

1. ΠΡΟΓΡΑΜΜΑΤΙΣΜΟΣ (programming):

είναι η διαδικασία γραφής προγράμματος. ¶

2. ΠΡΟΓΡΑΜΜΑ (program):

είναι αρχείο ή αρχεία που περιέχουν ένα σύνολο από ΟΔΗΓΙΕΣ (εντολές) που

α) καταλαβαίνει ένα κομπιούτερ και

β) με τις οποίες ο επεξεργαστής επεξεργάζεται ΠΛΗΡΟΦΟΡΙΕΣ (κείμενα, εικόνες και ήχους).

ΦΤΙΑΧΝΟΥΜΕ προγράμματα γιατί τα κομπιούτερ είναι μηχανήματα που λειτουργούν με ρεύμα, χωρίς μυαλό και χρειάζονται οδηγίες για να δουλέψουν. ¶

3. ΠΡΟΓΡΑΜΜΑΤΙΣΤΗΣ - ΧΡΗΣΤΗΣ (programer - user):

Προγραμματιστής είναι ο άνθρωπος που φτιάχνει το πρόγραμμα.

Χρήστης είναι ο άνθρωπος που το εκτελεί. ¶

4. ΓΛΩΣΣΑ-ΠΡΟΓΡΑΜΜΑΤΙΣΜΟΥ (programming-language):

είναι η γλώσσα που καταλαβαίνει ο υπολογιστής και με την οποία γράφουμε τις οδηγίες των προγραμμάτων. Σήμερα ΔΕΝ είναι φυσική γλώσσα.

Η βασική διαφορά των γλωσσών-προγραμματισμού με τις φυσικές-γλώσσες είναι η ΑΚΡΙΒΕΙΑ. Επειδή έχουμε να κάνουμε με μηχανήμα (ηλεκτρικό ρεύμα), οι γλώσσες-προγραμματισμού είναι ΑΠΟΛΥΤΑ ακριβείς, 100%. Ένα πρόγραμμα με 1.000.000 σωστές εντολές, αν λείπει μια τελεία, ΔΕΝ δουλεύει. ¶

5. ΑΛΓΟΡΙΘΜΟΣ (algorithm):

Αλγόριθμος (υπολογιστή | προγραμματισμού) είναι μια ΔΙΑΔΙΚΑΣΙΑ ΕΠΕΞΕΡΓΑΣΙΑΣ ΠΛΗΡΟΦΟΡΙΑΣ που μπορεί να εκτελέσει ένας ΥΠΟΛΟΓΙΣΤΗΣ.

Αν ο προγραμματιστής, στο μυαλό του, δεν έχει ξεκαθαρίσει τον αλγόριθμο στη μητρική του γλώσσα και μετά να τον μεταφράσει σε μια γλώσσα προγραμματισμού, ποτέ δε θα φτιάξει πρόγραμμα. Επίσης, επειδή οι υπολογιστές δεν κάνουν καφέδες, γιατί ο αλγόριθμος είναι μια διαδικασία επεξεργασίας πληροφορίας (= επεξεργασία συμβόλων, εικόνων και ήχων) και όχι μια οποιαδήποτε διαδικασία.

Τέλος αν τη διαδικασία αυτή επεξεργασίας πληροφορίας ΔΕΝ ΜΠΟΡΕΙ

να την εκτελέσει ένας υπολογιστής, τότε ΔΕΝ είναι αλγόριθμος-προγραμματισμού. ¶

6. ΠΡΟΒΛΗΜΑ (problem):

Ο αλγόριθμος είναι η λύση ενός προβλήματος. ¶

7. ΙΔΙΟΤΗΤΕΣ-ΑΛΓΟΡΙΘΜΟΥ (algorithm properties):

α) να είναι σωστός και ακριβής

β) να είναι περατός, δηλαδή να έχει τέλος. ¶

8. ΧΑΡΑΚΤΗΡΙΣΤΙΚΑ-ΓΛΩΣΣΩΝ (programming-language attributes):

Αλφάβητο, λεξιλόγιο, συντακτικό. ¶

9. ΓΛΩΣΣΑ-ΜΗΧΑΝΗΣ (machine language):

Η γλώσσα – προγραμματισμού που καταλαβαίνει ΑΜΕΣΑ το μηχάνημα, οι εντολές της γράφονται με 0 και 1. ¶

10. ΓΛΩΣΣΑ-ΥΨΗΛΟΥ-ΕΠΙΠΕΔΟΥ (high-level language):

Είναι η γλώσσα προγραμματισμού που έχει τις εντολές της με λέξεις από τις φυσικές-γλώσσες. Σε αυτές γράφονται τα προγράμματα. ¶

11.ΜΕΤΑΦΡΑΣΤΗΣ (translator):

Είναι το πρόγραμμα που μεταφράζει το πρόγραμμα από γλώσσα-υψηλού-επιπέδου σε γλώσσα μηχανής.

Μεταγλωτιστής (compiler) λέγεται ο μεταφραστής που μεταφράζει ολόκληρα προγράμματα.

Διερμηνέας (interpreter) λέγεται ο μεταφραστής που μεταφράζει εντολή εντολή. ¶

12. ΟΛΟΚΛΗΡΩΜΕΝΟ-ΠΡΟΓΡΑΜΜΑΤΙΣΤΙΚΟ-ΠΕΡΙΒΑΛΛΟΝ (IDE):

Είναι το πρόγραμμα (σύνολο προγραμμάτων) με το οποίο οι προγραμματιστές φτιάχνουν προγράμματα. Περιέχει editor, μεταφραστή και εκσφαλματωτή(debugger). ¶

13. ΛΑΘΗ (error):

Τα συντακτικά μπορεί να τα βρεί το ΟΠΠ τα λογικά όχι. ¶

14. ΣΤΑΔΙΑ-ΕΚΤΕΛΕΣΗΣ-ΑΛΓΟΡΙΘΜΟΥ:

είναι ΟΛΗ η διαδικασία δημιουργίας και εκτέλεσης προγράμματος.

α) Αλγόριθμος:

ο προγραμματιστής ξεκινάει από το μυαλό του και σκέφτεται στη μητρική του γλώσσα τι θέλει να κάνει το πρόγραμμά του (μια διαδικασία επεξεργασίας συμβόλων-εικόνων-ήχων).

β) Πρόγραμμα:

ο προγραμματιστής γράφει στο ΟΠΠ το πρόγραμμα, δηλαδή

'μεταφράζει' τις ενέργειες που θέλει να κάνει το κομπιούτερ του από τη μητρική του γλώσσα σε μια γλώσσα προγραμματισμού.

γ) Μετάφραση:

ο μεταφραστής μετατρέπει το πρόγραμμα που έγραψε ο προγραμματιστής με λέξεις σε πρόγραμμα με 0 και 1.

δ) Εκτέλεση από ΚΜΕ:

ο επεξεργαστής εκτελεί το τελικό πρόγραμμα.

Ερωτήσεις – Απαντήσεις

1. Γιατί πρέπει να κατανοούμε καλά ένα πρόβλημα, πριν να το επιλύσουμε;

για να επιλύσουμε ένα πρόβλημα πρέπει αρχικά να το **κατανοήσουμε**. Πρέπει δηλαδή να καταλάβουμε καλά το περιεχόμενο του, να διακρίνουμε τα δεδομένα που έχουμε στη διάθεσή μας και τα ζητούμενά του. Είναι σημαντικό, όμως, να προσδιορίσουμε και το «**περιβάλλον**» ή το πλαίσιο μέσα στο οποίο εντάσσεται το πρόβλημα (χώρος του προβλήματος). Η κατανόηση του προβλήματος που κρινόμαστε να επιλύσουμε αποτελεί τα γερά θεμέλια προκειμένου να βρούμε σύντομα και χωρίς κόπο τη λύση του.

2. Ποιες διαδικασίες μας βοηθούν στην κατανόηση ενός προβλήματος;

- a.** Η σωστή καταγραφή των δεδομένων του προβλήματος, ώστε να γίνει όσο το δυνατό πιο κατανοητό.
- b.** Ο σωστός εντοπισμός των ζητούμενων του προβλήματος.
- c.** Ο προσδιορισμός του χώρου του προβλήματος (του πλαισίου στο οποίο εντάσσεται).

3. Τι είναι ένας αλγόριθμος;

Οι οδηγίες που δίνουμε με λογική σειρά, ώστε να εκτελέσουμε μια εργασία ή να επιλύσουμε ένα πρόβλημα, συνθέτουν έναν **Αλγόριθμο**. Για παράδειγμα, οι οδηγίες για την κατασκευή ενός χαρταετού μπορεί να αποτελέσουν έναν αλγόριθμο.

Λύση παραδ. 1

Αλγόριθμος Παράδειγμα_1
Διάβασε a
Διάβασε b
 $c \leftarrow a + b$
Εκτύπωσε c
Τέλος Παράδειγμα_1

Πρόσθεση 2 αριθμών

- Δώσε τον 1^ο αριθμό
- Δώσε τον 2^ο αριθμό
- Πρόσθεσέ τους
- Εμφάνισε το αποτέλεσμα

Παραδειγμα 2

Αλγόριθμος για τη δημιουργία ελληνικού καφέ!!!

Αλγόριθμος 1

1. **Βάλε** ένα ποτήρι του καφέ νερό στο μπρίκι.
2. **Βάλε** ένα κουταλάκι του γλυκού καφέ στο μπρίκι.
3. **Βάλε** ένα κουταλάκι του γλυκού ζάχαρη στο μπρίκι.
4. **Βάλε** το μπρίκι στο μάτι της κουζίνας.
5. **Άναψε** το μάτι της κουζίνας.
6. **Ανακάτεψε** το μείγμα μέχρι να φουσκώσει.
7. **Βάλε** τον καφέ στο φλιτζάνι.

Αλγόριθμος 2

1. **Βάλε** ένα ποτήρι του καφέ νερό στο μπρίκι.
2. **Βάλε** καφέ στο μπρίκι.
3. **Βάλε** ζάχαρη στο μπρίκι.
4. **Βάλε** το μπρίκι στο μάτι της κουζίνας.
5. **Ανακάτεψε** το μείγμα.

4. Ποιες είναι οι βασικές ιδιότητες ενός Αλγορίθμου;

- a. Πρέπει να είμαστε σίγουροι ότι αν υλοποιήσουμε τον αλγόριθμο κάποια στιγμή θα τελειώσει επιτυγχάνοντας τον αρχικό σκοπό.
- b. Οι εντολές του αλγορίθμου πρέπει να έχουν ακρίβεια και σαφήνεια, ώστε να μην μπερδευτεί αυτός που θα υλοποιήσει τον αλγόριθμο και τις εκτελέσει με λανθασμένο τρόπο.
- c. Οι εντολές ενός αλγορίθμου πρέπει να είναι εκφρασμένες με απλά λόγια.

Τι είναι πρόγραμμα;

- Είναι μια σειρά από εντολές / οδηγίες, που συντονίζουν την λειτουργία του Η/Υ για την εκτέλεση μιας συγκεκριμένης εργασίας.



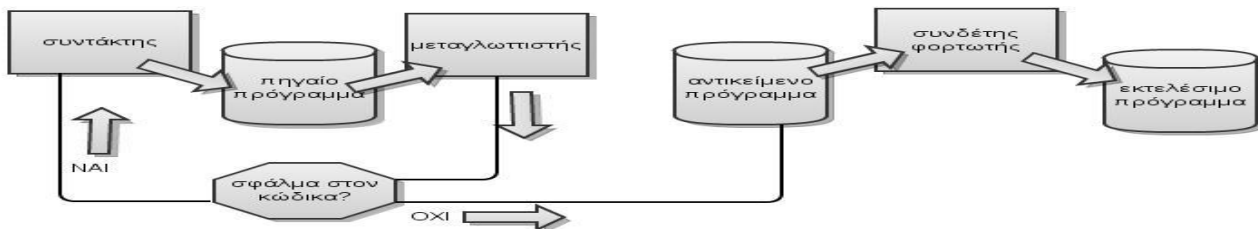
15

5. Τι είναι πρόγραμμα;

Ένα πρόγραμμα είναι η αναπαράσταση ενός αλγορίθμου γραμμένη σε γλώσσα κατανοητή για έναν υπολογιστή. Ένα πρόγραμμα, δηλαδή, αποτελείται από μία σειρά εντολών που δίνονται στον υπολογιστή με σκοπό να εκτελέσει κάποια συγκεκριμένη λειτουργία ή να υπολογίσει κάποιο επιθυμητό αποτέλεσμα. Η εργασία σύνταξης των προγραμμάτων ονομάζεται προγραμματισμός, ενώ τα άτομα που γράφουν και συντάσσουν ένα πρόγραμμα ονομάζονται προγραμματιστές.



6. Ποιο είναι το αλφάβητο της γλώσσας μηχανής του υπολογιστή;
Στα πρώτα βήματα της ιστορίας των υπολογιστών οι άνθρωποι, για να επικοινωνήσουν με τον υπολογιστή, έπρεπε να χρησιμοποιούν μία γλώσσα που είχε ως αλφάβητο το «0» και το «1».
7. Ποια είναι τα βασικά χαρακτηριστικά μιας γλώσσας προγραμματισμού;
- a. το αλφάβητο
 - b. το λεξιλόγιο και
 - c. το συντακτικό.



8. Ποια είναι τα στάδια για την εκτέλεση ενός αλγορίθμου από την Κ.Μ.Ε;
- a. Δημιουργία Αλγορίθμου (στο χαρτί ή σε περιβάλλοντα γραφής αλγορίθμων.)
 - b. Αλγόριθμος → Πρόγραμμα (σε ολοκληρωμένο προγραμματιστικό περιβάλλον που μας παρέχουν οι σύγχρονες γλώσσες προγραμματισμού οι οποίες μάς προσφέρουν ένα φιλικό περιβάλλον, έτσι ώστε γρήγορα να αναπτύσσουμε τα προγράμματά μας.)

- c. Πρόγραμμα \rightarrow 0 και 1 (ένα πρόγραμμα – μεταφραστής που μετατρέπει τις οδηγίες μας στη μορφή που τις καταλαβαίνει ο επεξεργαστής, δηλαδή σε μια σειρά από 0 και 1 δηλαδή σε γλώσσα μηχανής).
- d. Εκτέλεση του προγράμματος στην Κ.Μ.Ε

9. Να γραφεί αλγόριθμος που να υπολογίζει (και να εμφανίζει) το εμβαδού τριγώνου.

Λύση

Πινακάκι

Δεδομένα	Επεξεργασία (τύποι)	Ζητούμενα
Τρίγωνο Βάση β Ύψος υ	$E = (\beta * \upsilon) / 2$; $\delta \delta$	Υπολογισμός Εμβαδού E Εμφάνιση Εμβαδού E

Αλγόριθμος (θυμάμαι τη γενική μορφή αλγορίθμου: είσοδος – επεξεργασία – έξοδος)

- 1° βήμα: Διάβασε β
- 2° βήμα: Διάβασε υ
- 3° βήμα: Υπολόγισε $E = (\beta * \upsilon) / 2$
- 4° βήμα: Εμφάνισε E

Σημείωση: Μετατρέψτε τον αλγόριθμο ώστε να εμφανίζει τη βάση, το ύψος και μετά το εμβαδό.

Κορυφή