

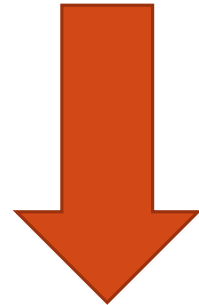
ΕΠΙΛΥΣΗ ΠΡΟΒΛΗΜΑΤΟΣ ΜΕ ΗΥ:

➤ 3 ΣΤΑΔΙΑ

ΑΚΡΙΒΗΣ ΠΡΟΣΔΙΟΡΙΣΜΟΣ
ΠΡΟΒΛΗΜΑΤΟΣ



ΑΝΑΠΤΥΞΗ ΑΛΓΟΡΙΘΜΟΥ



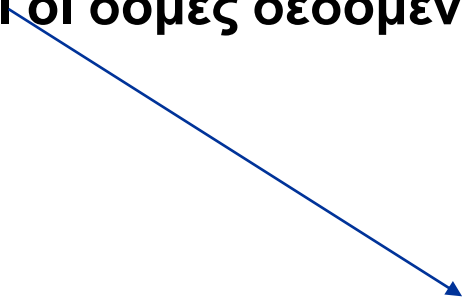
ΔΙΑΤΥΠΩΣΗ ΑΛΓΟΡΙΘΜΟΥ ΣΕ ΜΟΡΦΗ ΚΑΤΑΝΟΗΤΗ ΑΠΟ
ΥΠΟΛΟΓΙΣΤΗ

Προγραμματισμός ονομάζεται η διατύπωση του αλγορίθμου σε μορφή κατανοητή από τον υπολογιστή

Το πρόγραμμα ΔΕΝ είναι απλά οι εντολές του αλγορίθμου σε μορφή κατανοητή από τον ΗΥ.

Βασικά του στοιχεία:

Τα δεδομένα και οι δομές δεδομένων



Εδώ ενεργούν οι εντολές!!!

Οι γλώσσες προγραμματισμού αναπτύχθηκαν με σκοπό την επικοινωνία ανθρώπου και υπολογιστή

- Το Υλικό μπορεί μόνο
 - ✓ Να αποθηκεύει και να ανακτά ακολουθίες δυαδικών ψηφίων
 - ✓ Να κάνει στοιχειώδεις αριθμητικές πράξεις

Το οποίο σημαίνει ότι ο ΗΥ γίνεται ‘έξυπνος’ αφού προγραμματιστεί!!!

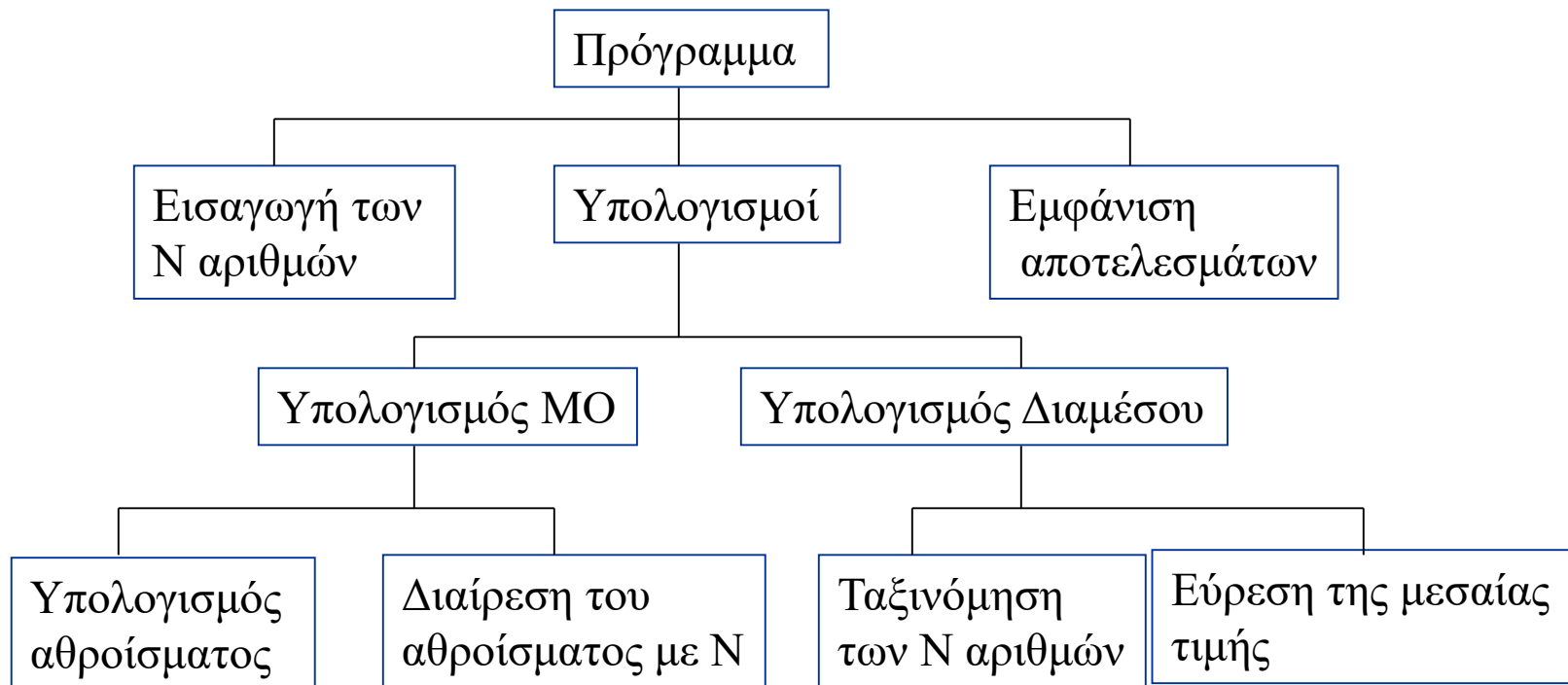
Τεχνικές Σχεδίασης/Ανάπτυξης προγράμματος

Ιεραρχική σχεδίαση

Τεχνική «Από επάνω προς τα κάτω»

Διάσπαση του προβλήματος σε απλούστερα υποπροβλήματα

Π.χ. Υπολογισμός του Μ.Ο. και της Διαμέσου N αριθμών:



Τεχνικές Σχεδίασης/Ανάπτυξης προγράμματος

Η Ιεραρχική Σχεδίαση υλοποιείται με τον Τμηματικό Προγραμματισμό.

Τμηματικός προγραμματισμός

- Αναλύω το πρόβλημα σε υποπροβλήματα
- Κάθε υποπρόβλημα είναι ξεχωριστή ενότητα του προγράμματος
(module)
- Κάθε ενότητα γράφεται ξεχωριστά από τα υπόλοιπα τμήματα του προγράμματος.

Τεχνικές Σχεδίασης/Ανάπτυξης προγράμματος

Η Ιεραρχική Σχεδίαση υλοποιείται με τον Τμηματικό Προγραμματισμό.

Τμηματικός προγραμματισμός

- Διευκολύνει τη δημιουργία του προγράμματος
- Μειώνει τα λάθη
- Επιτρέπει την ευκολότερη κατανόηση και συντήρηση του προγράμματος

Δομημένος Προγραμματισμός

- Παρουσιάστηκε στα μέσα του 1960
- Έχει επικρατήσει σε όλες τις σύγχρονες γλώσσες προγραμματισμού
- Στόχος ήταν να περιοριστεί η χρήση της εντολής GOTO.
- Ενθαρρύνει την δημιουργία προγραμμάτων σε επιμέρους τμήματα
- Εμπεριέχει τόσο την Ιεραρχική σχεδίαση όσο και τον τμηματικό προγραμματισμό.

Εντολή GOTO:

- Επιτρέπει την αλλαγή της ροής του προγράμματος σε μια άλλη εντολή (και όχι στη επόμενη!)

Χρήση της GoTo... το αποτέλεσμα σπαγγέτι!

```
Αν  $x > 0$  τότε goto 1
Αν  $x = 0$  τότε goto 2
Γράψε 'Αρνητικός'
1: Γράψε 'Θετικός'
GoTo 3
2: Γράψε 'Μηδέν'
3: Γράψε 'Τέλος'
```

```
Αν  $x > 0$  τότε
    γράψε 'θετικός'
Αλλιώς_αν  $x = 0$  τότε
    γράψε 'Μηδέν'
Αλλιώς
    Γράψε 'Αρνητικός'
Τέλος_αν
Γράψε τέλος
```

Βασικά χαρακτηριστικά Δομημένου Προγραμματισμού

- ✓ Χρήση των δομών: Ακολουθίας, Επιλογής, Επανάληψης
- ✓ Όχι της εντολής GOTO
- ✓ Κάθε πρόγραμμα ή ενότητα προγράμματος έχει ένα σημείο εισόδου και ένα σημείο εξόδου
- ✓ Εμπεριέχει την ιεραρχική σχεδίαση και τον τμηματικό προγραμματισμό

Πλεονεκτήματα δομημένου προγραμματισμού

- ✓ Απλούστερα προγράμματα
- ✓ Άμεση μετατροπή των αλγορίθμων σε προγράμματα
- ✓ Ευκολία στην ανάλυση του προγράμματος σε τμήματα
- ✓ Περιορισμός λαθών
- ✓ Ευκολία στην ανάγνωση και κατανόηση του προγράμματος από τρίτους
- ✓ Ευκολότερη διόρθωση και συντήρηση