

# ΚΕΦΑΛΑΙΟ 7<sup>ο</sup>

**ΠΡΟΓΡΑΜΜΑ** : Το πρόγραμμα αποτελείται από μια σειρά οδηγιών, που ονομάζονται εντολές, για την εκτέλεση τέτοιου είδους πράξεων, καθώς επίσης και από ένα σύνολο πρόσθετων οδηγιών ελέγχου, που επηρεάζουν τη σειρά με την οποία εκτελούνται οι εντολές.

## ΚΑΤΗΓΟΡΙΕΣ ΓΛΩΣΣΩΝ ΠΡΟΓΡΑΜΜΑΤΙΣΜΟΥ

### 1. ΓΛΩΣΣΑ ΜΗΧΑΝΗΣ

Επειδή ο κάθε τύπος ΚΜΕ έχει διαφορετικό ρεπερτόριο εντολών, τα προγράμματα που εκτελεί πρέπει να είναι διατυπωμένα στη δική της γλώσσα μηχανής. Αυτά τα προγράμματα αποτελούνται από ακολουθίες 0 και 1.

Οι εντολές που αναγνωρίζει μια τυπική ΚΜΕ ανήκουν σε μια από τις πιο κάτω κατηγορίες:

1. εντολές μεταφοράς δεδομένων μεταξύ της κεντρικής μνήμης και των καταχωρητών της ΚΜΕ
2. εντολές μεταφοράς δεδομένων μεταξύ των καταχωρητών
3. εντολές αριθμητικών πράξεων
4. εντολές λογικών πράξεων
5. εντολές ελέγχου της ροής εκτέλεσης των εντολών
6. διάφορες βοηθητικές εντολές. **Κωδικός λειτουργίας** **Τελεστής**

○ **κωδικός λειτουργίας** προσδιορίζει τη λειτουργία της εντολής, ○ **τελεστής** αφορά τα δεδομένα στα οποία δρα η εντολή ή τη διεύθυνση στην οποία βρίσκονται αποθηκευμένα.

**2. ΣΥΜΒΟΛΙΚΕΣ ΓΛΩΣΣΕΣ** Μια πρώτη προσπάθεια για τη διευκόλυνση της γραφής προγραμμάτων ήταν η γραφή τους σε συμβολική γλώσσα (assembly). Στις συμβολικές γλώσσες, σε κάθε εντολή της γλώσσας μηχανής αντιστοιχίζεται μια μνημονική λέξη η οποία θυμίζει το σκοπό της εντολής. Επί πλέον δίνεται η δυνατότητα να χρησιμοποιούνται, στη θέση αριθμών, **ονόματα που αντιπροσωπεύουν σταθερές ή διευθύνσεις μνήμης.**

Για την μετατροπή της ΣΥΜΒΟΛΙΚΗΣ ΓΛΩΣΣΑΣ σε γλώσσα μηχανής χρησιμοποιούμε ένα ειδικό πρόγραμμα που λέγεται ΣΥΜΒΟΛΟΜΕΤΑΦΡΑΣΤΗΣ.

**3.ΓΛΩΣΣΕΣ ΥΨΗΛΟΥ ΕΠΙΠΕΔΟΥ** Οι γλώσσες υψηλού επιπέδου χρησιμοποιούν ως εντολές απλές λέξεις της αγγλικής γλώσσας ακολουθώντας αυστηρούς κανόνες σύνταξης, οι οποίες μεταφράζονται από τον ίδιο τον υπολογιστή σε εντολές σε γλώσσα μηχανής

ΟΝΟΜΑΣΙΑ ΓΛΩΣΣΑΣ	ΤΟΜΕΑΣ ΕΦΑΡΜΟΓΩΝ	ΕΙΔΟΣ ΠΡΟΓΡΑΜΜΑΤΙΣΜΟΥ
1. COBOL	ΟΙΚΟΝΟΜΙΚΕΣ ΕΦΑΡΜΟΓΕΣ	ΔΙΑΔΙΚΑΣΙΑΚΟΣ
2. BASIC, PASCAL, ALGOL	ΓΕΝΙΚΕΣ ΕΦΑΡΜΟΓΕΣ	ΔΙΑΔΙΚΑΣΙΑΚΟΣ
3. FORTRAN	ΕΠΙΣΤΗΜΟΝΙΚΕΣ ΕΦΑΡΜΟΓΕΣ	ΔΙΑΔΙΚΑΣΙΑΚΟΣ
4. PROLOG, LISP	ΤΕΧΝΗΤΗΣ ΝΟΗΜΟΣΥΝΗΣ	
5. C, C++	ΠΡΟΓΡΑΜΜΑΤΙΣΜΟΣ ΣΥΣΤΗΜΑΤΩΝ	ΑΝΤΙΚΕΙΜΕΝΟΣΤΡΑΦΗΣ (C++ )
6. JAVA	ΠΡΟΓΡΑΜΜΑΤΙΣΜΟΣ ΔΙΑΔΙΚΤΥΟ	ΑΝΤΙΚΕΙΜΕΝΟΣΤΡΑΦΗΣ
7. LOGO	ΕΚΠΑΙΔΕΥΤΙΚΗ ΓΛΩΣΣΑ	
8. SQL	ΒΑΣΕΙΣ ΔΕΔΟΜΕΝΩΝ (ΓΛΩΣΣΑ ΕΡΩΤΟΑΠΑΝΤΗΣΕΩΝ )	
9. SMALLTALK	ΛΕΙΤΟΥΡΓΙΚΟ ΣΥΣΤΗΜΑ	ΑΝΤΙΚΕΙΜΕΝΟΣΤΡΑΦΗΣ

## Πλεονεκτήματα των γλωσσών υψηλού επιπέδου

Στα πλεονεκτήματα των γλωσσών προγραμματισμού υψηλού επιπέδου σε σχέση με τις συμβολικές μπορούν να αναφερθούν:

- ▶ Ο φυσικότερος και πιο "ανθρώπινος" τρόπος έκφρασης των προβλημάτων.
- ▶ Τα προγράμματα σε γλώσσα υψηλού επιπέδου είναι πιο κοντά στα προβλήματα που επιλύουν.
- ▶ Η ανεξαρτησία από τον τύπο του υπολογιστή. Προγράμματα σε μία γλώσσα υψηλού επιπέδου μπορούν να εκτελεστούν σε οποιονδήποτε υπολογιστή με ελάχιστες ή καθόλου μετατροπές. Η δυνατότητα της **μεταφερσιμότητας των προγραμμάτων** είναι σημαντικό προσόν.
- ▶ Η ευκολία της εκμάθησης και εκπαίδευσης ως απόρροια των προηγούμενων.
- ▶ Η διόρθωση λαθών και η συντήρηση προγραμμάτων σε γλώσσα υψηλού επιπέδου είναι πολύ ευκολότερο έργο.

### Εξάρτηση των γλωσσών από το σκοπό

Οι διάφορες γλώσσες ανάλογα με το σκοπό που εξυπηρετούν, ή ακριβέστερα σύμφωνα με το σκοπό για τον οποίο αρχικά σχεδιάστηκαν, χωρίζονται σε δύο γενικές κατηγορίες, στις **ειδικού και στις γενικού σκοπού γλώσσες**.

Στην κατηγορία των γλωσσών ειδικού σκοπού ανήκουν εκείνες οι οποίες είναι προσανατολισμένες σε μια συγκεκριμένη κατηγορία εφαρμογών.

### Γλώσσες ειδικού σκοπού

Για παράδειγμα:

- ▶ Η COBOL είναι προσανατολισμένη στην επίλυση εμπορικών προβλημάτων. Αυτή η κατηγορία προβλημάτων χαρακτηρίζεται από την αποθήκευση και διαχείριση μεγάλου όγκου δεδομένων με σχετικά περιορισμένες απαιτήσεις αριθμητικών πράξεων. Ως αποτέλεσμα η COBOL δίνει μεγάλη βαρύτητα στα θέματα περιγραφής και αποθήκευσης δεδομένων, ενώ μειονεκτεί στο θέμα της αποδοτικής εκτέλεσης μεγάλου αριθμού αριθμητικών πράξεων.
- ▶ Η FORTRAN είναι προσανατολισμένη στις επιστημονικές εφαρμογές, όπου κυρίαρχο ρόλο παίζει η ταχύτητα εκτέλεσης μεγάλου πλήθους αριθμητικών πράξεων πάνω σε μικρό σχετικά όγκο δεδομένων.
- ▶ Η LISP είναι προσανατολισμένη στη διαχείριση συμβόλων, γεγονός που την καθιστά κατάλληλη για εφαρμογές τεχνητής νοημοσύνης.

**Γλώσσες γενικού σκοπού**, περιλαμβάνει γλώσσες που μπορούν να χρησιμοποιηθούν σε διαφορετικές περιοχές εφαρμογών με εξίσου καλά αποτελέσματα. Για παράδειγμα, τέτοιες γλώσσες είναι η C και η Java.

# ΜΕΤΑΦΡΑΣΤΕΣ

1. **ΣΥΜΒΟΛΟΜΕΤΑΦΡΑΣΤΗΣ** Μετατρέπει την ΣΥΜΒΟΛΙΚΗ γλώσσα σε γλώσσα μηχανής .
2. **ΜΕΤΑΓΛΩΤΙΣΤΗΣ** Ο μεταγλωττιστής 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.
3. **ΔΙΕΡΜΗΝΕΥΤΗΣ** Ο διερμηνευτής διαβάζει μία προς μία τις εντολές του αρχικού προγράμματος και για κάθε μια εκτελεί αμέσως μια ισοδύναμη ακολουθία εντολών μηχανής.

## ΣΤΑΔΙΑ ΜΕΤΑΓΛΩΤΙΣΗΣ

1. Το αρχικό πρόγραμμα λέγεται **πηγαίο πρόγραμμα** (source), ενώ το πρόγραμμα που παράγεται από το μεταγλωττιστή λέγεται **αντικείμενο πρόγραμμα** (object).
2. Το αντικείμενο πρόγραμμα είναι μεν σε μορφή κατανοητή από τον υπολογιστή, αλλά συνήθως δεν είναι σε θέση να εκτελεστεί. Χρειάζεται να συμπληρωθεί και να συνδεθεί με άλλα τμήματα προγράμματος απαραίτητα για την εκτέλεσή του, τμήματα που είτε τα γράφει ο προγραμματιστής είτε βρίσκονται στις **βιβλιοθήκες** (libraries) της γλώσσας. Το πρόγραμμα που επιτρέπει αυτή τη σύνδεση ονομάζεται **συνδέτης - φορτωτής** (linker-loader). Το αποτέλεσμα του συνδέτη είναι η παραγωγή του εκτελέσιμου προγράμματος (executable), το οποίο είναι το τελικό πρόγραμμα που εκτελείται από τον υπολογιστή. Για το λόγο αυτό η συνολική διαδικασία αποκαλείται μεταγλώττιση και σύνδεση.



# ΑΡΧΕΣ ΚΑΤΑΣΚΕΥΗΣ ΛΟΓΙΣΜΙΚΟΥ

## 1. Ιεραρχική Σχεδίαση Προγράμματος

Η ιεραρχική σχεδίαση ή ιεραρχικός προγραμματισμός χρησιμοποιεί τη στρατηγική της συνεχούς διαίρεσης του προβλήματος σε υποπροβλήματα.

## 2. Τμηματικός προγραμματισμός

Η ιεραρχική σχεδίαση προγράμματος υλοποιείται με τον τμηματικό προγραμματισμό. Μετά την ανάλυση του προβλήματος σε αντίστοιχα υποπροβλήματα, κάθε υποπρόβλημα αποτελεί ανεξάρτητη ενότητα (module), που γράφεται ξεχωριστά από τα υπόλοιπα τμήματα προγράμματος. Κάθε υποπρόβλημα υλοποιείται με ένα ανεξάρτητο υποπρόγραμμα ή υπορουτίνα.

## 3. Δομημένος Προγραμματισμός

Βασική τεχνική του δομημένου προγραμματισμού είναι η διάσπαση των λειτουργιών του προγράμματος σε άλλες, απλούστερες και ανεξάρτητες κατά το δυνατό, επιμέρους λειτουργίες και στη συνέχεια η υλοποίησή τους με καθορισμένους τύπους δομών ελέγχου.

Οι δομές αυτές είναι:

**Η διαδοχή (ακολουθία).** Οι εντολές βρίσκονται σε ακολουθία και εκτελούνται με τη σειρά που είναι γραμμένες.

**Η επιλογή.** Η εκτέλεση των εντολών εξαρτάται από την τιμή αλήθειας της συνθήκης.

**Η επανάληψη.** Οι εντολές επαναλαμβάνονται, εφόσον αληθεύει ή δεν αληθεύει η συνθήκη. Όλες οι σύγχρονες γλώσσες προγραμματισμού εφαρμόζουν την αρχή του δομημένου προγραμματισμού.

# ΠΡΟΤΥΠΑ ΠΡΟΓΡΑΜΜΑΤΙΣΜΟΥ

## 1. Διαδικαστικός ή Προστακτικός Προγραμματισμός

Σύμφωνα με αυτό το πρότυπο, το πρόγραμμα αποτελείται από δύο ξεχωριστά δομικά στοιχεία:

- ▶ Από **εντολές** που περιγράφουν βήμα βήμα τη διαδικασία επίλυσης του προβλήματος
- ▶ Από **δομές δεδομένων**, στις οποίες αποθηκεύονται τα δεδομένα του προβλήματος τα οποία χειρίζονται οι εντολές.

Οι εντολές που, κατ' ελάχιστο, περιέχονται σε κάθε διαδικαστική γλώσσα προγραμματισμού είναι:

- ▶ Εντολές ανάθεσης
- ▶ Εντολές συνθήκης
- ▶ Εντολές επανάληψης
- ▶ Εντολές για είσοδο και έξοδο στοιχείων

## 2. Αντικειμενοστραφής Προγραμματισμός

Στην αντικειμενοστρεφή προσέγγιση προγραμματισμού, **το αντικείμενο (object)** είναι ένα προγραμματιστικό «πακέτο», το οποίο αποτελείται από **διαδικασίες και δεδομένα** σχετιζόμενα μεταξύ τους. Αυτές οι διαδικασίες ονομάζονται **μέθοδοι (methods)**, ενώ τα δεδομένα ονομάζονται **ιδιότητες (properties)**. Οι μέθοδοι σε πολλές περιπτώσεις είναι γραμμένες σε κάποια διαδικαστική γλώσσα.

Το πρόγραμμα, σε αυτήν την προσέγγιση, αποτελείται από **αντικείμενα**, τα οποία μπορούν να αλληλεπιδρούν μεταξύ τους με **μηνύματα (messages)**. Τα μηνύματα ενεργοποιούν τις μεθόδους των αντικειμένων, που με τη σειρά τους μπορεί να στείλουν άλλα μηνύματα. Για παράδειγμα, σε ένα πρόγραμμα γεωμετρίας ένα αντικείμενο θα μπορούσε να είναι ένα τρίγωνο.

Ως ιδιότητες θα μπορούσε να έχει τα μήκη των πλευρών του, ενώ ως μεθόδους τον υπολογισμό της περιμέτρου και του εμβαδού.

Πχ java ,C++

### 3. Λογικός Προγραμματισμός

Ο λογικός προγραμματισμός (logic programming) είναι επηρεασμένος και εμπνευσμένος από τη μαθηματική λογική. Ένα πρόγραμμα γραμμένο σε μια γλώσσα λογικού προγραμματισμού, όπως η Prolog, είναι ένα σύνολο από λογικές προτάσεις και από ένα μηχανισμό εξαγωγής συμπερασμάτων, μέσω του οποίου μπορούμε να υποβάλουμε ερωτήσεις στο πρόγραμμα και αυτό να μας απαντήσει αν αληθεύουν ή όχι.

### 4. Συναρτησιακός Προγραμματισμός

Ο συναρτησιακός προγραμματισμός (functional programming), που συχνά ονομάζεται και εφαρμοστικός προγραμματισμός (applicative programming), έχει ως βάση τη συνάρτηση με τη μαθηματική της έννοια, δηλαδή αυτή της απεικόνισης από ένα πεδίο ορισμού σε ένα πεδίο τιμών.

## ΜΕΘΟΔΟΣ ΑΝΑΠΤΥΞΗΣ ΛΟΓΙΣΜΙΚΟΥ

Τα βήματα αυτής της μεθόδου είναι:

- α) Προσδιορισμός των απαιτήσεων του προβλήματος.
- β) Ανάλυση του προβλήματος.
- γ) Σχεδιασμός αλγόριθμου για την επίλυση του προβλήματος.
- δ) Υλοποίηση του αλγόριθμου.
- ε) Έλεγχος και επαλήθευση του τελικού προγράμματος.
- στ) Συντήρηση και ενημέρωση του προγράμματος.
- ζ) Τεκμηρίωση.

#### **A. Προσδιορισμός των απαιτήσεων του προβλήματος (Ανάλυση προβλήματος)**

Ο προσδιορισμός των απαιτήσεων του προβλήματος είναι ένα στάδιο στο οποίο θα πρέπει να καθοριστεί με σαφήνεια το πρόβλημα και να υπάρξει ξεκάθαρη αντίληψη του είδους των απαιτήσεων για τη λύση του.

#### **B. Ανάλυση του προβλήματος.**

Η ανάλυση του προβλήματος περιλαμβάνει:

1. Τον καθορισμό των «εισόδων», δηλαδή των δεδομένων με τα οποία θα δουλέψουμε (inputs)
2. Τον καθορισμό των «εξόδων», δηλαδή των απαιτούμενων αποτελεσμάτων (outputs)
3. Την αναζήτηση άλλων απαιτήσεων ή περιορισμών.

### **Γ) Σχεδιασμός αλγόριθμου για την επίλυση του προβλήματος.**

Ο αλγόριθμος είναι μια σειρά από βήματα, τα οποία πρέπει να ακολουθήσουμε για να επιλύσουμε ένα συγκεκριμένο πρόβλημα. Τα βήματα πρέπει να είναι σαφή και να υπάρχει συγκεκριμένο σημείο περάτωσης της διαδικασίας.

### **Δ) Υλοποίηση του αλγόριθμου.**

Η υλοποίηση του αλγόριθμου αφορά τη συγγραφή του προγράμματος. Αυτό σημαίνει ότι κάθε βήμα του αλγόριθμου πρέπει να μετατραπεί σε μια ή περισσότερες εντολές κάποιας γλώσσας προγραμματισμού. Ο δομημένος προγραμματισμός είναι μια πειθαρχημένη προσέγγιση στη διαδικασία συγγραφής, που οδηγεί στη δημιουργία προγραμμάτων, τα οποία όχι μόνο είναι εύκολο να διαβαστούν και να γίνουν κατανοητά, αλλά και να περιέχουν τα λιγότερα λάθη .

### **Ε) Έλεγχος και επαλήθευση του τελικού προγράμματος.**

Αφορά την εκτέλεση του προγράμματος για διάφορα -σωστά επιλεγμένα- σύνολα τιμών εισόδου και τον έλεγχο των αντίστοιχων εξόδων

### **ΣΤ) Συντήρηση και ενημέρωση του προγράμματος.**

Δύο είναι οι βασικοί λόγοι:

- ▶ Για να διορθωθούν λάθη τα οποία εντοπίστηκαν, αφού το πρόγραμμα παραδόθηκε για χρήση, κατά τη διάρκεια δηλαδή χρήσης του προγράμματος
- ▶ Για να γίνουν αλλαγές που προέκυψαν με την πάροδο του χρόνου, λόγω αλλαγών στο περιβάλλον χρήσης του προγράμματος.

### **Ζ) Τεκμηρίωση.**

Η τεκμηρίωση αφορά το σύνολο των εγγράφων που προκύπτουν καθ' όλη τη διάρκεια της ανάπτυξης του προγράμματος και βοηθούν το έργο της συντήρησης του αλλά και τη χρήση του.

Διακρίνουμε την [Εξωτερική](#) και την [Εσωτερική Τεκμηρίωση](#) .



# ΤΡΟΠΟΙ ΠΑΡΑΣΤΑΣΗΣ ΑΛΓΟΡΙΘΜΩΝ

## α. Με φυσική γλώσσα

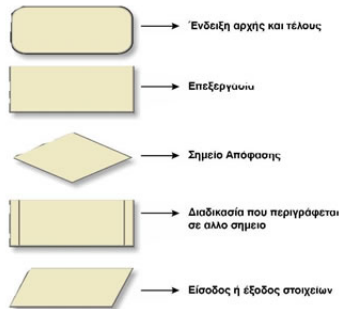
Σε αυτήν ο αλγόριθμος περιγράφεται σαν ένα απλό κείμενο

## β. Με ψευδοκώδικα

○ ψευδοκώδικας μοιάζει με την περιγραφή του αλγόριθμου σε φυσική γλώσσα, με τη διαφορά ότι χρησιμοποιεί συγκεκριμένες δομές για την περιγραφή του αλγόριθμου, που είναι πολύ κοντά στις εντολές των γλωσσών υψηλού επιπέδου.

## γ. Με Λογικά διαγράμματα

Τα λογικά διαγράμματα είναι ένας γραφικός τρόπος παρουσίασης των βημάτων του αλγόριθμου. Υπάρχει ένα σύνολο από τυποποιημένα σχήματα που παριστάνουν τις διάφορες δομές ενός αλγόριθμου



## ΠΡΟΓΡΑΜΜΑΤΙΣΤΙΚΑ ΠΕΡΙΒΑΛΛΟΝΤΑ

Ένα προγραμματιστικό περιβάλλον συνήθως αποτελείται από εργαλεία όπως

**A. Ένα συντάκτη κειμένων**, με τον οποίο γράφεται το πηγαίο πρόγραμμα.

**B. Μεταφραστικά προγράμματα** (assembler, compiler, interpreter).

**Γ. Εργαλεία εντοπισμού λαθών** (debugger).

## ΟΠΤΙΚΟΣ ΠΡΟΓΡΑΜΜΑΤΙΣΜΟΣ

Η αξιοποίηση γραφικών μεθόδων στην ανάπτυξη προγραμμάτων αναφέρεται ως οπτικός προγραμματισμός (visual programming). Με αυτόν η παραγωγή του κώδικα επιτυγχάνεται με γραφικές μεθόδους,