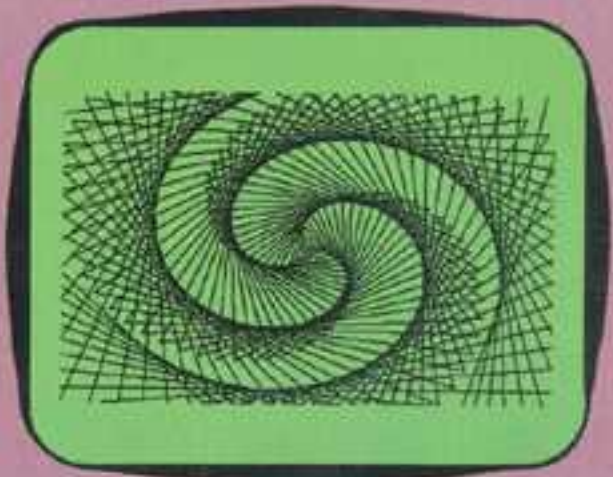
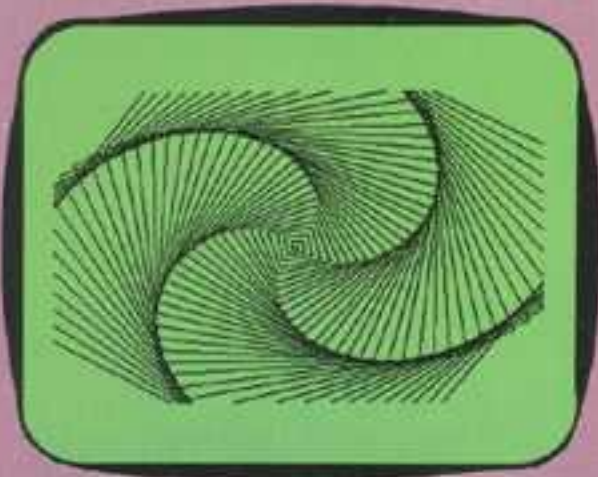
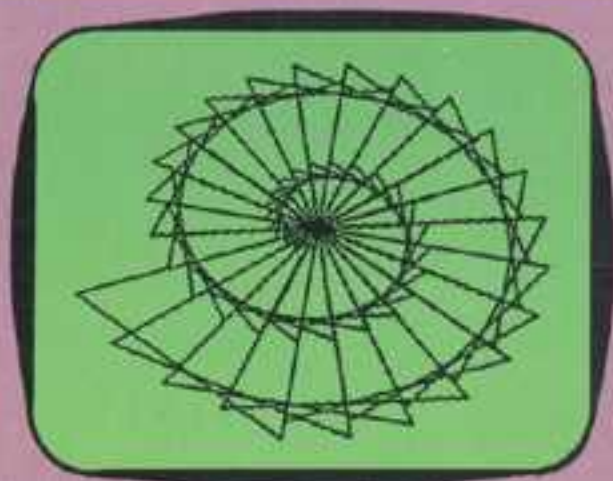


ΜΑΘΕΤΕ ΤΗΝ **LOGO**

στον **IBM PC** και τους συμβατούς

ΠΕΡΙΕΧΕΙ ΡΟΥΤΙΝΕΣ ΓΙΑ ΧΡΩΜΑ, ΓΡΑΦΙΚΑ ΚΑΙ ΗΧΟ



Dietrich Senfleben

Περιεχόμενα

Πρόλογος	7
1 Το μικρό αλφάβητο της Logo με γραφικά	9
Μάθημα 1: Καλώς ήλθατε στην IBM-Logo	9
Μάθημα 2: Γραμμές προς κάθε κατεύθυνση	17
Μάθημα 3: Η REPEAT επαναλαμβάνει τα πάντα	30
Μάθημα 4: Τα λάθη πληκτρολόγησης δεν μας ενοχλούν	39
Μάθημα 5: Σχήματα στην οθόνη	43
Μάθημα 6: Έτσι φτιάχνονται τα προγράμματα	61
Μάθημα 7: Η διόρθωση των προγραμμάτων	71
Μάθημα 8: Πώς αποθηκεύουμε ένα πρόγραμμα, πώς το φορτώνου- με, και πώς παίρνουμε μια λίστα του.	78
Μάθημα 9: Προγράμματα καλούν προγράμματα	82
Μάθημα 10: Μεγάλο, μεγαλύτερο, μεταβλητό	87
Μάθημα 11: Το χρώμα είναι πλεονέκτημα	97
Μάθημα 12: Ιδέες και πειραματισμοί	103
2 Καλημέρα κύριε Γουτεμβέργι —ξέρουμε κι εμείς να τυπώνουμε ..	121
2.1 Λέξεις και προτάσεις	121
2.2 Εικόνες από γράμματα και κείμενα	122
2.3 Διαχείριση διευθύνσεων	127
2.4 Θα τυπώσουμε σε χαρτί	128
2.5 Ασκήσεις	130
3. Ο σπουδαιότερος προμηθευτής μας είναι η OUTPUT	133
3.1 Τι μας δίνει λοιπόν η OUTPUT;	133
3.2 Λαξεύοντας νέα εργαλεία	135
3.3 Η Logo καταλαβαίνει ελληνικά	139

6 Περιεχόμενα

3.4 Μεγάλες ιστορίες	141
3.5 Ασκήσεις	147
4 Μία φορά ίσον καμία φορά - Βρόχοι στα προγράμματα	151
4.1 Τι συμβαίνει, αν...;	151
4.2 Τα φρένα	154
4.3 Η μεγάλη στροφή	157
4.4 Ασκήσεις	162
5. Σε ονομάζω	167
5.1 Κάθε τι έχει ένα όνομα	167
5.2 Επαναληπτικοί χειρισμοί	169
5.3 Ασκήσεις	172
6. Έχεις και ήχο	177
6.1 Η εντολή TONE	177
6.2 Θα πεις κι ένα τραγούδι	179
6.3 Εδώ χρειάζεται ένα πιάνο	184
6.4 Ασκήσεις	187
Παράρτημα	191
Σύντομη γραμματική της Logo	191
Χρήσιμα υποπρογράμματα	197
Φορμάρισμα και αντιγραφή δισκετών	199
Επισκόπηση των όρων της IBM-Logo	201
Επισκόπηση των πλήκτρων ελέγχου και των λειτουργιών διόρθωσης	213
Ευρετήριο	215

4

Μια Φορά ίσον Καμιά Φορά — Βρόχοι στα Προγράμματα

4.1 Τι Συμβαίνει, αν...;

Στις προηγούμενες παραγράφους, όλες οι γραμμές εντολών εκτελούνταν όμορφα, γραμμή-γραμμή. Όμως, συχνά υπάρχουν συνθήκες από τις οποίες είναι εξαρτημένα κι άλλα πράγματα. Με το σχηματισμό μερικών προτάσεων που περιέχουν το *Αν*, ξεκαθαρίζουν τα πάντα.

- Αν* δεν έχουν πεθάνει,
 τότε σήμερα ζουν ακόμη.
- Αν* ο έλεγχος έχει πεντάρια,
 τότε θα έχουμε γκρίνιες,
 αλλιώς όχι.
- Αν* έρθει η φιλενάδα μου,
 τότε θα χαρώ,
 αλλιώς θα βρω καινούργια.
- Αν* το γράμμα ζυγίζει μέχρι 20 γραμμάρια,
 τότε στοιχίζει το κανονικό τέλος,
 αλλιώς στοιχίζει περισσότερο.

Αντί για την λέξη «αν» θα μπορούσε να επιλεγεί η λέξη «εφόσον» και αντί του «αλλιώς» η «σε αντίθετη περίπτωση». Στ' αγγλικά οι λέξεις που χρησιμοποιούνται είναι :

- IF* (συνθήκη)
 THEN εντολές για την αληθή περίπτωση (διακλάδωση ναι)
 ELSE εντολές για την ψευδή περίπτωση (διακλάδωση όχι)

Αν έχει προβλεφθεί μόνο η αληθής περίπτωση, τότε μιλάμε για μια μονόπλευρη διακλάδωση του προγράμματος. Στην άλλη περίπτωση, γίνεται λόγος για διπλή διακλάδωση του προγράμματος.

Η IBM-Logo χρησιμοποιεί μόνο τον όρο IF. Τα THEN και ELSE είναι περιττά. Μετά τη συνθήκη ακολουθεί μια λίστα εντολών για την αληθή περίπτωση ή μια δεύτερη λίστα για την ψευδή περίπτωση.

Το επομενο απλό παράδειγμα δείχνει μια διπλή διακλάδωση. Ανάλογα με την ώρα της ημέρας θα στέλνεται ο χαιρετισμός «Καλημέρα» ή «Καληνύχτα».

```

TO ΧΑΙΡΕΤΙΣΜΟΣ :ΩΡΑ
IF :ΩΡΑ > 18 [PR [ΚΑΛΗΝΥΧΤΑ]] [PR [ΚΑΛΗΜΕΡΑ]]
END

```

```

?ΧΑΙΡΕΤΙΣΜΟΣ 19.10
ΚΑΛΗΝΥΧΤΑ
?ΧΑΙΡΕΤΙΣΜΟΣ 12
ΚΑΛΗΜΕΡΑ

```

Το ίδιο πρόβλημα θα μπορούσε να λυθεί χρησιμοποιώντας μια μονάχα διακλάδωση.

```

TO ΧΑΙΡΕ :ΩΡΑ
IF :ΩΡΑ > 18 [PR [ΚΑΛΗΝΥΧΤΑ]]
IF :ΩΡΑ < 18 [PR [ΚΑΛΗΜΕΡΑ]]
END

```

```

?ΧΑΙΡΕ 19.10
ΚΑΛΗΝΥΧΤΑ
?ΧΑΙΡΕ 12
ΚΑΛΗΜΕΡΑ

```

Από τις διακλαδώσεις ενός προγράμματος μπορούν να κληθούν άλλα προγράμματα. Ας πάρουμε την περίπτωση όπου ανάλογα με τον τύπο των δεδομένων (λίστα ή λέξη), καλείται μια ρουτίνα για επεξεργασία λίστας ή μια ρουτίνα για επεξεργασία λέξεων. Στην συνέχεια τρέχει ένα τρίτο πρόγραμμα ανεξάρτητα από τον τύπο των δεδομένων.

```

TO ΔΙΑΚΛΑΔΩΣΗ :ΑΝΤΙΚΕΙΜΕΝΟ
IF LISTP :ΑΝΤΙΚΕΙΜΕΝΟ [ΠΡΟΓΡΑΜΜΑ1 :ΑΝΤΙΚΕΙΜΕΝΟ] [ΠΡΟΓΡΑΜΜΑ2 :ΑΝΤΙΚΕΙΜΕΝΟ]
ΠΡΟΓΡΑΜΜΑ3
END

```

```

ΤΟ ΠΡΟΓΡΑΜΜΑ1 :ΛΙΣΤΑ
PR [ΕΙΜΑΙ ΤΟ ΠΡΟΓΡΑΜΜΑ ΓΙΑ ΤΙΣ ΛΙΣΤΕΣ]
( PR [ΚΑΙ ΕΠΕΞΕΡΓΑΖΟΜΑΙ ΤΗ ΛΙΣΤΑ] :ΛΙΣΤΑ )
END

```

```

ΤΟ ΠΡΟΓΡΑΜΜΑ2 :ΛΕΞΗ
PR [ΕΙΜΑΙ ΤΟ ΠΡΟΓΡΑΜΜΑ ΓΙΑ ΤΙΣ ΛΕΞΕΙΣ]
( PR [ΚΑΙ ΕΠΕΞΕΡΓΑΖΟΜΑΙ ΤΗ ΛΕΞΗ] :ΛΕΞΗ )
END

```

```

ΤΟ ΠΡΟΓΡΑΜΜΑ3
PR [ΕΙΜΑΙ ΤΟ ΤΜΗΜΑ ΜΕΤΑ ΤΗ ΔΙΑΚΛΑΔΩΣΗ]
PR [ΚΑΙ ΛΕΓΟΜΑΙ ΠΡΟΓΡΑΜΜΑ3]
END

```

```

?ΔΙΑΚΛΑΔΩΣΗ [ΕΓΩ ΕΣΥ ΑΥΤΟΣ]
ΕΙΜΑΙ ΤΟ ΠΡΟΓΡΑΜΜΑ ΓΙΑ ΤΙΣ ΛΙΣΤΕΣ
ΚΑΙ ΕΠΕΞΕΡΓΑΖΟΜΑΙ ΤΗ ΛΙΣΤΑ ΕΓΩ ΕΣΥ ΑΥΤΟΣ
ΕΙΜΑΙ ΤΟ ΤΜΗΜΑ ΜΕΤΑ ΤΗ ΔΙΑΚΛΑΔΩΣΗ
ΚΑΙ ΛΕΓΟΜΑΙ ΠΡΟΓΡΑΜΜΑ3
?ΔΙΑΚΛΑΔΩΣΗ "ΧΑΧΑΧΑ
ΕΙΜΑΙ ΤΟ ΠΡΟΓΡΑΜΜΑ ΓΙΑ ΤΙΣ ΛΕΞΕΙΣ
ΚΑΙ ΕΠΕΞΕΡΓΑΖΟΜΑΙ ΤΗ ΛΕΞΗ ΧΑΧΑΧΑ
ΕΙΜΑΙ ΤΟ ΤΜΗΜΑ ΜΕΤΑ ΤΗΝ ΔΙΑΚΛΑΔΩΣΗ
ΚΑΙ ΛΕΓΟΜΑΙ ΠΡΟΓΡΑΜΜΑ3

```

Μπορείτε να μελετήσετε δεδομένα ως προς τον τύπο τους.

```

ΤΟ ΠΟΙΟΣ.ΤΥΠΟΣ :ΑΝΤΙΚΕΙΜΕΝΟ
IF LISTP :ΑΝΤΙΚΕΙΜΕΝΟ [PR [ΕΙΝΑΙ ΛΙΣΤΑ]]
IF NUMBERP :ΑΝΤΙΚΕΙΜΕΝΟ [PR [ΕΙΝΑΙ ΑΡΙΘΜΟΣ]]
IF AND NOT NUMBERP :ΑΝΤΙΚΕΙΜΕΝΟ WORDP :ΑΝΤΙΚΕΙΜΕΝΟ [PR [ΕΙΝΑΙ ΛΕΞΗ]]
END

```

```

?ΠΟΙΟΣ.ΤΥΠΟΣ [Α Β Γ]
ΕΙΝΑΙ ΛΙΣΤΑ

```

```

?ΠΟΙΟΣ.ΤΥΠΟΣ "ΚΑΛΗΜΕΡΑ
ΕΙΝΑΙ ΛΕΞΗ

```

```

?ΠΟΙΟΣ.ΤΥΠΟΣ 123
ΕΙΝΑΙ ΑΡΙΘΜΟΣ

```

4.2 Τα Φρένα

Στα προγράμματα διακλαδώσεων προκύπτουν πολύ γρήγορα μικρά προβλήματα. Ξανακοιτάξτε το πρόγραμμα ΠΟΙΟΣ.ΤΥΠΟΣ. Αν, για παράδειγμα, το δεδομένο είναι μια λίστα, τότε με την πρώτη εντολή γίνεται αμέσως η σωστή αναγνώριση. Η εξέταση των επόμενων δύο εντολών είναι στην περίπτωση αυτή περιττή.

Μια ακόμη δυσχέρεια δημιουργείται αν θελήσετε να χρησιμοποιήσετε το πρόγραμμα ελέγχου που ακολουθεί, στη μορφή που παρουσιάζεται. Δέχεται μονάχα μια λέξη που ξεκινά με σύμφωνο. Οποιοσδήποτε άλλος τύπος αμφισβητείται.

```

ΤΟ ΕΛΕΓΧΟΣ :ΑΝΤΙΚΕΙΜΕΝΟ
IF LISTP :ΑΝΤΙΚΕΙΜΕΝΟ [PR [ΛΙΣΤΑ ΜΗ ΑΠΟΔΕΚΤΗ]]
IF NUMBERP :ΑΝΤΙΚΕΙΜΕΝΟ [PR [ΑΡΙΘΜΟΣ ΜΗ ΑΠΟΔΕΚΤΟΣ]]
IF MEMBERP FIRST :ΑΝΤΙΚΕΙΜΕΝΟ [Α Ε Ι Ο Υ] [PR [ΑΡΧΙΚΟ ΓΡΑΜΜΑ ΜΗ ΑΠΟΔΕΚΤΟ]]
( PR [ΟΛΑ ΕΝΤΑΞΥ ΜΕ] :ΑΝΤΙΚΕΙΜΕΝΟ
END

```

```

?ΕΛΕΓΧΟΣ "ΕΓΩ
ΤΟ ΑΡΧΙΚΟ ΓΡΑΜΜΑ ΜΗ ΑΠΟΔΕΚΤΟ
ΟΛΑ ΕΝΤΑΞΥ ΜΕ ΕΓΩ

```

```

?ΕΛΕΓΧΟΣ [ΕΙΜΑΙ ΕΔΩ]
ΛΙΣΤΑ ΜΗ ΑΠΟΔΕΚΤΗ
ΟΛΑ ΕΝΤΑΞΥ ΜΕ ΕΙΜΑΙ ΕΔΩ

```

Το πρόβλημα υπερνικάται αν το πρόγραμμα διακοπεί μετά την είσοδο στην αληθή περίπτωση. Μια τέτοια συνθήκη διακοπής βρίσκουμε στο φρένο μας, την εντολή STOP. Το πρόγραμμα μπορεί να διακοπεί σε οποιοδήποτε επιθυμητό σημείο με STOP.

Προσθέστε στο πρόγραμμα ΕΛΕΓΧΟΣ.ΣΤΟΙΧΕΙΩΝ σε κατάλληλες θέσεις εντολές STOP και δοκιμάστε το.

```

ΤΟ ΕΛΕΓΧΟΣ.ΣΤΟΙΧΕΙΩΝ :ΑΝΤΙΚΕΙΜΕΝΟ
IF LISTP :ΑΝΤΙΚΕΙΜΕΝΟ [PR [ΛΙΣΤΑ ΜΗ ΑΠΟΔΕΚΤΗ] STOP]
IF NUMBERP :ΑΝΤΙΚΕΙΜΕΝΟ [PR [ΑΡΙΘΜΟΣ ΜΗ ΑΠΟΔΕΚΤΟΣ] STOP]
IF MEMBERP FIRST :ΑΝΤΙΚΕΙΜΕΝΟ [Α Ε Ι Ο Υ Ω] [PR [ΑΡΧΙΚΟ ΓΡΑΜΜΑ ΜΗ ΑΠΟΔΕΚΤΟ] STOP]
( PR [ΟΛΑ ΕΝΤΑΞΥ ΜΕ] :ΑΝΤΙΚΕΙΜΕΝΟ
END

```

```

?ΕΛΕΓΧΟΣ.ΣΤΟΙΧΕΙΩΝ "ΕΓΩ
ΑΡΧΙΚΟ ΓΡΑΜΜΑ ΜΗ ΑΠΟΔΕΚΤΟ

```

```

?ΕΛΕΓΧΟΣ ΣΤΟΙΧΕΙΩΝ "ΜΕΝΑ
ΟΛΑ ΕΝΤΑΞΥ ΜΕ ΜΕΝΑ

```


Για να διευκρινιστούν καλύτερα όλ' αυτά, ξαναδείτε ένα πρόγραμμα χαιρετισμού που χωρίς την συνθήκη διακοπής STOP δεν θα είχε νόημα. Δοκιμάστε το αρχικά χωρίς την εντολή STOP, και μετά με την εντολή STOP.

```

TO HALLO :ΩΡΑ
IF OR :ΩΡΑ > 22 :ΩΡΑ < 4 [PR [ΚΑΛΗΝΥΧΤΑ]]
IF :ΩΡΑ > 18 [PR [ΚΑΛΗΣΠΕΡΑ]]
IF :ΩΡΑ >12 [PR [ΓΕΙΑ ΧΑΡΑ]]
IF :ΩΡΑ > 4 [PR [ΚΑΛΗΜΕΡΑ]]
END

```

```

?HALLO 10
ΚΑΛΟ ΞΗΜΕΡΩΜΑ
?HALLO 23
ΚΑΛΗΝΥΧΤΑ
ΚΑΛΗΣΠΕΡΑ
ΚΑΛΗΜΕΡΑ
ΚΑΛΟ ΞΗΜΕΡΩΜΑ

```

```

TO HALLO :ΩΡΑ
IF OR :ΩΡΑ > 22 :ΩΡΑ < 4 [PR [ΚΑΛΗΝΥΧΤΑ] STOP]
IF :ΩΡΑ > 18 [PR [ΚΑΛΗΣΠΕΡΑ] STOP]
IF :ΩΡΑ > 12 [PR [ΓΕΙΑ ΧΑΡΑ] STOP]
IF :ΩΡΑ > 4 [PR [ΚΑΛΗΜΕΡΑ]]
END

```

Ένας κοντινός συγγενής του φρένου που μέχρι τώρα χρησιμοποιήσαμε, είναι ο ήδη γνωστός μας προμηθευτής OUTPUT. Η OUTPUT σταματάει τα πάντα. Σε σύγκριση με το μέχρι τώρα φρένο μας, δεν θέτει απλά τέρμα, αλλά εκμαιεύει ακόμη αποτελέσματα από το πρόγραμμα και τα παραδίνει για ενδεχόμενη παραπέρα επεξεργασία. Δείτε κατάλληλα μορφοποιημένα προγράμματα χαιρετισμού και προγράμματα ελέγχου δεδομένων.

```

TO HALLO :ΩΡΑ
IF OR :ΩΡΑ >22 :ΩΡΑ < 4 [OP [ΚΑΛΗΝΥΧΤΑ]]
IF :ΩΡΑ >18 [OP [ΚΑΛΗΣΠΕΡΑ]]
IF :ΩΡΑ > 12 [OP [ΓΕΙΑ ΧΑΡΑ]]
IF :ΩΡΑ > 4 [OP [ΚΑΛΗΜΕΡΑ]]
END

```

```

ΤΟ ΕΛΕΓΧΟΣ.ΣΤΟΙΧΕΙΩΝ :ΑΝΤΙΚΕΙΜΕΝΟ
IF LISTP :ΑΝΤΙΚΕΙΜΕΝΟ [ΟΡ [ΛΙΣΤΑ ΜΗ ΑΠΟΔΕΚΤΗ]]
IF NUMBERP :ΑΝΤΙΚΕΙΜΕΝΟ [ΟΡ [ΑΡΙΘΜΟΣ ΜΗ ΑΠΟΔΕΚΤΟΣ]]
IF MEMBERP FIRST :ΑΝΤΙΚΕΙΜΕΝΟ [Α Ε Η Ι Ο Υ Ω] [ΟΡ [ΑΡΧΙΚΟ ΓΡΑΜΜΑ ΜΗ ΑΠΟΔΕΚΤΟ]]
ΟΡ SE [ΟΛΑ ΕΝΤΑΞΕΙ ΜΕ] :ΑΝΤΙΚΕΙΜΕΝΟ
END

```

```

?PRINT ΕΛΕΓΧΟΣ.ΣΤΟΙΧΕΙΩΝ "ΕΓΩ
ΑΡΧΙΚΟ ΓΡΑΜΜΑ ΜΗ ΑΠΟΔΕΚΤΟ

```

Το παραπάνω πρόγραμμα ελέγχου μπορεί να μετατραπεί σε λειτουργία ελέγχου που θα παρέχει "TRUE και "FALSE και επιπλέον κάποια σχόλια.

```

ΤΟ ΛΑΘΟΣ.ΣΤΟΙΧΕΙΩΝ :ΑΝΤΙΚΕΙΜΕΝΟ
IF LISTP :ΑΝΤΙΚΕΙΜΕΝΟ [PR [ΛΙΣΤΑ ΜΗ ΑΠΟΔΕΚΤΗ] ΟΡ "TRUE]
IF NUMBERP :ΑΝΤΙΚΕΙΜΕΝΟ [PR [ΑΡΙΘΜΟΣ ΜΗ ΑΠΟΔΕΚΤΟΣ] ΟΡ "TRUE]
IF MEMBERP FIRST :ΑΝΤΙΚΕΙΜΕΝΟ [Α Ε Η Ι Ο Υ Ω] [PR [ΑΡΧΙΚΟ ΓΡΑΜΜΑ ΜΗ ΑΠΟΔΕΚΤΟ] ΟΡ "TRUE]
OUTPUT "FALSE
END

```

```

?PR ΛΑΘΟΣ.ΣΤΟΙΧΕΙΩΝ "ΕΓΩ
ΑΡΧΙΚΟ ΓΡΑΜΜΑ ΜΗ ΑΠΟΔΕΚΤΟ
TRUE

```

Χρησιμοποιήστε την καινούργια λέξη ελέγχου στο επόμενο παράδειγμα.

```

ΤΟ DEMO: ΛΕΞΗ
IF ΛΑΘΟΣ.ΣΤΟΙΧΕΙΩΝ :ΛΕΞΗ [PR "ΔΙΑΚΟΠΗ ΠΡΟΓΡΑΜΜΑΤΟΣ STOP]
(PR [ΜΕ ΤΗ ΣΩΣΤΗ ΤΙΜΗ] :ΛΕΞΗ [ΘΑ ΣΥΝΕΧΙΣΩ])
END

```

```

?DEMO "ΕΓΩ
ΑΡΧΙΚΟ ΓΡΑΜΜΑ ΜΗ ΑΠΟΔΕΚΤΟ
ΔΙΑΚΟΠΗ.ΠΡΟΓΡΑΜΜΑΤΟΣ

```

```

? DEMO "ΣΥ
ΜΕ ΤΗ ΣΩΣΤΗ ΤΙΜΗ ΣΥ ΘΑ ΣΥΝΕΧΙΣΩ

```

4.3 Η Μεγάλη Στροφή

Κατά τη διάρκεια της εκμάθησης του μικρού αλφαβήτου της Logo χρησιμοποιήθηκε μια πληθώρα προγραμμάτων που καλούν τον εαυτό τους. Αυτές οι ατέρμονες αλυσίδες σταματούσαν μόνο βίαια μέσω του πληκτρολόγιου. Στη θέση του άκομψου τραβήγματος του φρένου κινδύνου (CTRL-BREAK) θα οδηγήσουμε τώρα τα επαναλαμβανόμενα προγράμματα σε ελεγχόμενο τέλος.

Ξανακοιτάξετε άλλη μια φορά το πρόγραμμα ΟΛΑ.ΤΑ.ΧΡΩΜΑΤΑ του Μαθήματος 11. Το πρόγραμμα διακόπτεται βίαια όταν η εντολή SETBG φτάσει στον αριθμό 16.

```

TO ΟΛΑ.ΤΑ.ΧΡΩΜΑΤΑ :ΔΙΑΚ.ΧΡΩΜΑ
  CLEARTEXT
  SETBG :ΔΙΑΚ.ΧΡΩΜΑ
  PRINT :ΔΙΑΚ.ΧΡΩΜΑ
  WAIT 60
  ΟΛΑ.ΤΑ.ΧΡΩΜΑΤΑ :ΔΙΑΚ.ΧΡΩΜΑ + 1
END

?ΟΛΑ.ΤΑ.ΧΡΩΜΑΤΑ 13
13
14
15
SETBG DOESN'T LIKE 16 AS INPUT IN ΟΛΑ.ΤΑ.ΧΡΩΜΑΤΑ

```

Η οριακή αυτή περίπτωση μπορεί ν' αντιμετωπιστεί τώρα με ακρίβεια. Αν παρουσιαστεί, ο φίλος μας το φρένο ξέρει να κάνει την δουλειά του.

```

TO ΟΛΑ.ΤΑ.ΧΡΩΜΑΤΑ :ΔΙΑΚ.ΧΡΩΜΑ
  IF :ΔΙΑΚ.ΧΡΩΜΑ > 15 [STOP]
  CLEARTEXT
  SETBG :ΔΙΑΚ.ΧΡΩΜΑ
  PRINT :ΔΙΑΚ.ΧΡΩΜΑ
  WAIT 60
  ΟΛΑ.ΤΑ.ΧΡΩΜΑΤΑ :ΔΙΑΚ.ΧΡΩΜΑ + 1
END

```

Όλοι οι πειραματισμοί που κάνατε με τα ελικοειδή και τ' άλλα αναδρομικά προγράμματα είχαν το ίδιο μειονέκτημα. Για να διακοπεί ελεγχόμενα ένα πρόγραμμα δεν χρειάζεται παρά να προβλέψετε έναν αριθμό που να ορίζει το πλήθος των επαναλήψεων, ή μια συγκριτική τιμή.

```

TO ΣΟΥΠΕΡ.ΕΛΙΚ :ΒΗΜΑΤΑ :ΕΠΕΚΤΑΣΗ :ΜΟΙΡΕΣ :ΤΕΛΟΣ
IF :ΒΗΜΑΤΑ > :ΤΕΛΟΣ [STOP]
FD :ΒΗΜΑΤΑ RT :ΜΟΙΡΕΣ
ΣΟΥΠΕΡ.ΕΛΙΚ :ΒΗΜΑΤΑ + :ΕΠΕΚΤΑΣΗ :ΕΠΕΚΤΑΣΗ :ΜΟΙΡΕΣ :ΤΕΛΟΣ
END

```

```

TO ΣΟΥΠΕΡ.ΕΛΙΚ1 :ΒΗΜΑΤΑ :ΕΠΕΚΤΑΣΗ :ΜΟΙΡΕΣ :ΠΑΥΣΟΣ
IF :ΠΑΥΣΟΣ = 0 [STOP]
FD :ΒΗΜΑΤΑ RT :ΜΟΙΡΕΣ
ΣΟΥΠΕΡ.ΕΛΙΚ :ΒΗΜΑΤΑ + :ΕΠΕΚΤΑΣΗ :ΕΠΕΚΤΑΣΗ :ΜΟΙΡΕΣ :ΠΑΥΣΟΣ - 1
END

```

Θυμηθείτε και το πρόγραμμα ΣΥΝΔΕΣΗ.ΣΗΜΕΙΩΝ του Μαθήματος 12. Το σύμβολο ενός περιγράμματος τελειώνει ελεγχόμενα, μετά το τελευταίο ζεύγος τιμών.

```

TO ΣΥΝΔΕΣΗ :ΤΙΜΕΣ
IF EMPTY? :ΤΙΜΕΣ [STOP]
SETPOS FIRST :ΤΙΜΕΣ
ΣΥΝΔΕΣΗ BUTFIRST :ΤΙΜΕΣ
END

```

Στη συνέχεια παρουσιάζονται μερικά ακόμη παραδείγματα που αναφέρονται στην επεξεργασία λέξεων και λιστών. Το επόμενο πρόγραμμα εκτύπωσης ΚΑΤΩ τυπώνει τα γράμματα μιας λέξης ή τα στοιχεία μιας λίστας το ένα κάτω από το άλλο.

```

TO ΚΑΤΩ :ΑΝΤ
IF EMPTY? :ΑΝΤ [STOP]
PRINT FIRST :ΑΝΤ
ΚΑΤΩ BF :ΑΝΤ
END

```

```

?ΚΑΤΩ "ΣΗΜΕΡΑ
Σ
Η
Μ
Ε
Ρ
Α

```

Το επόμενο πρόγραμμα γράφει τις λέξεις που δίνονται σαν δεδομένα, αντεστραμένα. Τι συμβαίνει αλήθεια αν η εντολή PRINT αντικατασταθεί με την εντολή TYPE;

```

ΤΟ ΚΑΘΡΕΦΤΗΣ :ΤΙ
IF ΕΜΠΤΥΡ :ΤΙ [STOP]
TYPE LAST :ΤΙ
ΚΑΘΡΕΦΤΗΣ ΒΛ :ΤΙ
END

```

```

?ΚΑΘΡΕΦΤΗΣ "ΣΗΜΕΡΑ
ΑΡΕΜΗΣ?

```

```

?ΚΑΘΡΕΦΤΗΣ [ΕΓΩ ΕΣΥ ΑΥΤΟΣ]
ΑΥΤΟΣ ΕΣΥ ΕΓΩ?

```

Μετατρέψτε λίγο το πρόγραμμα ΚΑΘΡΕΦΤΗΣ δημιουργώντας το πρόγραμμα ΝΕΟΣ.ΚΑΘΡΕΦΤΗΣ. Φτιάξτε το έτσι ώστε να επεξεργάζεται μόνο λίστες (έλεγχος δεδομένων!) και επιπλέον να αντιστρέφει κάθε μεμονωμένη λέξη. Για παράδειγμα, όταν δίνεται η λίστα [ΕΓΩ ΕΣΥ ΑΥΤΟΣ] να γράφεται [ΣΟΤΥΑ ΥΣΕ ΩΓΕ].

```

ΤΟ ΝΕΟΣ.ΚΑΘΡΕΦΤΗΣ :ΛΙΣΤΑ
IF ΕΜΠΤΥΡ :ΛΙΣΤΑ [STOP]
ΚΑΘΡΕΦΤΗΣ LAST :ΛΙΣΤΑ
TYPE "
ΝΕΟΣ.ΚΑΘΡΕΦΤΗΣ ΒΛ :ΛΙΣΤΑ
END

```

```

?ΚΑΘΡΕΦΤΗΣ [ΣΗΜΕΡΑ ΑΥΡΙΟ ΧΘΕΣ]
ΧΘΕΣΣΗΜΕΡΑΑΥΡΙΟ?

```

```

?ΝΕΟΣ.ΚΑΘΡΕΦΤΗΣ [ΣΗΜΕΡΑ ΑΥΡΙΟ ΧΘΕΣ]
ΣΕΒΧΟΙΡΥΑΑΡΕΜΗΣ ?

```

Υπολογίστε το άθροισμα των ψηφίων ενός αριθμού. Τα ψηφία διαβάζονται μεμονωμένα και η επεξεργασία τους γίνεται σε μια μνήμη. Το κόλπο είναι να εισαχθεί αμέσως στην πρώτη σειρά η κατάλληλη μεταβλητή.

```

ΤΟ ΑΘΡΟΙΣΜΑ :ΑΡΙΘΜΟΣ :ΣΥΝΟΛΟ
IF ΕΜΡΤΥΡ :ΑΡΙΘΜΟΣ [PR :ΣΥΝΟΛΟ STOP]
ΑΘΡΟΙΣΜΑ ΒF :ΑΡΙΘΜΟΣ :ΣΥΝΟΛΟ + FIRST :ΑΡΙΘΜΟΣ
END

```

```

?ΑΘΡΟΙΣΜΑ 123 0
6

```

Τελειώνοντας, γράψτε ελέγχους.

Οι αντίστοιχοι βαθμοί κάθε μαθήματος να δοθούν σαν κείμενο και όχι σαν αριθμοί. Πάρτε τη συνηθισμένη κλίμακα βαθμών από άριστα μέχρι κακώς.

Το πρόγραμμα ΜΙΝΙ.ΕΛΕΓΧΟΣ θα τυπώνει βαθμούς που έχουν δοθεί σαν αριθμοί σε μια λίστα. Η δεύτερη σειρά σας είναι ήδη αρκετά οικεία. Το ΝΙΟΣΤΟ το γνωρίσατε στην παράγραφο 3.4 με την μορφή ΝΙΟΣΤΟ.ΣΤΟΙΧΕΙΟ. Εδώ όμως το ΝΙΟΣΤΟ δεν είναι λειτουργία αλλά σε τελική ανάλυση είναι μια εντολή εκτύπωσης. Το ΚΛΙΜΑΚΑ.ΒΑΘΜΩΝ προσδιορίζει την λίστα των δεδομένων που απαιτούνται.

```

ΤΟ ΜΙΝΙ.ΕΛΕΓΧΟΣ :ΒΑΘΜΟΙ
IF ΕΜΡΤΥΡ :ΒΑΘΜΟΙ [STOP]
ΝΙΟΣΤΟ ΚΛΙΜΑΚΑ.ΒΑΘΜΩΝ FIRST :ΒΑΘΜΟΙ
ΜΙΝΙ.ΕΛΕΓΧΟΣ ΒF :ΒΑΘΜΟΙ
END

```

```

ΤΟ ΚΛΙΜΑΚΑ.ΒΑΘΜΩΝ
ΟΡ [ΑΡΙΣΤΑ ΛΙΑΝ_ΚΑΛΩΣ ΚΑΛΩΣ ΜΕΤΡΙΩΣ ΑΝΕΠΑΡΚΩΣ ΚΑΚΩΣ]
END

```

```

ΤΟ ΝΙΟΣΤΟ :ΑΝΤ :ΘΕΣΗ
IF :ΘΕΣΗ = 1 [PR FIRST :ΑΝΤ STOP]
ΝΙΟΣΤΟ ΒF :ΑΝΤ :ΘΕΣΗ - 1
END

```

```

ΜΙΝΙ.ΕΛΕΓΧΟΣ [6 1 5 2 3 4 2]
ΚΑΚΩΣ
ΑΡΙΣΤΑ
ΑΝΕΠΑΡΚΩΣ
ΛΙΑΝ_ΚΑΛΩΣ
ΚΑΛΩΣ
ΜΕΤΡΙΩΣ
ΛΙΑΝ_ΚΑΛΩΣ

```

Το πρόγραμμα MINI.ΕΛΕΓΧΟΣ πρέπει να διευρυνθεί με την εισαγωγή :ΜΑΘΗΜΑ, και στη δεύτερη γραμμή του να τυπώνεται επιπλέον το πρώτο στοιχείο της μεταβλητής :ΜΑΘΗΜΑ. Με τις προσθήκες αυτές το πρόγραμμα των ελέγχων θα έχει ολοκληρωθεί:

```

ΤΟ ΕΛΕΓΧΟΣ :ΒΑΘΜΟΙ :ΜΑΘΗΜΑ
IF EMPTY :ΒΑΘΜΟΙ [STOP]
TYPE FIRST :ΜΑΘΗΜΑ
ΝΙΟΣΤΟ ΚΛΙΜΑΚΑ.ΒΑΘΜΩΝ FIRST :ΒΑΘΜΟΙ
ΕΛΕΓΧΟΣ BF :ΒΑΘΜΟΙ BF :ΜΑΘΗΜΑ
END

```

```

?ΕΛΕΓΧΟΣ [6 1 2] [ΒΙΟΛΟΓΙΑ ΓΕΡΜΑΝΙΚΑ ΜΑΘΗΜΑΤΙΚΑ]
ΒΙΟΛΟΓΙΑΚΑΚΩΣ
ΓΕΡΜΑΝΙΚΑΑΡΙΣΤΑ
ΜΑΘΗΜΑΤΙΚΑΛΙΑΝΚΑΛΩΣ

```

Η δοκιμή του παραπάνω προγράμματος δείχνει ότι υπάρχει ένα μικρό αισθητικό λάθος. Οι βαθμοί θα έπρεπε να είναι γραμμένοι λίγο δεξιότερα και τα αρχικά γράμματα να βρίσκονται ακριβώς το ένα κάτω από το άλλο. Πρέπει να προβλεφθεί ένα σημείο απ' όπου σε κάθε σειρά να ξεκινά η γραφή κάθε μαθήματος. Μια τέτοια συνάρτηση είναι η TAB:

```

ΤΟ TAB :ΘΕΣΗ
IF LAST CURSOR = :ΘΕΣΗ [STOP]
TYPE ".
TAB :ΘΕΣΗ
END

```

Η μεταβλητή :ΘΕΣΗ αποτελεί το επιθυμητό σημείο γραφής και μπορεί να πάρει τιμές από 1 μέχρι 40. Η συνάρτηση CURSOR της Logo επιστρέφει την τρέχουσα θέση του δρομέα πάνω στην οθόνη. Η CURSOR δίνει μια λίστα με δύο στοιχεία. Το πρώτο στοιχείο προσδιορίζει το σημείο της συγκεκριμένης γραμμής, ενώ το δεύτερο τον αριθμό της γραμμής. Η TAB αφήνει κενά κατά τη διάρκεια της εκτύπωσης, μέχρι το σημείο που έχει προσδιοριστεί.

Προσθέστε την TAB στο πρόγραμμά σας και δοκιμάστε πως δουλεύει:

```

ΤΟ ΕΛΕΓΧΟΣ :ΒΑΘΜΟΙ :ΜΑΘΗΜΑ
IF EMPTY :ΒΑΘΜΟΙ [STOP]
TYPE FIRST :ΜΑΘΗΜΑ
TAB 15

```

```

ΝΙΟΣΤΟ ΚΛΙΜΑΚΑ.ΒΑΘΜΩΝ FIRST :ΒΑΘΜΟΙ
ΕΛΕΓΧΟΣ BF :ΒΑΘΜΟΙ BF :ΜΑΘΗΜΑ
END

?ΕΛΕΓΧΟΣ [6 1 2] [ΒΙΟΛΟΓΙΑ ΓΕΡΜΑΝΙΚΑ ΜΑΘΗΜΑΤΙΚΑ]
ΒΙΟΛΟΓΙΑ.....ΚΑΚΩΣ
ΓΕΡΜΑΝΙΚΑ.....ΑΡΙΣΤΑ
ΜΑΘΗΜΑΤΙΚΑ....ΛΙΑΝ_ΚΑΛΩΣ

```

4.4 Ασκήσεις

Μέχρι τώρα, τα αναδρομικά μας προγράμματα το μόνο που έκαναν ήταν να τυπώνουν. Τι θα λέγατε αν αντί αυτά τα προγράμματα να σταματούν με τη βοήθεια του φίλου μας του φρένου, διακόπτονταν από το συγγενή του, τον προμηθευτή OUTPUT. Το πλεονέκτημα θα ήταν ότι θα έβγαινε και κάποιο αποτέλεσμα το οποίο μπορεί να χρησιμοποιηθεί σε κάποια άλλη επεξεργασία. Η απλή εκτύπωση δεν είναι παρά μια ειδική περίπτωση. Ξαναδείτε το πρόγραμμα ΚΑΘΡΕΦΤΗΣ και σκεφτείτε τι μορφή θα έπαιρνε με την εντολή OUTPUT.

Επομένως, το νέο πρόγραμμα ΚΑΘΡΕΦΤΗΣ πρέπει να παρέχει στην θέση του ειδώλου το σύνολο. Αν σαν δεδομένο εισάγεται μια λέξη, τότε η εντολή OUTPUT παρέχει μια ολόκληρη λέξη. Αντιπαραθέτουμε εδώ το πρόγραμμα στην πρώτη του μορφή με την παλιά έκδοσή του που περιείχε τη STOP.

```

ΤΟ ΚΑΘΡΕΦΤΗΣ :ANT      ΤΟ ΚΑΘΡΕΦΤΗΣ :ΛΕΞΗ
IF EMPTYP :ANT [STOP]  IF EMPTYP :ΛΕΞΗ [OP :ΛΕΞΗ]
TYPE LAST :ANT         ??????
ΚΑΘΡΕΦΤΗΣ BL :ANT     END
END

```

Μέχρι αυτό το σημείο ουσιαστικά δεν αναμένεται τίποτα διαφορετικό. Το STOP έπρεπε να φύγει, αφού το πρόγραμμα σταματά με OUTPUT. Επιπλέον έπρεπε να φύγει και η εντολή εκτύπωσης. Τι θα μπει στη θέση τους;

Συμπληρώνεται ακόμη πως το καινούργιο πρόγραμμα του καθρέφτη, στη μορφή που είναι απεικονισμένο παραπάνω, τρέχει, τρέχει όμως μόνο για την ιδιάζουσα περίπτωση της κενής λέξης.

```

?PR ΚΑΘΡΕΦΤΗΣ "
?
```


ΜΑΘΕΤΕ ΤΗΝ LOGO

στον **IBM PC** και τους συμβατούς

Καλωσορίσατε στη Logo: Η απλή και φιλική αυτή γλώσσα προγραμματισμού θα βοηθήσει το μαθητή στην πρώτη του γνωριμία με τον κόσμο των υπολογιστών, ενώ θα επιτρέψει και στους γονείς να έχουν ενεργή συμμετοχή σ' αυτό.

Στο βιβλίο αυτό θα δουλέψετε, θα παίξετε και θα πειραματιστείτε με γραφικά, κείμενο, και μουσική. Με τη βοήθεια της χελώνας, σε 12 μαθήματα, θα μάθετε το «μικρό αλφαβητάρι» της Logo. Στη συνέχεια θα μπειτε σε πιο προχωρημένα θέματα, και θα μάθετε να φτιάχνετε δικά σας προγράμματα, χρησιμοποιώντας και τα μικρά, βοηθητικά προγράμματα του βιβλίου. Παρουσιάζεται επίσης η γραμματική της Logo.

Το βιβλίο απαιτεί από σας ενεργή συμμετοχή. Θα πρέπει να το μελετάτε καθισμένοι στον υπολογιστή σας, για να βλέπετε άμεσα τα αποτελέσματα της μελέτης σας. Θα δείτε ότι έτσι η εκμάθηση της Logo είναι διασκέδαση!

Τέλος, θα βρείτε Παραρτήματα με χρήσιμα υπο-προγράμματα, όλες τις εντολές της Logo και τη χρήση τους, τη χρήση των ειδικών πλήκτρων, κ.α.



ΚΛΕΙΔΑΡΙΘΜΟΣ

ΒΙΒΛΙΑ ΥΠΟΛΟΓΙΣΤΩΝ

ISBN 960-209-024-3