτερ, αλλά στοιχίζουν αρκετά.

Έπειδή όπως είπαμε ή μνήμη RAM χάνει τή στοιχεία που έχουν καταγραφεί σ' αυτήν κάθε φορά που βγάζουμε τόν κομπιούτερ από τήν πρίζα. (μέ αποτέλεσμα να "έξαφανίζονται" προγράμματα και πληροφορίες που μπορεί να χρειάστηκαν πολλές ώρες δουλειάς για να επινοηθούν και να "περαστούν" στον κομπιούτερ), γι' αυτόν άκριβώς τή λόγο δημιουργήθηκε ή ανάγκη για κάποιο μηχανισμό που θά μπορούσε να συγκρατεί τέτοια στοιχεία σε μια πιο μόνιμη μορφή. Σήμερα τή λειτουργία αυτή τήν έχουν αναλάβει οι κοινές ΚΑΣΕΤΕΣ ΜΑΓΝΗΤΟΦΩΝΟΥ, οι εύκαμπτοι δίσκοι (ΔΙΣΚΕΤΕΣ) και οι "ΣΚΛΗΡΟΙ" ΔΙΣΚΟΙ. Έτσι ένα συνηθισμένο κασετόφωνο μπορεί να χρησιμοποιηθεί για να μετατρέψει μια σειρά πληροφοριών σε ήχητικά σήματα που καταγράφονται στην κασέτα και που αργότερα μπορούν με μια αντίστροφη διαδικασία να ξαναπεραστούν στον κομπιούτερ. Τή σύστημα τής αποθήκευσης στοιχείων σε κασέτα, αν και εξαιρετικά οικονομικό, παρουσιάζει μια σειρά μειονεκτημάτων όπως π.χ. τή μεγάλη καθυστέρηση που παρατηρείται όταν θέλουμε να "φορτώσουμε" κάποιο συγκεκριμένο στοιχείο από τήν κασέτα στον κομπιούτερ. Έτσι σήμερα τή πιο δημοφιλές σύστημα που χρησιμοποιούν οι μικροκομπιούτερ για τήν αποθήκευση στοιχείων είναι ή δισκέτα, που επιτρέπει πολύ γρήγορα "προσέλαση" όπουδήποτε και αν βρίσκονται καταχωρημένα τή στοιχεία που ψάχνουμε. Όπως όμως ή κασέτα χρειάζεται ένα κασετόφωνο για να λειτουργήσει, έτσι και ή δισκέτα απαιτεί ένα ειδικό μηχανισμό που ονομάζεται "disk drive" και που σήμερα κοστίζει ένα σημαντικό ποσό χρημάτων. Ακόμα ακριβότερο από τή σύστημα τής δισκέτας είναι τή σύστημα του λεγόμενου "σκληρού" ή άκαμπτου δίσκου, ή χρήση του οποίου περιορίζεται για τήν ώρα στους μεγάλους κομπιούτερ.

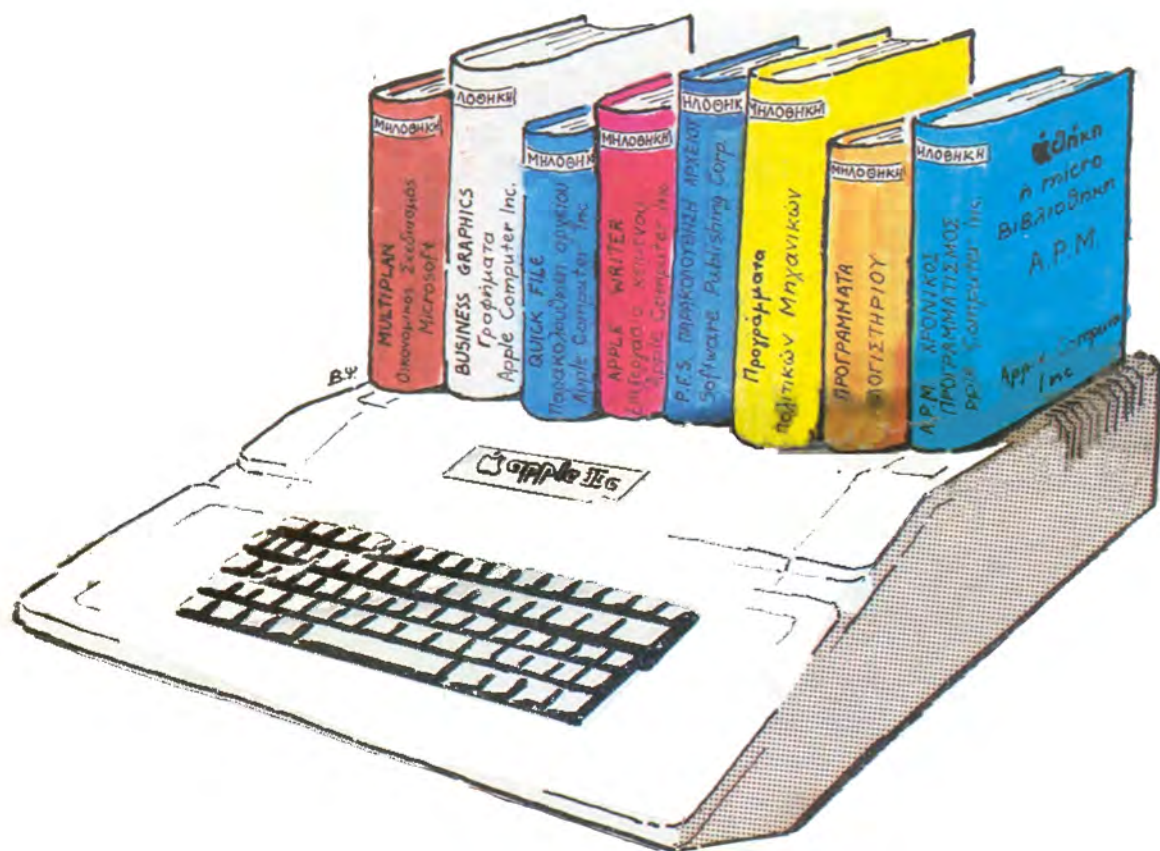
Ό εκάστοτε χειριστής ενός κομπιούτερ πρέπει να έχει φυσι-

κά τή δυνατότητα επικοινωνίας με τή μηχανήμα. Αυτό επιτυγχάνεται μέσω δύο εξαρτημάτων, του ΠΛΗΚΤΡΟΛΟΓΙΟΥ που χρησιμεύει για να "μιλάμε" έμεις στον κομπιούτερ, και τής «ΜΟΝΑΔΑΣ ΟΠΤΙΚΗΣ ΑΠΕΙΚΟΝΙΣΗΣ» (Visual Display Unit-VDU) που είναι συνήθως ή όθόνη μιας τηλεόρασης ή ενός μόνιτορ και που χρησιμεύει για να "μιλάει" ή κομπιούτερ σ' έμας. Εκτός όμως από τή VDU, ή κομπιούτερ έχει τή δυνατότητα να καταγράψει, μέσω ενός ΕΚΤΥΠΩΤΗ, διάφορα στοιχεία σε χαρτί, δίνοντας έτσι στο χειριστή του ένα "HARD COPY" των πληροφοριών αυτών. Έδώ θά πρέπει ν' αναφέρουμε ότι τή σύνολο των εντολών και πληροφοριών που δέχεται ή κομπιούτερ μέσω του πληκτρολόγιου ονομάζεται διεθνώς "INPUT", ενώ τή αποτέλεσμα τής επεξεργασίας που πραγματοποιεί τή μηχανήμα στα στοιχεία αυτά και που προσφέρει στη συνέχεια στο χειριστή ονομάζεται "OUTPUT". Ο κομπιούτερ μπορεί να δέχεται και να μεταβιβάζει πληροφορίες στον εκτυπωτή ή σε κάποιο άλλο περιφερειακό με δύο τρόπους: "έν σειρά" και "έν παράλληλα". Οι διαφορετικοί αυτοί μέθοδοι επικοινωνίας απαιτούν και διαφορετικού είδους "INTERFACE" (εξαρτήματα που εξασφαλίζουν τή σύνδεση μεταξύ του κομπιούτερ και του περιφερειακού). Έτσι για επικοινωνία "έν σειρά" χρησιμοποιείται συνήθως τή RS232 interface ενώ για επικοινωνία "έν παράλληλα" ιδιαίτερα δημοφιλές είναι τή interface τύπου "Centronics".

Προτού κλείσουμε τή σύντομη αυτή εισαγωγή στην όρολογία των μικροκομπιούτερ, θά πρέπει ίσως ν' αναφερθούμε και στην έννοια του "MODEM". Ένα modem (modulator-demodulator) προσφέρει στον κομπιούτερ τή δυνατότητα να...επικοινωνήσει μέσω του τηλεφώνου μ' ένα άλλο κομπιούτερ που διαθέτει επίσης modem και ν' ανταλλάξει μαζί του πληροφορίες. Έπειδή τή modem πρέπει να συνδεθεί με τή τηλεφωνικό σύστημα, τή τελευταία χρόνια έχουν εμφανισθεί διάφορες τροποποιημένες μορφές (ACOUSTIC COUPLER) που επιτρέπουν να γίνεται μια τέτοια "τηλεφωνική συνδιάλεξη" χωρίς ν' απαιτείται άμεση ηλεκτρική σύνδεση με τή σύστημα του τηλεφώνου.

# 🍏 αθήκη

## η μικτο Βιβλιοθήκη σας



 **apple computer**

Προς την  
**RAINBOW COMPUTER APPLICATIONS**  
Έλ. Βενιζέλου 184, Καλλιθέα

ΝΑΙ! Θά ήθελα, χωρίς καμιά υποχρέωση μου,  
περισσότερες πληροφορίες για τον Apple IIe.

ΟΝΟΜΑ \_\_\_\_\_  
Δ/ΝΣΗ \_\_\_\_\_  
ΤΗΛ \_\_\_\_\_

Αποκλειστικοί αντιπρόσωποι για την Ελλάδα  
**RAINBOW COMPUTER APPLICATIONS**  
Έλ. Βενιζέλου 184 - Καλλιθέα Τηλ. 9594082 TELEX: 221793 RAIN

# NewBrain

## τώρα πρώτος στον κόσμο μιλάμε απ' ευθείας Ελληνικά

- Ιδανικό για Επιτόρους, Μηχανικούς, Αρχιτέκτονες, Πατρούς, Δασκάλους, Δικηγόρους, Τεχνικούς, Φραστικούς κ.λ.π.
- Διαθέτει πλήρες πληκτρολόγιο γραφομηχανής με πεζά και κεφαλαία Ελληνικά και Αγγλικά γράμματα, σύμβολα και γραμμικός χαρακτήρας
- Έχει 5ηχη τον εθόνη 16 χαρακτήρων ή συνοδεύεται με MONITOR ή τηλεόραση όπου εμφανίζει 24 ή 30 γραμμές, από 40 ή 80 χαρακτήρων
- Διαθέτει μνήμη 32 K RAM και 256 K ROM
- Βασίζεται στην μικροεπεξεργαστή Z8018
- Έχει 2 εξόδους για κασετοφώνη, έξοδο για κασετοταινία, κασετοεισοδο και έξοδο/είσοδο για κωδικοποιητή ή κωδικοποιητή
- Η μνήμη του επεκτείνεται σε βήματα των 64 K, 128 K, 256 K, 512 K
- 2MB RAM
- Δέχεται επέκταση της ROM με προγράμματα σε EPROM
- Δέχεται επέκταση διαύλων 5 1/4 ή σκληρό δίσκο Winchester με λειτουργία απευθείας CP/M
- Το μόντελο MICRO με μνήμη 128 Kbytes



**GRUNDY ELECTRONICS INTERNATIONAL S.A.**

Λεωφ. Μεσογείων 274  
Χολαργός - Τηλ. 0525 27715 - Π.Υ. Χολαργών GRUNDY S.A.

a member of the Grundy Group of Companies

**NewBrain**

**ΚΕΝΤΡΑ ΠΩΛΗΣΕΩΣ:**

**BYTE SHOP COMPUTER**  
Πινδάρου 25  
τηλ.: 3631361  
106 73 ΑΘΗΝΑ

**ΧΡ. ΕΥΑΓΓΕΛΙΔΗΣ**  
Εγγατίας 65  
τηλ.: 031-270054  
546 31 ΘΕΣΣΑΛΟΝΙΚΗ

**CAT COMPUTERS**  
Ιπποκράτους 57  
τηλ.: 3643044  
106 80 ΑΘΗΝΑ

**ACC (ATHENS COMPUTER CENTER)**  
Σολωμού 26  
τηλ.: 3609217  
106 82 ΑΘΗΝΑ