

## 6ο Κεφάλαιο – Εισαγωγή στον προγραμματισμό.

**Ερώτηση 6.1:** Ποια είναι τα τρία στάδια επίλυσης ενός προβλήματος και με ποιο από αυτά 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.

Ο προγραμματισμός είναι αυτός που δίνει την εντύπωση ότι, οι υπολογιστές είναι έξυπνες μηχανές που επιλύουν τα πολύπλοκα προβλήματα.

**Ερώτηση 6.2:** Ποιος και πότε διατύπωσε τις αρχές λειτουργίας των υπολογιστών;

Ενώ λοιπόν το υλικό (hardware) των υπολογιστών βελτιώνεται, τελειοποιείται και ταυτόχρονα παρέχει νέες δυνατότητες επεξεργασίας, οι βασικές αρχές λειτουργίας των υπολογιστών που διατυπώθηκαν το μακρινό 1945 από τον Φον Νούμαν, δεν άλλαξαν πρακτικά καθόλου.

**Ερώτηση 6.3:** Τι γνωρίζετε για τη γλώσσα μηχανής;

*Ένα πρόγραμμα σε γλώσσα μηχανής είναι μια ακολουθία δυαδικών ψηφίων, που αποτελούν εντολές προς τον επεξεργαστή για στοιχειώδεις λειτουργίες.*

Οι εντολές ενός προγράμματος και σήμερα μετατρέπονται σε ακολουθίες που αποτελούνται από 0 και 1, τις εντολές σε γλώσσα μηχανής, όπως ονομάζονται, οι οποίες εκτελούνται από τον υπολογιστή.

**Ερώτηση 6.4:** Τι γνωρίζετε για τις γλώσσες χαμηλού επιπέδου (κυρίως τα μειονεκτήματα);

*Οι εντολές σε συμβολική γλώσσα αποτελούνται από συμβολικά ονόματα που αντιστοιχούν σε εντολές της γλώσσας μηχανής.*

Τα προγράμματα που είναι γραμμένα σε μια γλώσσα χαμηλού επιπέδου, θα πρέπει να μεταφραστούν σε γλώσσα μηχανής .

Το έργο της μετάφρασης το αναλαμβάνει ένα ειδικό πρόγραμμα, ο συμβολομεταφραστής (assembler).

Μειονεκτήματα γλωσσών χαμηλού επιπέδου :

- Παρέμεναν στενά συνδεδεμένες με την αρχιτεκτονική του κάθε υπολογιστή.
- Δεν διέθεταν εντολές πιο σύνθετων λειτουργιών οδηγώντας έτσι σε μακροσκελή προγράμματα, που ήταν δύσκολο να γραφούν και κύρια να συντηρηθούν.
- Τα προγράμματα δεν μπορούν να μεταφερθούν σε άλλον διαφορετικό υπολογιστή, ακόμη και του ίδιου κατασκευαστή.

**Ερώτηση 6.5: Τι γνωρίζετε για την FORTRAN;**

1957 : Η IBM αναπτύσσει την 1<sup>η</sup> γλώσσα υψηλού επιπέδου, την FORTRAN (FORMula TRANslation).

- Η FORTRAN αναπτύχθηκε ως γλώσσα κατάλληλη για την επίλυση μαθηματικών και επιστημονικών προβλημάτων.
- Το πρόγραμμα που γράφεται σε FORTRAN ή σε οποιαδήποτε άλλη γλώσσα υψηλού επιπέδου, μεταφράζεται από τον ίδιο τον υπολογιστή στις ακολουθίες των εντολών της μηχανής με τη βοήθεια ενός ειδικού προγράμματος, που ονομάζεται μεταγλωττιστής
- Ταξινόμηση γλώσσας: Επιστημονικής κατεύθυνσης.

**Ερώτηση 6.6: Τι γνωρίζετε για την COBOL;**

1960: Αναπτύχθηκε μία άλλη γλώσσα, σταθμός στον προγραμματισμό η γλώσσα COBOL. (Common Business Oriented Language -Κοινή γλώσσα προσανατολισμένη στις επιχειρήσεις)

- Η COBOL όπως δηλώνει και το όνομα της είναι κατάλληλη για ανάπτυξη εμπορικών εφαρμογών, και γενικότερα διαχειριστικών εφαρμογών, τομέας όπου η FORTRAN υστερούσε.
- Η COBOL καθιερώθηκε ως πρότυπο και χρησιμοποιήθηκε από πολλές επιχειρήσεις καθώς και από όλη τη δημόσια διοίκηση.
- Η γλώσσα γνώρισε πολλές εκδόσεις και πάρα πολλές εφαρμογές βρίσκονται σε χρήση ακόμη και σήμερα.
- Ταξινόμηση γλώσσας: Εμπορικής κατεύθυνσης.

**Ερώτηση 6.7: Τι γνωρίζετε για τις ALGOL και PL/1;**

1960: ALGOL (ALGOrithmic Language – Αλγοριθμική γλώσσα).

- Μια από τις σημαντικότερες γλώσσες προγραμματισμού με ελάχιστη πρακτική εφαρμογή αλλά που επηρέασε ιδιαίτερα τον προγραμματισμό και τις επόμενες γλώσσες.
- Αναπτύχθηκε από Ευρωπαίους επιστήμονες, αρχικά το 1960, με σκοπό τη δημιουργία γενικής φύσης προγραμμάτων που να μη συνδέονται με συγκεκριμένες εφαρμογές.

Στα μέσα της δεκαετίας του 60 αναπτύχθηκε η γλώσσα PL/1.

- (Programming Language/1 – Γλώσσα Προγραμματισμού υπ' αριθμόν 1)
- Προσπάθησε, χωρίς επιτυχία, να καλύψει όλους τους τομείς του προγραμματισμού, επιστημονικούς και εμπορικούς, αντικαθιστώντας τόσο τη FORTRAN όσο και την COBOL .

### Ερώτηση 6.8: Τι γνωρίζετε για τις LISP και PROLOG;

Στο χώρο της Τεχνητής Νοημοσύνης αναπτύχθηκαν δύο γλώσσες αρκετά διαφορετικές από όλες τις άλλες. Στα μέσα του 60 αναπτύχθηκε στο MIT η LISP (LISt Processor-Επεξεργαστής Λίστας), γλώσσα η οποία προσανατολίζεται σε χειρισμό λιστών από σύμβολα και η PROLOG (PROgramming LOGic –Λογικός Προγραμματισμός) στις αρχές του 70. Οι δύο αυτές γλώσσες χρησιμοποιούνται σε προβλήματα Τεχνητής νοημοσύνης (έμπειρα συστήματα, παιχνίδια, επεξεργασία φυσικών γλωσσώνκ.λπ.).

Ταξινόμηση γλωσσών :

Lisp: Συναρτησιακός προγραμματισμός, Τεχνητής νοημοσύνης.

Prolog : Μη διαδικασιακή, Τεχνητής νοημοσύνης.

### Ερώτηση 6.9: Τι γνωρίζετε για την BASIC;

BASIC

(Beginner's All Purpose Symbolic Instruction Code – Συμβολικός Κώδικας Εντολών Γενικής Χρήσης για Αρχάριους)

- Αρχικά αναπτύχθηκε, όπως δηλώνει και το όνομα της, ως γλώσσα για την εκπαίδευση αρχαρίων στον προγραμματισμό.
- Σχεδιάστηκε για να γράφονται σύντομα προγράμματα, τα οποία εκτελούνται με τη βοήθεια διερμηνευτή (interpreter).
- Η ανάπτυξη όμως των μικροϋπολογιστών και οι συνεχείς εκδόσεις της γλώσσας βοήθησαν στην εξάπλωσή της, τόσο ώστε να γίνει ίσως η δημοφιλέστερη γλώσσα στους προσωπικούς υπολογιστές.
- Η τυποποίηση της δε από τη Microsoft με τις εκδόσεις QuickBasic και κύρια με τη Visual Basic, καθιέρωσε τη γλώσσα ως πρότυπο για ανάπτυξη εφαρμογών σε προσωπικούς υπολογιστές.
- Ταξινόμηση γλώσσας: Γενικής χρήσης.

### Ερώτηση 6.10: Τι γνωρίζετε για την PASCAL;

Η γλώσσα PASCAL (δημιούργημα του καθηγητή Niklaus Wirth) έφερε μεγάλες αλλαγές στον προγραμματισμό.

- Παρουσιάστηκε το 1970 και στηρίχτηκε πάνω στην ALGOL.
- Είναι μία γλώσσα γενικής χρήσης, η οποία είναι κατάλληλη τόσο για την εκπαίδευση όσο και τη δημιουργία ισχυρών προγραμμάτων κάθε τύπου.
- Χαρακτηριστικό της γλώσσας είναι η καταλληλότητα για τη δημιουργία δομημένων προγραμμάτων.
- Η PASCAL γνώρισε και συνεχίζει να γνωρίζει τεράστια εξάπλωση ειδικά στο χώρο των μικροϋπολογιστών και αποτέλεσε τη βάση για την ανάπτυξη άλλων ισχυρότερων γλωσσών όπως η ADA και η Modula-2.
- Ταξινόμηση γλώσσας: Γενικής χρήσης.

### **Ερώτηση 6.11: Τι γνωρίζετε για τις C και C++ ;**

#### **Η γλώσσα C.**

Η C αναπτύχθηκε στα εργαστήρια της εταιρείας BELL και χρησιμοποιήθηκε για την ανάπτυξη του λειτουργικού συστήματος Unix,

- Είναι γλώσσα με ισχυρά χαρακτηριστικά, μερικά από αυτά κοινά με την Pascal κατάλληλη για ανάπτυξη δομημένων εφαρμογών αλλά και με πολλές δυνατότητες γλώσσας χαμηλού επιπέδου.
- Η C εξελίχτηκε στη γλώσσα C++, που είναι αντικειμενοστραφής.
- **Ταξινόμηση γλώσσας:** Προγραμματισμού συστημάτων.

### **Ερώτηση 6.12: Τι γνωρίζετε για την JAVA;**

Τα τελευταία χρόνια χρησιμοποιείται ιδιαίτερα, ειδικά για προγραμματισμό στο Διαδίκτυο (Internet), η **JAVA**.

- Η JAVA είναι μία αντικειμενοστραφής γλώσσα που αναπτύχθηκε από την εταιρεία SUN με σκοπό την ανάπτυξη εφαρμογών, που θα εκτελούνται σε κατακευματισμένα περιβάλλοντα, δηλαδή σε διαφορετικούς υπολογιστές οι οποίοι είναι συνδεδεμένοι στο Διαδίκτυο.
- Τα προγράμματα αυτά μπορούν να εκτελούνται από διαφορετικούς υπολογιστές, προσωπικούς ή μεγάλα συστήματα με διαφορετικά λειτουργικά συστήματα χωρίς αλλαγές.

### **Ερώτηση 6.13: Τι γνωρίζετε για τον οδηγούμενο από το γεγονός προγραμματισμό (object driven programming) και για τον οπτικό προγραμματισμό (visual programming);**

Με τον όρο οπτικό προγραμματισμό εννοούμε τη δυνατότητα να δημιουργούμε γραφικά ολόκληρο το περιβάλλον της εφαρμογής για παράδειγμα τα πλαίσια διαλόγου ή τα μενού.

Με τον όρο οδηγούμενο από το γεγονός εννοούμε τη δυνατότητα να ενεργοποιούνται λειτουργίες του προγράμματος με την εκτέλεση ενός γεγονότος, για παράδειγμα την επιλογή μίας εντολής από ένα μενού ή το κλικ του ποντικιού.

Οι πιο διαδεδομένες γλώσσες προγραμματισμού σε γραφικό περιβάλλον για προσωπικούς υπολογιστές είναι η Visual Basic, η Visual C++ και η Java.

## 6ο Κεφάλαιο – Εισαγωγή στον προγραμματισμό.

### **Ερώτηση 6.14: Ποια είναι τα πλεονεκτήματα των γλωσσών υψηλού επιπέδου ;**

Πλεονεκτήματα των γλωσσών προγραμματισμού υψηλού επιπέδου:

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

### **Ερώτηση 6.15: Τι γνωρίζετε για τις γλώσσες 4<sup>ης</sup> γενιάς;**

Στις γλώσσες αυτές ο χρήστης ενός υπολογιστή έχει τη δυνατότητα, σχετικά εύκολα, να υποβάλει ερωτήσεις στο σύστημα ή να αναπτύσσει εφαρμογές που ανακτούν πληροφορίες από βάσεις δεδομένων και να καθορίζει τον ακριβή τρόπο εμφάνισης αυτών των πληροφοριών.

### **Ερώτηση 6.16: Από ποια στοιχεία προσδιορίζεται μια γλώσσα, φυσική ή τεχνητή;**

Οι γλώσσες προγραμματισμού αναπτύχθηκαν, για να μπορεί ο προγραμματιστής να δίνει τις εντολές που πρέπει να εκτελέσει ο υπολογιστής. Χρησιμοποιούνται δηλαδή για την επικοινωνία του ανθρώπου και της μηχανής, όπως αντίστοιχα οι φυσικές γλώσσες χρησιμοποιούνται για την επικοινωνία μεταξύ των ανθρώπων. Οι γλώσσες προγραμματισμού, που είναι τεχνητές γλώσσες, ακολουθούν τις βασικές έννοιες και αρχές της γλωσσολογίας, επιστήμη που μελετά τις φυσικές γλώσσες.

### **Ερώτηση 6.17: Ποιες είναι οι διαφορές φυσικών και τεχνητών γλωσσών;**

Οι φυσικές γλώσσες εξελίσσονται συνεχώς και αυτό γιατί η γλώσσα χρησιμοποιείται για την επικοινωνία μεταξύ ανθρώπων, που εξελίσσονται και αλλάζουν ανάλογα με τις εποχές και τον κοινωνικό περίγυρο.

Αντίθετα οι τεχνητές γλώσσες χαρακτηρίζονται από στασιμότητα, αφού κατασκευάζονται συνειδητά για ένα συγκεκριμένο σκοπό. Οι γλώσσες προγραμματισμού αλλάζουν σε επίπεδο διαλέκτου (για παράδειγμα GW-Basic και QuickBasic) ή σε επίπεδο επέκτασης(για παράδειγμα Basic και Visual Basic).

## 6ο Κεφάλαιο – Εισαγωγή στον προγραμματισμό.

### **Ερώτηση 6.18: Τι γνωρίζετε για την ιεραρχική σχεδίαση προγράμματος;**

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

### **Ερώτηση 6.19: Τι γνωρίζετε για τον τμηματικό προγραμματισμό;**

Η ιεραρχική σχεδίαση προγράμματος υλοποιείται με τον τμηματικό προγραμματισμό. ο τμηματικός προγραμματισμός διευκολύνει τη δημιουργία του προγράμματος, μειώνει τα λάθη και επιτρέπει την ευκολότερη παρακολούθηση, κατανόηση και συντήρηση του προγράμματος από τρίτους.

### **Ερώτηση 6.20: Τι είναι ο δομημένος προγραμματισμός και στην χρήση ποιών δομών στηρίζεται;**

Ο 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 προγράμματος έχει μόνο μία είσοδο και μόνο μία έξοδο.

### **Ερώτηση 6.21: Ποια είναι τα πλεονεκτήματα του δομημένου προγραμματισμού**

Επιγραμματικά μπορούμε να αναφέρουμε τα εξής πλεονεκτήματα του δομημένου προγραμματισμού.

1. Δημιουργία απλούστερων προγραμμάτων.
2. Άμεση μεταφορά των αλγορίθμων σε προγράμματα.
3. Διευκόλυνση ανάλυσης του προγράμματος σε τμήματα.
4. Περιορισμός των λαθών κατά την ανάπτυξη του προγράμματος.
5. Διευκόλυνση στην ανάγνωση και κατανόηση του προγράμματος από τρίτους.
6. Ευκολότερη διόρθωση και συντήρηση.

## 6ο Κεφάλαιο – Εισαγωγή στον προγραμματισμό.

### **Ερώτηση 6.22: Τι είναι οι μεταγλωττιστές και τι οι διερμηνευτές;**

Κάθε πρόγραμμα που γράφτηκε σε οποιαδήποτε γλώσσα προγραμματισμού, πρέπει να μετατραπεί σε μορφή αναγνωρίσιμη και εκτελέσιμη από τον υπολογιστή, δηλαδή σε εντολές γλώσσας μηχανής.

Η μετατροπή αυτή επιτυγχάνεται με τη χρήση ειδικών μεταφραστικών προγραμμάτων.

Υπάρχουν δύο μεγάλες κατηγορίες τέτοιων προγραμμάτων, οι μεταγλωττιστές (compilers) και οι διερμηνευτές (interpreters).

Ο μεταγλωττιστής,

δέχεται στην είσοδο ένα πρόγραμμα γραμμένο σε μια γλώσσα υψηλού επιπέδου και παράγει ένα ισοδύναμο πρόγραμμα σε γλώσσα μηχανής.

Αντίθετα,

Ο διερμηνευτής,

διαβάζει μία προς μία τις εντολές του αρχικού προγράμματος και για κάθε μια εκτελεί αμέσως μια ισοδύναμη ακολουθία εντολών μηχανής.

### **Ερώτηση 6.23: Ποια είναι η διαδικασία ώστε να πάρουμε το εκτελέσιμο πρόγραμμα;**

Το αρχικό πρόγραμμα λέγεται πηγαίο πρόγραμμα (source), ενώ το πρόγραμμα που παράγεται από το μεταγλωττιστή λέγεται αντικείμενο πρόγραμμα (object).

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.

### **Ερώτηση 6.24: Ποιες είναι οι κατηγορίες λαθών και τι γνωρίζετε σε γενικές γραμμές για αυτές; Ποια λάθη ανιχνεύονται από το μεταγλωττιστή ή τον διερμηνευτή;**

Ο μεταγλωττιστής ή ο διερμηνευτής ανιχνεύει λοιπόν τα λάθη και εμφανίζει κατάλληλα διαγνωστικά μηνύματα. Το στάδιο που ακολουθεί είναι η διόρθωση των λαθών. Το διορθωμένο πρόγραμμα επαναυποβάλλεται για μεταγλώττιση και η διαδικασία αυτή επαναλαμβάνεται, μέχρις ότου εξαληφθούν πλήρως όλα τα λάθη. Τα λογικά λάθη είναι τα πλέον σοβαρά και δύσκολα στη διόρθωση τους, οφείλονται σε σφάλματα κατά την υλοποίηση του αλγορίθμου, ενώ τα συντακτικά οφείλονται σε

## 6ο Κεφάλαιο – Εισαγωγή στον προγραμματισμό.

αναγραμματισμούς ονομάτων εντολών, παράληψη δήλωσης δεδομένων και πρέπει πάντα να διορθωθούν, ώστε να παραχθεί το τελικό εκτελέσιμο πρόγραμμα.

**Ερώτηση 6.25:** Ποια είναι τα πλεονεκτήματα και τα μειονεκτήματα του μεταγλωττιστή και του διερμηνευτή;

Η χρήση μεταγλωττιστή έχει το μειονέκτημα, ότι προτού χρησιμοποιηθεί ένα πρόγραμμα, πρέπει να περάσει από τη διαδικασία της μεταγλώττισης και σύνδεσης. Από την άλλη μεριά η χρήση διερμηνευτή έχει το πλεονέκτημα της άμεσης εκτέλεσης και συνεπώς και της άμεσης διόρθωσης. Όμως η εκτέλεση του προγράμματος καθίσταται πιο αργή, σημαντικά μερικές φορές, από εκείνη του ισοδύναμου εκτελέσιμου προγράμματος που παράγει ο μεταγλωττιστής.

**Ερώτηση 6.26:** Ποια είναι τα προγράμματα που χρησιμοποιούνται για τη δημιουργία, τη μετάφραση και την εκτέλεση ενός προγράμματος και αναφέρατε λίγα λόγια για το κάθε ένα.

Για την αρχική σύνταξη των προγραμμάτων και τη διόρθωσή τους στη συνέχεια χρησιμοποιείται ένα ειδικό πρόγραμμα που ονομάζεται συντάκτης (editor).

Ο συντάκτης είναι ουσιαστικά ένας μικρός επεξεργαστής κειμένου, με δυνατότητες όμως που διευκολύνουν τη γρήγορη γραφή των εντολών των προγραμμάτων.

Για τη δημιουργία, τη μετάφραση και την εκτέλεση ενός προγράμματος απαιτούνται τουλάχιστον τρία προγράμματα: ο συντάκτης, ο μεταγλωττιστής και ο συνδέτης.

Τα σύγχρονα προγραμματιστικά περιβάλλοντα παρέχουν αυτά τα προγράμματα με ενιαίο τρόπο.