# Μάθημα 110

## Εισαγωγή στον προγραμματισμό

# Η έννοια του προγράμματος

**Τι είναι πρόγραμμα;**

Πρόγραμμα είναι το σύνολο των εντολών που πρέπει να δοθούν στον υπολογιστή, ώστε να υλοποιηθεί ο αλγόριθμος για την επίλυση του προβλήματος.

**Ποια είναι τα βασικά στοιχεία ενός προγράμματος;**

Βασικά στοιχεία του προγράμματος είναι ο αλγόριθμος, τα δεδομένα και οι δομές δεδομένων επί των οποίων ενεργεί.

**Στάδια επίλυσης ενός προβλήματος με υπολογιστή. Με ποιο ασχολείται ο προγραμματισμός;**

Η επίλυση ενός προβλήματος με τον υπολογιστή περιλαμβάνει τρία εξίσου σημαντικά στάδια.

- Τον ακριβή προσδιορισμό του προβλήματος.
- Την ανάπτυξη του αντίστοιχου αλγορίθμου.
- Τη διατύπωση του αλγορίθμου σε κατανοητή μορφή από τον υπολογιστή.

Με αυτό το στάδιο ασχολείται ο προγραμματισμός, δηλαδή τη δημιουργία του προγράμματος.

# Φυσικές και Τεχνητές Γλώσσες

Οι φυσικές γλώσσες εξελίσσονται συνεχώς, ενώ οι τεχνητές παραμένουν στάσιμες

**Ποια είναι τα στοιχεία που προσδιορίζουν μία γλώσσα;**

Μία γλώσσα προσδιορίζεται από το αλφάβητό της, το λεξιλόγιό της, τη γραμματική της (τυπικό και συντακτικό) και τέλος τη σημασιολογία της.

πχ. οι χαρακτήρες **Α-Ω** και **α-ω** και τα **σημεία στίξης** αποτελούν το αλφάβητο της ελληνικής γλώσσας.

**Τι είναι το αλφάβητο μιας γλώσσας; Τι είναι το λεξιλόγιο μιας γλώσσας;**

**Αλφάβητο** μίας γλώσσας καλείται το σύνολο των στοιχείων που χρησιμοποιείται από τη γλώσσα.

Το **λεξιλόγιο** αποτελείται από ένα υποσύνολο όλων των ακολουθιών που δημιουργούνται από τα στοιχεία του αλφαβήτου, δηλαδή τις λέξεις που είναι δεκτές από τη γλώσσα.

πχ. η ακολουθία χαρακτήρων **ΓΡΑΨΕ** είναι δεκτή στην ΓΛΩΣΣΑ, καθώς αποτελεί λέξη της και άρα ανήκει στο λεξιλόγιό της, ενώ η ακολουθία **ΓΡΑΦΕ** δεν είναι δεκτή καθώς δεν αποτελεί λέξη της.

**Τι είναι το τυπικό και τι το συντακτικό της γραμματικής μιας γλώσσας;**

**Τυπικό** είναι το σύνολο των κανόνων που ορίζει τις μορφές με τις οποίες μία λέξη είναι αποδεκτή.

**Συντακτικό** είναι το σύνολο των κανόνων που καθορίζει τη νομιμ[illegible] της διάταξης και της σύνδεσης των λέξεων της γλώσσας για τη δημιουργία προτάσεων.

πχ. οι λέξεις **αλλιώς** και **Αλλιώς** είναι αποδεκτές μορφές της ίδιας λέξης στην ψευδογλώσσα , όχι όμως και η λέξη **αλιώς**

π.χ. η έκφραση **για** i **μέχρι** 3 **από** 1 δεν είναι σωστά συνταγμένη στην ψευδογλώσσα

**Τι σημαίνει σημασιολογία μίας γλώσσας;**

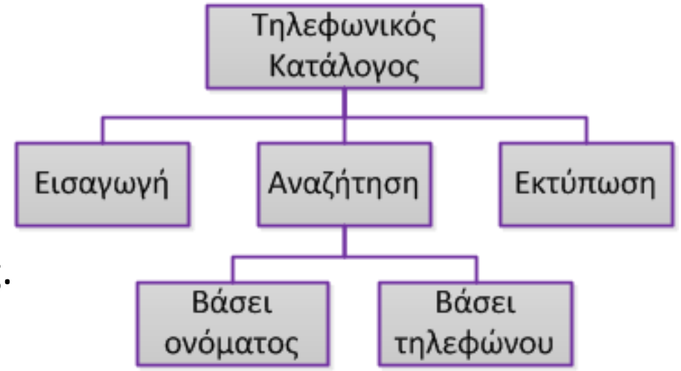
Η **σημασιολογία** είναι το σύνολο των κανόνων που καθορίζει το νόημα των λέξεων και κατά επέκταση των εκφράσεων και προτάσεων που χρησιμοποιούνται σε μία γλώσσα.

# Μεθοδολογίες προγραμματισμού

**Ιεραρχική σχεδίαση προγράμματος (από πάνω προς τα κάτω)**

Σκοπός της είναι η διάσπαση του προβλήματος σε μία σειρά από απλούστερα υποπροβλήματα, τα οποία να είναι εύκολο να επιλυθούν, οδηγώντας έτσι στην επίλυση του αρχικού προβλήματος.

Ως υποβοήθηση της ιεραρχικής σχεδίασης, χρησιμοποιούνται διαγραμματικές τεχνικές.




**Δομημένος προγραμματισμός**

Στηρίζεται στη χρήση τριών και μόνο στοιχειωδών λογικών δομών, τη δομή της ακολουθίας, τη δομή της επιλογής και τη δομή της επανάληψης.

Όλα τα προγράμματα μπορούν να γραφούν χρησιμοποιώντας μόνο αυτές τις τρεις δομές καθώς και συνδυασμό τους.

Κάθε πρόγραμμα, όπως και κάθε ενότητα, έχει μόνο μία είσοδο και μόνο μία έξοδο.

Αρχικά (μέσα 1960) εμφανίστηκε σαν μία προσπάθεια περιορισμού των εντολών GOTO, οι οποίες άλλαζαν συνεχώς τη ροή του προγράμματος και έχει εξελιχθεί στη βασική μεθοδολογία προγραμματισμού.

**Πλεονεκτήματα δομημένου προγραμματισμού**

- Δημιουργία απλούστερων προγραμμάτων.
- Άμεση μεταφορά των αλγορίθμων σε προγράμματα.
- Διευκόλυνση ανάλυσης του προγράμματος σε τμήματα.
- Περιορισμός των λαθών κατά την ανάπτυξη του προγράμματος.
- Διευκόλυνση στην ανάγνωση, κατανόηση του προγράμματος από τρίτους.
- Ευκολότερη διόρθωση και συντήρηση.

# Προγραμματιστικά περιβάλλοντα / μετατροπή σε γλώσσα μηχανής

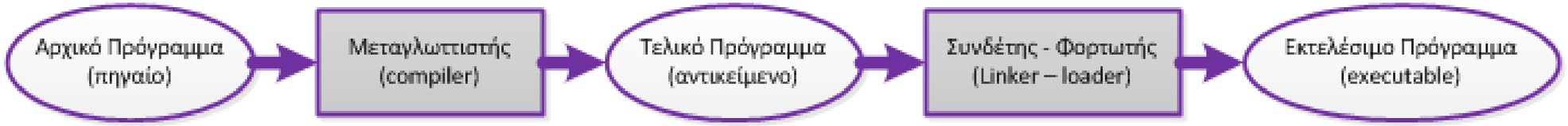
**Τι είναι ο μεταγλωττιστής;**

Ο μεταγλωττιστής είναι ένα μεταφραστικό πρόγραμμα που δέχεται στην είσοδο ένα πρόγραμμα γραμμένο σε μια γλώσσα υψηλού επιπέδου και παράγει ένα ισοδύναμο πρόγραμμα σε γλώσσα μηχανής.

Το τελευταίο μπορεί να εκτελείται οποτεδήποτε από τον υπολογιστή και είναι τελείως ανεξάρτητο από το αρχικό πρόγραμμα.

**Πως λειτουργεί ο μεταγλωττιστής**

1. Λαμβάνει στην είσοδό του ένα πρόγραμμα γραμμένο σε μία γλώσσα υψηλού επιπέδου, το λεγόμενο πηγαίο πρόγραμμα.
2. Παράγει ένα ισοδύναμο πρόγραμμα σε γλώσσα μηχανής, το οποίο ονομάζεται αντικείμενο πρόγραμμα.
3. Το αντικείμενο πρόγραμμα συνδέεται με άλλα τμήματα προγράμματος που είτε τα γράφει ο προγραμματιστής, είτε βρίσκονται σε βιβλιοθήκες, μέσω ενός προγράμματος που ονομάζεται συνδέτης – φορτωτής.
4. Ο συνδέτης παράγει το εκτελέσιμο πρόγραμμα, το οποίο είναι αυτό που εκτελείται από τον υπολογιστή.



# Προγραμματιστικά περιβάλλοντα / μετατροπή σε γλώσσα μηχανής

**Τι είναι ο διερμηνευτής και πως λειτουργεί ;**

Ο διερμηνευτής είναι ένα μεταφραστικό πρόγραμμα που διαβάζει μία προς μία τις εντολές του αρχικού προγράμματος και για καθεμία εκτελεί αμέσως μια ισοδύναμη ακολουθία εντολών μηχανής.

**Τι είναι ο συντάκτης;**

Για την αρχική σύνταξη των προγραμμάτων και τη διόρθωσή τους στη συνέχεια χρησιμοποιείται ένα ειδικό πρόγραμμα που ονομάζεται συντάκτης. Ο συντάκτης είναι ουσιαστικά ένας μικρός επεξεργαστής κειμένου, με δυνατότητες όμως που διευκολύνουν τη γρήγορη γραφή των εντολών των προγραμμάτων.

**Πλεονεκτήματα και μειονεκτήματα μεταγλωττιστών και διερμηνευτών;**

Η χρήση μεταγλωττιστή έχει το μειονέκτημα της μεταγλώττισης και σύνδεσης ολόκληρου του προγράμματος πριν από την εκτέλεση σε αντίθεση με την χρήση διερμηνευτή που έχει το πλεονέκτημα της άμεσης εκτέλεσης και διόρθωσης.

Από την άλλη η χρήση διερμηνευτή καθιστά την εκτέλεση του προγράμματος πιο αργή σε σχέση με αυτή του ισοδύναμου εκτελέσιμου προγράμματος που παράγει ο μεταγλωττιστής.

# Άσκηση 1 / σελίδα 255

Το παρακάτω τμήμα αλγορίθμου αποτελεί τμήμα μη δομημένου προγράμματος.

Να γραφεί τμήμα αλγορίθμου σχεδιασμένο με τις αρχές του δομημένου προγραμματισμού, που να εκτελεί τις ίδιες λειτουργίες.

```
    Αν X mod 2 = 0 τότε
      GOTO 1
    Τέλος_αν
    GOTO 2
1:  Εμφάνισε "ΑΡΤΙΟΣ"
2:  ! Συνέχεια
```

```
Αν X mod 2 = 0 τότε
  Εμφάνισε "ΑΡΤΙΟΣ"
Τέλος_αν
! Συνέχεια
```

# Άσκηση 3 / σελίδα 255

Το παρακάτω τμήμα αλγορίθμου αποτελεί τμήμα μη δομημένου προγράμματος.

Να γραφεί τμήμα αλγορίθμου σχεδιασμένο με τις αρχές του δομημένου προγραμματισμού, που να εκτελεί τις ίδιες λειτουργίες.

```
    Αν X mod 2 = 0 τότε GOTO 1
    Αν X mod 2 <> 0 τότε GOTO 2
1:  Εμφάνισε "ΑΡΤΙΟΣ"
    GOTO 3
2:  Εμφάνισε "ΠΕΡΙΤΤΟΣ"
3:  ! Συνέχεια
```

```
Αν X mod 2 = 0 τότε
   Εμφάνισε "ΑΡΤΙΟΣ"
Τέλος_αν
Αν X mod 2 <> 0 τότε
   Εμφάνισε "ΠΕΡΙΤΤΟΣ"
Τέλος_αν
! Συνέχεια
```

```
Αν X mod 2 = 0 τότε
   Εμφάνισε "ΑΡΤΙΟΣ"
Αλλιώς
   Εμφάνισε "ΠΕΡΙΤΤΟΣ"
Τέλος_αν
! Συνέχεια
```

# Άσκηση 5 / σελίδα 256

Το παρακάτω τμήμα αλγορίθμου αποτελεί τμήμα μη δομημένου προγράμματος.

Να γραφεί τμήμα αλγορίθμου σχεδιασμένο με τις αρχές του δομημένου προγραμματισμού, που να εκτελεί τις ίδιες λειτουργίες.

```
    Αν Β > 18.5 τότε GOTO 1
    Αν Β > 17.5 τότε GOTO 2
    Αν Β > 13.5 τότε GOTO 3
    Αν Β >= 10 τότε GOTO 4
    Εμφάνισε "Ανεπαρκής"
    GOTO 5
1:  Εμφάνισε "Άριστα"
    GOTO 5
2:  Εμφάνισε "Λίαν Καλώς"
    GOTO 5
3:  Εμφάνισε "Καλώς"
    GOTO 5
4:  Εμφάνισε "Επαρκής"
5:  ! Συνέχεια
```

```
Αν Β > 18.5 τότε
    Εμφάνισε "Άριστα"
Τέλος_αν
Αν Β > 17.5 τότε
    Εμφάνισε "Λίαν Καλώς"
Τέλος_αν
Αν Β > 13.5 τότε
    Εμφάνισε "Καλώς"
Τέλος_αν
Αν Β >= 10 τότε
    Εμφάνισε "Επαρκής"
Τέλος_αν
Εμφάνισε "Ανεπαρκής"
Τέλος_αν
! Συνέχεια
```

```
Αν Β > 18.5 τότε
    Εμφάνισε "Άριστα"
Αλλιώς_αν Β > 17.5 τότε
    Εμφάνισε "Λίαν Καλώς"
Αλλιώς_αν Β > 13.5 τότε
    Εμφάνισε "Καλώς"
Αλλιώς_αν Β >= 10 τότε
    Εμφάνισε "Επαρκής"
Αλλιώς
    Εμφάνισε "Ανεπαρκής"
Τέλος_αν
! Συνέχεια
```

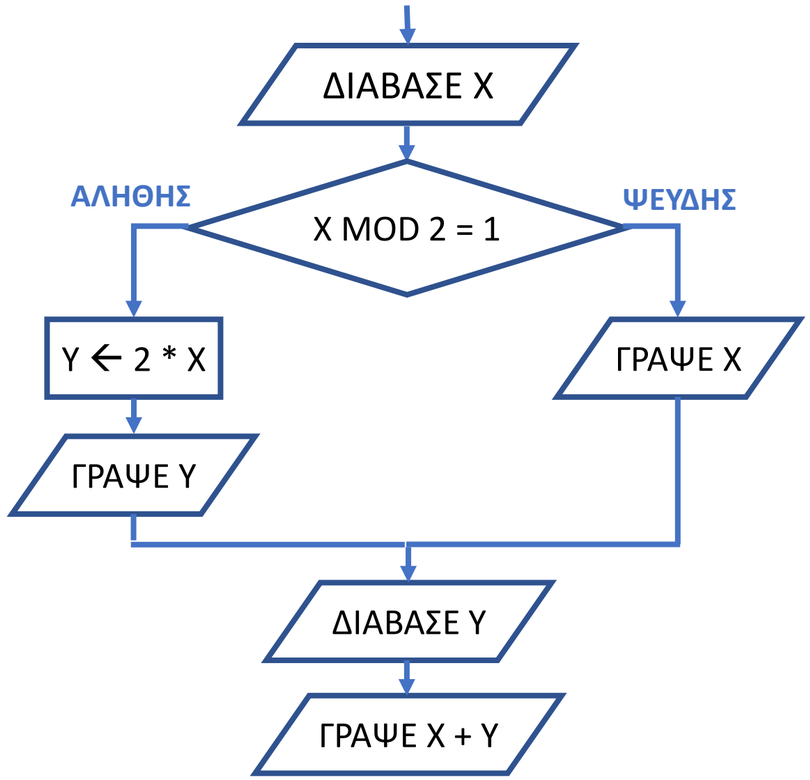
Εάν π.χ. Β = 20 θα εμφανίσει τα πάντα

# Άσκηση 7 / σελίδα 256

Να μετατραπεί το παρακάτω αδόμητο τμήμα προγράμματος, σε αντίστοιχο που να ακολουθεί τις αρχές του δομημένου προγραμματισμού.

<u>Υπόδειξη</u>: Αρχικά σχηματίστε το διάγραμμα ροής.

```
            ΔΙΑΒΑΣΕ Χ
            ΑΝ Χ MOD 2 = 1 ΤΟΤΕ
               ΠΗΓΑΙΝΕ ΣΤΟ ΣΗΜΕΙΟ Α
            ΤΕΛΟΣ_ΑΝ
            ΑΝ Χ MOD 2 = 0 ΤΟΤΕ
               ΓΡΑΨΕ Χ
            ΤΕΛΟΣ_ΑΝ
            ΠΗΓΑΙΝΕ ΣΤΟ ΣΗΜΕΙΟ Β
ΣΗΜΕΙΟ Α:   Υ ← 2 * Χ
            ΓΡΑΨΕ Υ
ΣΗΜΕΙΟ Β:   ΔΙΑΒΑΣΕ Υ
            ΓΡΑΨΕ Χ + Υ
```




```
ΔΙΑΒΑΣΕ Χ
ΑΝ Χ MOD 2 = 1 ΤΟΤΕ
   Υ ← 2 * Χ
   ΓΡΑΨΕ Υ
ΑΛΛΙΩΣ
   ΓΡΑΨΕ Χ
ΤΕΛΟΣ_ΑΝ
ΔΙΑΒΑΣΕ Υ
ΓΡΑΨΕ Χ + Υ
```

# Άσκηση 10 / σελίδα 257

(2019_A5) Δίνεται ο παρακάτω αλγόριθμος πολλαπλασιασμού αλά ρωσικά δύο θετικών ακεραίων αριθμών M1 και M2 σε φυσική γλώσσα κατά βήματα:

Βήμα 1 Θέσε P = 0

Βήμα 2 Αν M2 > 0, τότε πήγαινε στο Βήμα 3, αλλιώς πήγαινε στο Βήμα 7

Βήμα 3 Αν ο M2 είναι περιττός, τότε θέσε P = P + M1

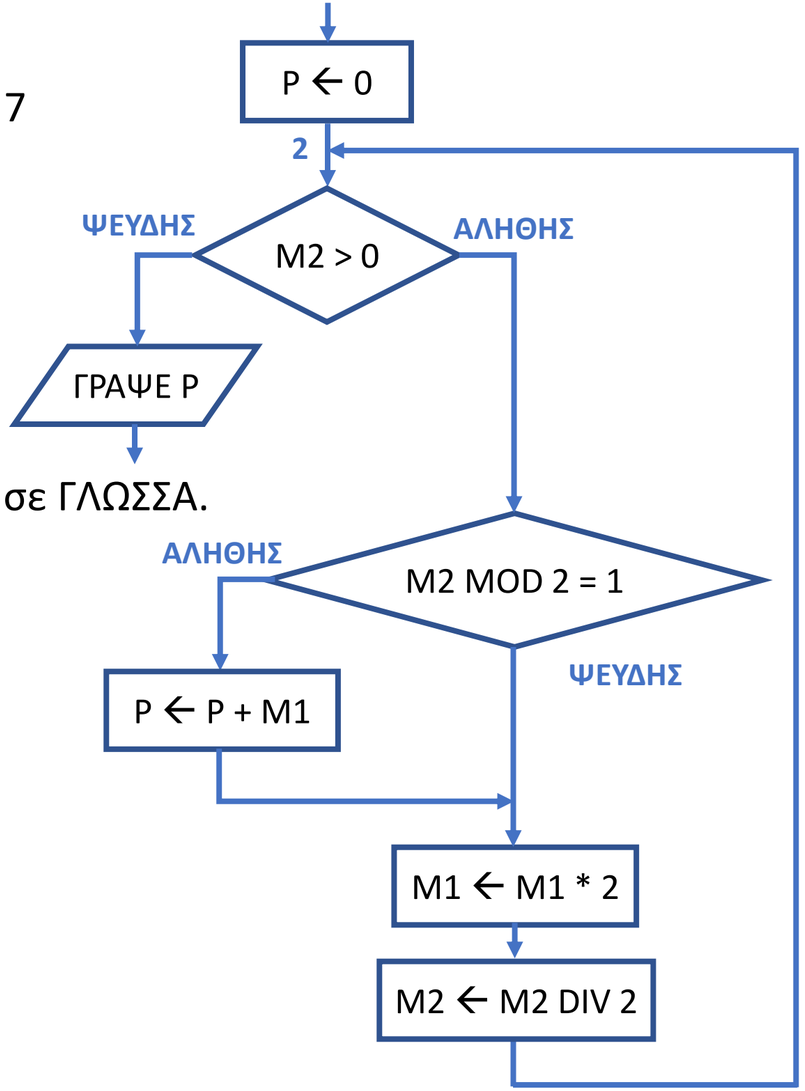
Βήμα 4 Θέσε M1 = M1 * 2

Βήμα 5 Θέσε M2 = M2 / 2 (θεώρησε μόνο το ακέραιο μέρος)

Βήμα 6 Πήγαινε στο Βήμα 2

Βήμα 7 Τύπωσε τον P

Να γράψετε στο τετράδιό σας την κωδικοποίηση των παραπάνω βημάτων σε ΓΛΩΣΣΑ.




P ← 0

**Όσο** M2 > 0 **επανάλαβε**

  **Αν** M2 **mod** 2 = 1 **τότε**

    P ← P + M1

  **Τέλος_αν**

  M1 ← M1 * 2

  M2 ← **A_M**(M2 / 2)

**Τέλος_επανάληψης**

**Εκτύπωσε** P

# Ερωτήσεις θεωρίας / σελίδες 355 – 359

**21.** Τι είναι η απλά συνδεδεμένη λίστα;

**22.** Πως προσπελαύνουμε τους κόμβους μίας απλά συνδεδεμένης λίστας;

**23.** Τι είδους μεταβλητή είναι η διεύθυνση ενός κόμβου της λίστας;

**24.** Πως γίνεται η πρόσβαση στους κόμβους μιας συνδεδεμένης λίστας;

**25.** Τι είναι η διπλά συνδεδεμένη λίστα;

---

**26.** Ποιες είναι οι διαφορές της Λίστας σε σχέση με τον Πίνακα;

**27.** Ποιες είναι οι βασικές πράξεις των συνδεδεμένων λιστών:

**28.** Πλεονεκτήματα – Μειονεκτήματα λίστας και πίνακα:

**29.** Τι είναι η δομή δένδρο;

**33.** Τι ονομάζεται διατεταγμένο δένδρα;

---

**37.** Τι είναι τα δυαδικά δένδρα;

**38.** Τι είναι τα δυαδικά δένδρα αναζήτησης;

**41.** Τι είναι η δομή γράφος;

**42.** Ποιες μορφές μπορεί να έχει ένας γράφος;

**44.** Σε τι διαφέρει ένας γράφος από ένα δένδρο;

# Ενότητα 5

**<u>Σωστό – Λάθος</u>**

**1 – 13 (χωρίς την 3) / σελίδα 251**

**<u>Ασκήσεις</u>**

**2, 4 / σελίδα 255**

**6 / σελίδα 256**

**8, 9 / σελίδα 257**

**Θα ξεκινήσουμε με τεστ στις ερωτήσεις της προηγούμενης σελίδας**