

# Κεφάλαιο 7<sup>ο</sup>

## Προγραμματισμός Η/Υ

### **Τι είναι αλγόριθμος;**

Αλγόριθμος είναι μια πεπερασμένη σειρά ενεργειών, αυστηρά καθορισμένων και εκτελέσιμων σε πεπερασμένο χρόνο, που στοχεύουν στην επίλυση ενός προβλήματος. Κάθε μία από τις αριθμημένες ενέργειες καλείται εντολή αλγορίθμου.

### **Ποια κριτήρια πρέπει να ικανοποιεί κάθε αλγόριθμος;**

Κάθε αλγόριθμος πρέπει να ικανοποιεί τα επόμενα κριτήρια:

1. Είσοδος είναι ένα σύνολο μεταβλητών, οι οποίες αποτελούν τα δεδομένα του αλγορίθμου. Η είσοδος είναι απαραίτητη σε έναν αλγόριθμο. Για αυτό το λόγο σε έναν αλγόριθμο πρέπει να δίνεται ως είσοδος καμία, μία ή περισσότερες τιμές.
2. Έξοδος είναι ένα σύνολο μεταβλητών, οι οποίες αποτελούν τα αποτελέσματα του αλγορίθμου. Ο αλγόριθμος πρέπει να δημιουργεί τουλάχιστον μία τιμή ως αποτέλεσμα προς το χρήστη ή προς έναν άλλο αλγόριθμο.
3. Κάθε εντολή ενός αλγορίθμου πρέπει να καθορίζεται χωρίς καμία αμφιβολία για τον τρόπο εκτέλεσής της. Για αυτό το λόγο ο αλγόριθμος πρέπει να χαρακτηρίζεται από καθοριστικότητα.
4. Ένας αλγόριθμος πρέπει να τελειώνει μετά από πεπερασμένα βήματα εκτέλεσης των εντολών του, δηλαδή να έχει περατότητα. Όταν μια διαδικασία δεν τελειώνει μετά από ένα συγκεκριμένο αριθμό βημάτων δεν αποτελεί αλγόριθμο, αλλά απλά υπολογιστική διαδικασία.
5. Κάθε μεμονωμένη εντολή του αλγορίθμου πρέπει να είναι διατυπωμένη απλά. Αυτό σημαίνει ότι μία εντολή δεν αρκεί να έχει ορισθεί, αλλά πρέπει να είναι και εκτελέσιμη. Το κριτήριο αυτό καλείται αποτελεσματικότητα.

### **Ποιοι είναι οι τρόποι αναπαράστασης ενός αλγορίθμου;**

Η αναπαράσταση ενός αλγορίθμου μπορεί να γίνει με:

1. Ελεύθερο κείμενο αποτελεί τον πιο ανεπεξέργαστο και

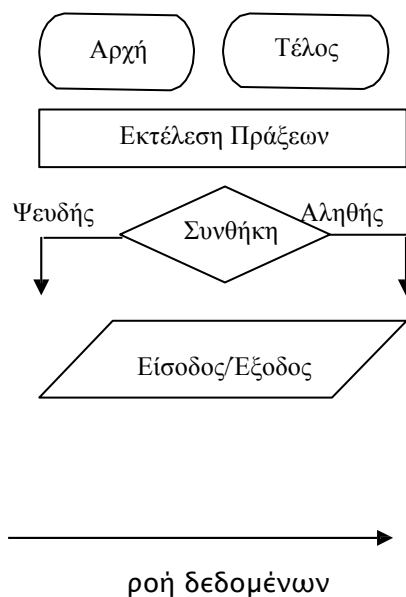
αδόμητο τρόπο παρουσίασης αλγορίθμου.

2. Φυσική γλώσσα με βήματα.
3. Διαγραμματικές τεχνικές που συνιστούν ένα γραφικό τρόπο παρουσίασης του αλγορίθμου. Η πιο γνωστή τεχνική είναι το διάγραμμα ροής.
4. Κωδικοποίηση δηλαδή με ένα πρόγραμμα που όταν εκτελεσθεί θα δώσει τα ίδια αποτελέσματα με τον αλγόριθμο.

Για να γραφεί ένας αλγόριθμος σε ελεύθερο κείμενο χρησιμοποιούμε την καθομιλουμένη γλώσσα, όπως ακριβώς προσπαθούμε να εξηγήσουμε σε κάποιον τον τρόπο με τον οποίο θα κάνει κάτι.

Για να γραφεί ένας αλγόριθμος σε φυσική γλώσσα με βήματα χρησιμοποιούμε την καθομιλουμένη, με τις προτάσεις να είναι αριθμημένες και να εκτελούνται σε βήματα.

Για να φτιαχτεί ένα διάγραμμα ροής χρησιμοποιούνται τα ακόλουθα σύμβολα:



Με την έλλειψη δηλώνεται η αρχή και το τέλος του αλγορίθμου.

Με το πλάγιο παραλληλόγραμμο δηλώνεται η είσοδος ή η έξοδος στοιχείων.

Με το ορθογώνιο δηλώνεται η εκτέλεση των πράξεων.

Με το ρόμβο δηλώνεται μία ερώτηση με δύο ή περισσότερες εξόδους για απάντηση

Με το βέλος δηλώνεται η ροή των δεδομένων.

Ας δούμε ένα απλό παράδειγμα προβλήματος, την υλοποίηση της λύσης του και τους τρόπους αναπαράστασης του αλγόριθμου.

Να γράψετε αλγόριθμο ο οποίος θα διαβάζει τρεις αριθμούς και θα υπολογίζει και θα εμφανίζει το μέσο όρο τους.

1. Ψευδογλώσσα

Αλγόριθμος μέσος\_όρος\_τριών\_αριθμών

Διάβασε  $\alpha, \beta, \gamma$

$\text{άθροισμα} \leftarrow \alpha + \beta + \gamma$

$\text{πηλίκιο} \leftarrow \text{άθροισμα} / 3$

Εκτύπωσε πηλίκιο

Τέλος μέσος\_όρος\_τριών\_αριθμών

2. Ελεύθερο κείμενο

Διάβασε τρεις αριθμούς. Στη συνέχεια πρόσθεσέ τους και το αποτέλεσμα διαίρεσέ το με 3. Εκτύπωσε το αποτέλεσμα μετά τη διαίρεση.

3. Φυσική γλώσσα με βήματα

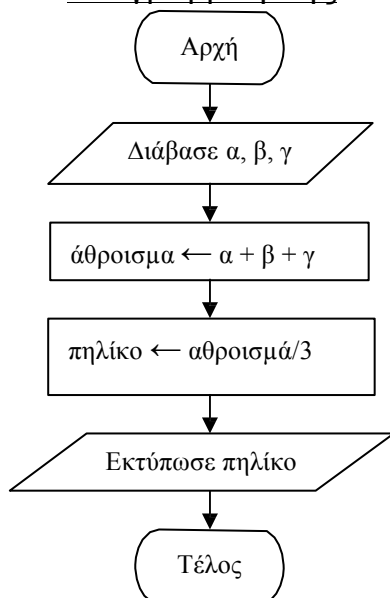
1. Διάβασε τρεις αριθμούς  $\alpha, \beta, \gamma$

2. Θέσε το άθροισμα ίσο με  $\alpha + \beta + \gamma$

3. Θέσε το πηλίκιο ίσο με το άθροισμα δια 3.

4. Εκτύπωσε το πηλίκιο

4. Διάγραμμα ροής



Στο παράδειγμα αυτό δόθηκε ο αλγόριθμος και στις τέσσερις μορφές περιγραφής του.

### 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.

Αλγόριθμοι + Δομές Δεδομένων=πρόγραμμα

Βασικό στοιχείο του προγράμματος είναι και τα δεδομένα και οι δομές δεδομένων επί των οποίων ενεργεί.

### Γιατί αναπτύχθηκε ο προγραμματισμός και οι γλώσσες προγραμματισμού;

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

Οι γλώσσες προγραμματισμού αναπτύχθηκαν με σκοπό την επικοινωνία του ανθρώπου (προγραμματιστή) με τη μηχανή (υπολογιστή)

### Τι είναι οι γλώσσες μηχανής και ποια τα μειονεκτήματά τους;

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

Οι ακολουθίες δυαδικών ψηφίων είναι δυσνόητες σχεδόν για το σύνολο των ανθρώπων και πολύ δύσκολο να απομνημονευθούν

Η υλοποίηση ενός προγράμματος σε γλώσσα μηχανής είναι αρκετά επίπονη και απαιτεί βαθιά γνώση του υλικού και της αρχιτεκτονικής του υπολογιστή.

Τι είναι οι συμβολικές γλώσσες και ποια τα μειονεκτήματά τους;

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

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

Οι συμβολικές γλώσσες εξαρτώνται από την αρχιτεκτονική του υπολογιστή.

Τα προγράμματα είναι δύσκολο να γραφούν, να συντηρηθούν και να μεταφερθούν σε άλλο υπολογιστή.

### Τι είναι οι γλώσσες υψηλού επιπέδου;

Στις γλώσσες υψηλού επιπέδου το πρόγραμμα που γράφεται,

μεταφράζεται από τον ίδιο τον υπολογιστή στις ακολουθίες των εντολών μηχανής, με τη βοήθεια ενός ειδικού προγράμματος που ονομάζεται μεταγλωττιστής (compiler).

Το ίδιο πρόγραμμα μπορεί να εκτελεσθεί σε οποιοδήποτε υπολογιστή, αρκεί να υπάρχει ο αντίστοιχος μεταγλωττιστής.

**Ποιες γλώσσες προγραμματισμού αποτέλεσαν σημεία αναφοράς στον προγραμματισμό;**

Το 1957 η IBM ανέπτυξε την πρώτη γλώσσα υψηλού επιπέδου και την ονόμασε FORTRAN. Το όνομα FORTRAN προέρχεται από τις λέξεις FORmula TRANslation. Αναπτύχθηκε ως γλώσσα κατάλληλη για την επίλυση μαθηματικών και επιστημονικών εφαρμογών.

Το 1960 αναπτύχθηκε η COBOL (COmmon Business Oriented Language). Είναι κατάλληλη για ανάπτυξη εμπορικών και γενικά διαχειριστικών εφαρμογών. Χρησιμοποιείται από επιχειρήσεις και από δημόσια διοίκηση.

Η ALGOL (ALGOrithmic Language) αναπτύχθηκε στην Ευρώπη με σκοπό τη δημιουργία γενικής φύσης προγραμμάτων που να μην συνδέονται με συγκεκριμένες εφαρμογές. Είχε ελάχιστη πρακτική εφαρμογή.

Η PL/1 (Programming Language 1) αναπτύχθηκε με σκοπό να αντικαταστήσει τη FORTRAN και τη COBOL χωρίς όμως επιτυχία.

Στο χώρο της τεχνητής νοημοσύνης αναπτύχθηκαν δύο γλώσσες αρκετά διαφορετικές από όλες τις άλλες η LISP και η PROLOG. Χρησιμοποιούνται σε έμπειρα συστήματα, παιχνίδια, επεξεργασία φυσικών γλωσσών.

Η BASIC (Beginner's All Purpose Symbolic Instruction Code) αναπτύχθηκε για την εκπαίδευση αρχαρίων στον προγραμματισμό. Σχεδιάστηκε για να γράφονται σύντομα προγράμματα τα οποία εκτελούνται με τη βοήθεια διερμηνευτή (interpreter). Η ανάπτυξη των μικροϋπολογιστών και η τυποποίηση της από τη Microsoft τη καθιέρωσε ως πρότυπο για ανάπτυξη εφαρμογών σε προσωπικούς υπολογιστές.

Η PASCAL (Niklaus Wirth) βασίστηκε στην ALGOL και είναι μια γλώσσα γενικής χρήσης. Διέπεται από τις αρχές του δομημένου προγραμματισμού. Γνωρίζει τεράστια εξάπλωση, και αποτέλεσε τη βάση για την ανάπτυξη άλλων ισχυρότερων γλωσσών όπως η ADA και η Modula-2.

Η C και η μετεξέλιξη της C++ δημιουργήθηκε από τα εργαστήρια BELL και χρησιμοποιήθηκε για την ανάπτυξη του λειτουργικού συστήματος UNIX. Είναι γλώσσα με ισχυρά χαρακτηριστικά. Η C++ είναι γλώσσα αντικειμενοστραφούς προγραμματισμού.

Η JAVA είναι αντικειμενοστραφής γλώσσα που αναπτύχθηκε από τη SUN με σκοπό την ανάπτυξη εφαρμογών για το διαδίκτυο.

### Ποια είναι τα πλεονεκτήματα των Γλωσσών Υψηλού Επιπέδου;

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.

Συνολικά οι γλώσσες υψηλού επιπέδου ελάττωσαν σημαντικά το χρόνο και το κόστος παραγωγής νέων προγραμμάτων, αφού λιγότεροι προγραμματιστές μπορούν σε μικρότερο χρόνο να αναπτύξουν προγράμματα που χρησιμοποιούνται σε περισσότερους υπολογιστές.

### Τι είναι οι γλώσσες 4<sup>ης</sup> γενιάς;

Οι γλώσσες υψηλού επιπέδου απευθύνονται μόνο σε προγραμματιστές. Σταδιακά όμως πολλές γλώσσες εφοδιάστηκαν με εργαλεία προγραμματισμού που αποκρύπτουν πολλές

λεπτομέρειες από τις τεχνικές υλοποίησης και έτσι ο χρήστης μπορεί να επιλύει μόνος του μικρά προβλήματα εφαρμογών.

Η αυξανόμενη τάση απόκρυψης της αρχιτεκτονικής του υλικού και της τεχνικής του προγραμματισμού οδήγησε στις γλώσσες 4ης γενιάς. Παραδείγματα τέτοιων γλωσσών είναι η dBASE, ο Clipper, η Access, η SQL.

### Ποια είναι η καλύτερη γλώσσα προγραμματισμού;

Γλώσσα προγραμματισμού που να είναι αντικειμενικά καλύτερη από τις άλλες δεν υπάρχει, ούτε πρόκειται να υπάρξει.

Ο προγραμματιστής επιλέγει τη γλώσσα με βάση τα εξής:

1. το είδος της εφαρμογής
2. το υπολογιστικό περιβάλλον στο οποίο θα εκτελεστεί το πρόγραμμα
3. τα προγραμματιστικά περιβάλλοντα που διαθέτει
4. τις γνώσεις του και τις προσωπικές του προτιμήσεις

### Πως ταξινομούνται οι γλώσσες προγραμματισμού με βάση τον τρόπο συγγραφής των εντολών;

Με βάση τον τρόπο συγγραφής των εντολών σε:

1. Διαδικασιακές (procedural) που είναι επίσης γνωστές και ως αλγοριθμικές γλώσσες, γιατί είναι σχεδιασμένες για να επιτρέπουν την υλοποίηση αλγορίθμων.
2. Αντικειμενοστραφείς γλώσσες (object oriented languages).
3. Συναρτησιακές γλώσσες (functional languages) π.χ. LISP.
4. Μη διαδικασιακές γλώσσες (non procedural languages) π.χ. PROLOG. Χαρακτηρίζονται επίσης και ως γλώσσες πολύ υψηλού επιπέδου.

### Πως ταξινομούνται οι γλώσσες προγραμματισμού με βάση την περιοχή χρήσης;

1. Γλώσσες γενικής χρήσης
  - I. Γλώσσες επιστημονικής κατεύθυνσης (science oriented languages) π.χ FORTRAN
  - II. Γλώσσες εμπορικής κατεύθυνσης (business oriented languages) π.χ. COBOL
  - III. Μερικές γλώσσες τα καταφέρνουν εξίσου καλά και

- στους δυο παραπάνω τομείς π.χ. BASIC, PASCAL
2. Γλώσσες προγραμματισμού συστημάτων (system programming languages) π.χ. C
  3. Γλώσσες τεχνητής νοημοσύνης (artificial intelligence languages) π.χ. LISP, PROLOG
  4. Γλώσσες ειδικής χρήσης. Χρησιμοποιούνται σε ειδικές εφαρμογές όπως γραφικά, ρομποτική, εκπαίδευση μέσω υπολογιστή, σχεδίαση ολοκληρωμένων κυκλωμάτων

### Τι είναι οι τεχνική της ιεραρχικής σχεδίασης προγραμμάτων;

Η ιεραρχική σχεδίαση ή ιεραρχικός προγραμματισμός χρησιμοποιεί τη στρατηγική της συνεχούς διαίρεσης του προβλήματος σε υποπροβλήματα. Τα υποπροβλήματα είναι απλούστερα και είναι πιο εύκολο να επιλυθούν οδηγώντας στην λύση του αρχικού προβλήματος. Στην ιεραρχική σχεδίαση χρησιμοποιούνται διαγραμματικές τεχνικές.

### Τι είναι ο τμηματικός προγραμματισμός;

Η ιεραρχική σχεδίαση προγράμματος υλοποιείται με τον τμηματικό προγραμματισμό. Κάθε υποπρόβλημα αποτελεί ανεξάρτητη ενότητα (module) που γράφεται ξεχωριστά από τα υπόλοιπα τμήματα προγράμματος. Ο τμηματικός προγραμματισμός διευκολύνει τη δημιουργία του προγράμματος, μειώνει τα λάθη και επιτρέπει την ευκολότερη παρακολούθηση, κατανόηση και συντήρηση του προγράμματος από τρίτους.

### Τι είναι ο δομημένος προγραμματισμός;

Η μεθοδολογία που σήμερα έχει επικρατήσει απόλυτα και υποστηρίζεται από όλες τις σύγχρονες γλώσσες ονομάζεται δομημένος προγραμματισμός (structured programming). Ο δομημένος προγραμματισμός αναπτύχθηκε από την ανάγκη να υπάρξει μια κοινή μεθοδολογία στην ανάπτυξη των προγραμμάτων.

Ο δομημένος προγραμματισμός στηρίζεται στη χρήση τριών και μόνο στοιχειωδών λογικών δομών:

1. Δομή Ακολουθίας
2. Δομή Επιλογής



### 3. Δομή επανάληψης

Όλα τα προγράμματα μπορούν να γραφούν χρησιμοποιώντας μόνο τις τρεις αυτές δομές.

#### Τι είναι η δομή ακολουθίας;

Η ακολουθιακή δομή εντολών (σειριακών βημάτων) χρησιμοποιείται για την αντιμετώπιση απλών προβλημάτων, όπου είναι δεδομένη η σειρά εκτέλεσης ενός συνόλου ενεργειών

#### Τι είναι η δομή επιλογής;

Στην επίλυση προβλημάτων η πλέον συνηθισμένη περίπτωση είναι να λαμβάνονται κάποιες αποφάσεις με βάση κάποια δεδομένα κριτήρια. Γενικά η διαδικασία της επιλογής περιλαμβάνει τον έλεγχο κάποιας συνθήκης που μπορεί να έχει δύο τιμές (Αληθής ή Ψευδής) και ακολουθεί η απόφαση εκτέλεσης κάποιας ενέργειας με βάση την τιμή της λογικής αυτής συνθήκης.

#### Τι είναι η δομή επανάληψης;

Η δομή επανάληψης ή βρόχος επιτρέπει την εκτέλεση εντολών περισσότερο από μία φορά. Οι επαναλήψεις ελέγχονται πάντα από κάποια συνθήκη, η οποία καθορίζει την έξοδο από το βρόχο.

#### Ποια είναι τα 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

#### Τι εξυπηρετούν τα μεταφραστικά προγράμματα;

Κάθε πρόγραμμα σε οποιαδήποτε γλώσσα 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.

### Ποια η διαφορά των μεταφραστών και των διερμηνέων;

Οι μεταγλωττιστές (compilers) δέχονται στην είσοδο ένα πρόγραμμα γραμμένο σε οποιαδήποτε γλώσσα υψηλού επιπέδου και παράγουν ένα ισοδύναμο πρόγραμμα σε γλώσσα μηχανής.

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

Οι διερμηνευτές (interpreters) διαβάζουν μία προς μία τις εντολές του αρχικού προγράμματος και για κάθε μια εκτελούν αμέσως μια ισοδύναμη ακολουθία εντολών μηχανής.

### Ποια η διαδικασία δημιουργίας εκτελέσιμου προγράμματος με τη χρήση compiler;

Το αρχικό πρόγραμμα λέγεται πηγαίο πρόγραμμα (source).

Το πρόγραμμα που παράγεται από το μεταγλωττιστή λέγεται αντικείμενο πρόγραμμα (object).

Το αντικείμενο πρόγραμμα δεν είναι σε θέση να εκτελεστεί. Χρειάζεται να συνδεθεί με άλλα τμήματα προγράμματος τα οποία 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 είναι η παραγωγή του εκτελέσιμου προγράμματος (executable)

### Τι ρόλο παίζουν τα λάθη στην δημιουργία του προγράμματος;

Η δημιουργία του εκτελέσιμου προγράμματος γίνεται μόνο στην περίπτωση, που το αρχικό πρόγραμμα δεν περιέχει λάθη. Τα λάθη που εμφανίζονται κατά τη μεταγλώττιση ονομάζονται συντακτικά. Τα συντακτικά λάθη οφείλονται σε αναγραμματισμούς ονομάτων εντολών, και παράληψη δήλωσης δεδομένων. Ο μεταγλωττιστής ή ο διερμηνευτής ανιχνεύει τα λάθη και εμφανίζει κατάλληλα διαγνωστικά μηνύματα. Στη συνέχεια ακολουθεί η διόρθωση τους

Εκτός από τα συντακτικά λάθη που ανιχνεύονται κατά τη μεταγλώττιση υπάρχουν και τα λογικά λάθη.

Είναι τα πλέον σοβαρά και δύσκολα στη διόρθωση τους, και οφείλονται σε σφάλματα κατά την υλοποίηση του αλγορίθμου.

**Πως λέγεται η διαδικασία ανίχνευσης και διόρθωσης λαθών;**

Η διαδικασία αυτή ονομάζεται εκσφαλμάτωση.

**Ποια τα πλεονεκτήματα της χρήσης μεταφραστή ή διερμηνευτή και πως συνδυάζεται η χρήση τους από τα σύγχρονα προγραμματιστικά περιβάλλοντα;**

Η χρήση μεταγλωττιστή έχει το μειονέκτημα ότι προτού χρησιμοποιηθεί ένα πρόγραμμα πρέπει να περάσει από τη διαδικασία της μεταγλώττισης και της σύνδεσης.

Η χρήση του διερμηνευτή έχει το πλεονέκτημα της άμεσης εκτέλεσης και συνεπώς της άμεσης διόρθωσης. Όμως η εκτέλεση του προγράμματος καθίσταται πιο αργή. Τα σύγχρονα εργαλεία προγραμματισμού χρησιμοποιούν διερμηνευτή κατά τη φάση της δημιουργίας και μεταγλωττιστή για την τελική έκδοση του προγράμματος.