

# ΕΙΣΑΓΩΓΗ ΣΤΗΝ ΕΝΝΟΙΑ ΤΟΥ ΑΛΓΟΡΙΘΜΟΥ ΚΑΙ ΤΟΥ ΠΡΟΓΡΑΜΜΑΤΙΣΜΟΥ

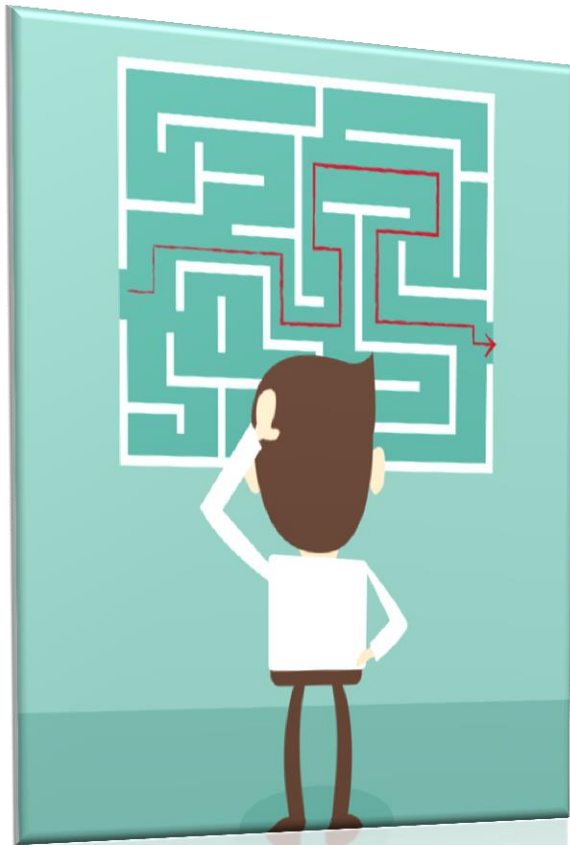
ΣΑΪΤΑΚΗ ΑΓΓΕΛΙΚΗ  
ΠΛΗΡΟΦΟΡΙΚΗ Γ' ΓΥΜΝΑΣΙΟΥ  
ΚΕΦΑΛΑΙΟ 1<sup>ο</sup>

# ΕΝΝΟΙΑ ΤΟΥ ΠΡΟΒΛΗΜΑΤΟΣ

- Ως **πρόβλημα** θεωρούμε κάθε ζήτημα που τίθεται προς επίλυση, κάθε κατάσταση που μας απασχολεί και πρέπει να αντιμετωπιστεί.
- Η λύση ενός προβλήματος δεν μας είναι γνωστή, ούτε προφανής.



# ΕΙΔΗ ΠΡΟΒΛΗΜΑΤΩΝ



Με κριτήριο τη δυνατότητα επίλυσης ενός προβλήματος διακρίνουμε 3 κατηγορίες προβλημάτων:

- **Επιλύσιμα** προβλήματα: εκείνα για τα οποία η λύση έχει βρεθεί και έχει διατυπωθεί (π.χ. λύση της β/θμιας εξίσωσης).
- **Μη επιλύσιμα** προβλήματα: εκείνα για τα οποία έχει αποδειχτεί, ότι δεν επιδέχονται λύση (π.χ. τετραγωνισμός του κύκλου).
- **Ανοικτά** προβλήματα: εκείνα για τα οποία η λύση τους δεν έχει ακόμα βρεθεί, ενώ ταυτόχρονα δεν έχει αποδειχτεί, ότι δεν επιδέχονται λύση (π.χ. θεραπεία καρκίνου).

# ΕΙΔΗ ΠΡΟΒΛΗΜΑΤΩΝ

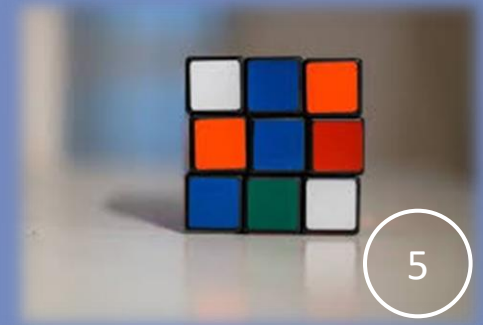
## Υπολογιστικά προβλήματα

Οποιοδήποτε πρόβλημα μπορεί να λυθεί  
και μέσω του υπολογιστή,  
χαρακτηρίζεται **υπολογιστικό πρόβλημα**  
(π.χ. μαθηματικά προβλήματα)



# ΕΠΙΛΥΣΗ ΠΡΟΒΛΗΜΑΤΟΣ

Η διαδικασία μέσω της οποίας βρίσκουμε το ζητούμενο και επιτυγχάνουμε τον επιθυμητό στόχο ονομάζεται **επίλυση προβλήματος**



# ΚΑΤΑΝΟΗΣΗ ΤΟΥ ΠΡΟΒΛΗΜΑΤΟΣ

- Να καταλάβουμε καλά το **περιεχόμενό** του
- Να διακρίνουμε τα **δεδομένα** που έχουμε στη διάθεσή μας
- Να διακρίνουμε το **ζητούμενο**
- Να προσδιορίσουμε το **«περιβάλλον»** ή το πλαίσιο μέσα στο οποίο εντάσσεται το πρόβλημα (χώρος του προβλήματος)



# ΔΕΔΟΜΕΝΑ

**Δεδομένα προβλήματος**  
είναι τα στοιχεία που  
μας είναι γνωστά και  
μπορούν να μας  
βοηθήσουν στη λύση του  
προβλήματος.

# ΖΗΤΟΥΜΕΝΟ

Σε κάθε πρόβλημα ψάχνουμε  
να βρούμε  
την απάντηση σε μια ερώτηση.  
Αυτό που ψάχνουμε είναι το  
**ζητούμενο**

# ΑΝΑΛΥΣΗ ΠΡΟΒΛΗΜΑΤΟΣ

Για να μπορέσουμε να επιλύσουμε ένα σύνθετο πρόβλημα, είναι αναγκαίο να το αναλύσουμε (σπάσουμε) σε απλούστερα προβλήματα

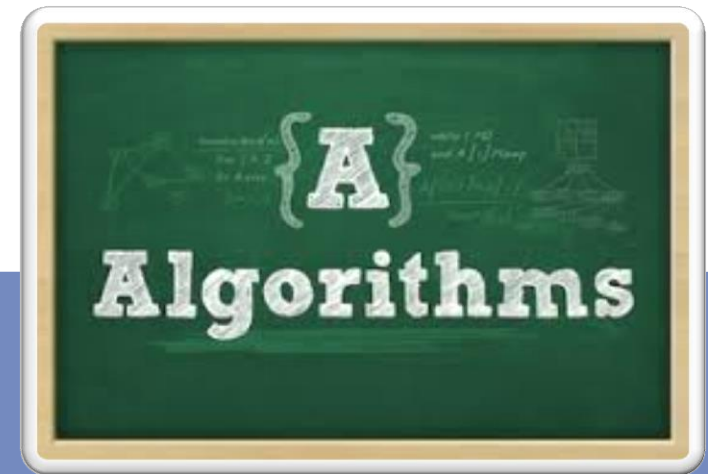


Αντιμετωπίζοντας καθένα από τα απλούστερα προβλήματα ξεχωριστά, στο τέλος θα καταφέρουμε να επιλύσουμε και το πιο πολύπλοκο πρόβλημα της «οργάνωσης σχολικής εκδρομής».



# ΑΛΓΟΡΙΘΜΟΣ

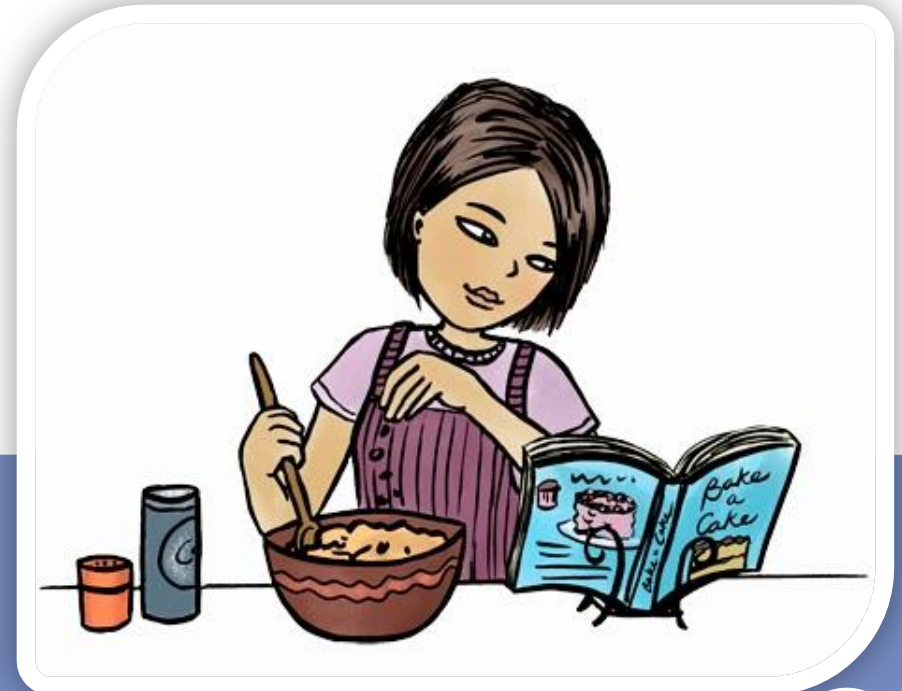
**Αλγόριθμο** ονομάζουμε τη σαφή και ακριβή περιγραφή μιας σειράς ξεχωριστών οδηγιών - βημάτων, με σκοπό την επίλυση ενός προβλήματος.



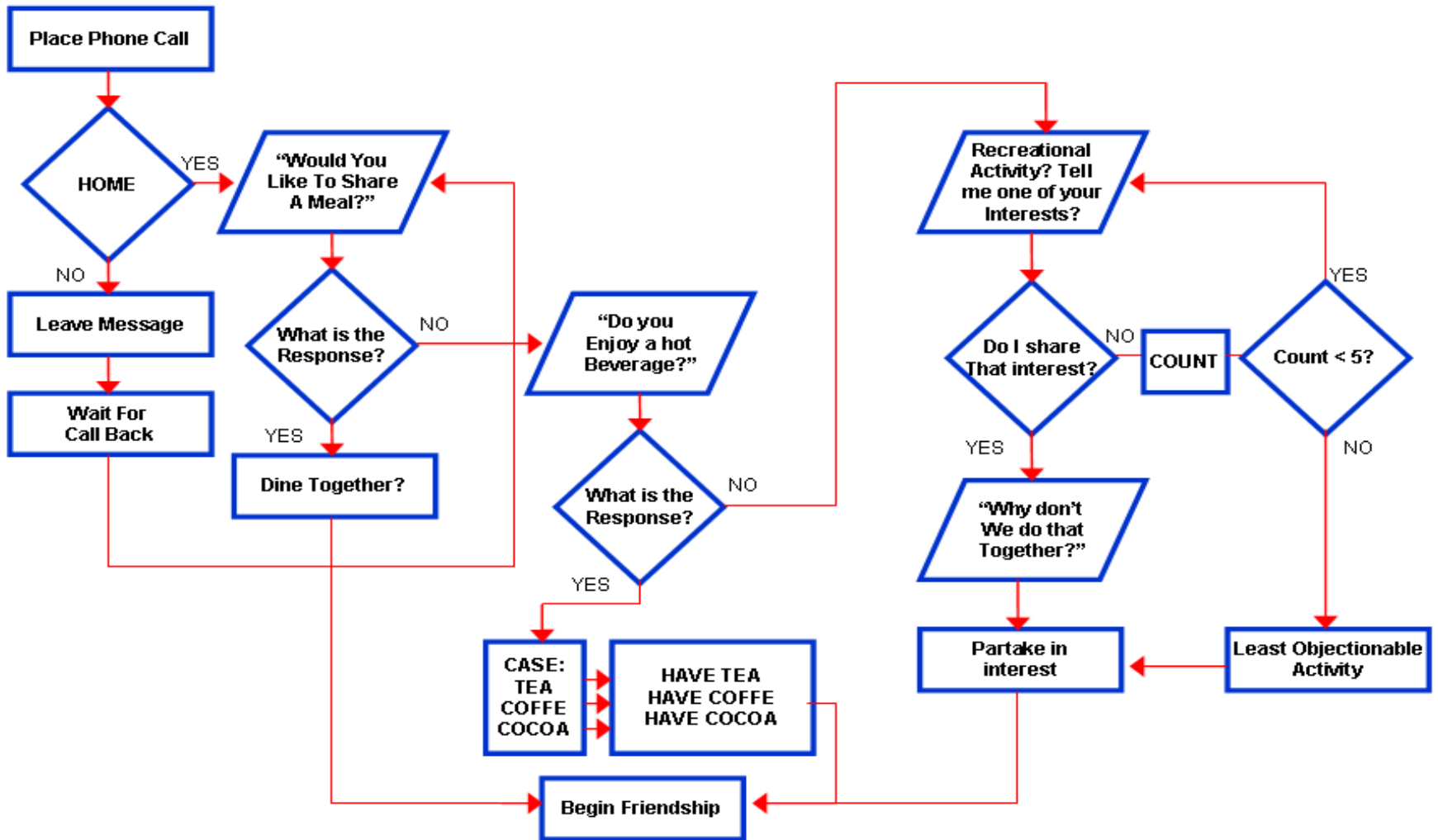
# ΠΑΡΑΔΕΙΓΜΑΤΑ ΑΛΓΟΡΙΘΜΩΝ

Βιβλίο σελ. 178 (τέλος σελίδας)

Βιβλίο σελ.181 (το κίτρινο)

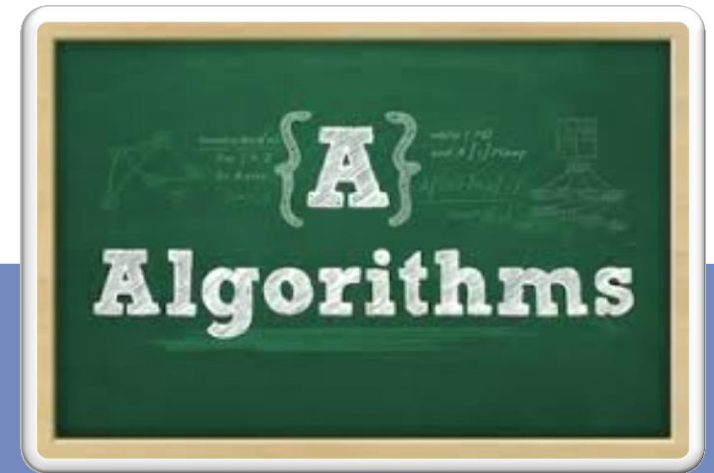


# ΠΑΡΑΔΕΙΓΜΑΤΑ ΑΛΓΟΡΙΘΜΩΝ - ΔΡ



# ΙΔΙΟΤΗΤΕΣ ΑΛΓΟΡΙΘΜΟΥ

- Οδηγίες σε **λογική σειρά**
- **Περατότητα** (ο αλγόριθμος πρέπει κάποτε να τελειώσει επιτυγχάνοντας τον αρχικό σκοπό)
- **Σαφείς** και **ακριβείς** οδηγίες (αυτός που θα εκτελέσει τον αλγόριθμο δεν πρέπει να μπερδευτεί)
- **Απλές** και **κατανοητές** οδηγίες (απλά λόγια)



# ΠΡΟΓΡΑΜΜΑ

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.

# ΠΡΟΓΡΑΜΜΑΤΙΣΜΟΣ - ΠΡΟΓΡΑΜΜΑΤΙΣΤΕΣ

Η εργασία σύνταξης των προγραμμάτων ονομάζεται **προγραμματισμός**, ενώ τα άτομα που γράφουν και συντάσσουν ένα πρόγραμμα ονομάζονται **προγραμματιστές**.

# ΓΛΩΣΣΑ ΜΗΧΑΝΗΣ

*η λειτουργία των υπολογιστών βασίζεται στην αναπαράσταση μόνο δύο ψηφίων, των «0» και «1»*

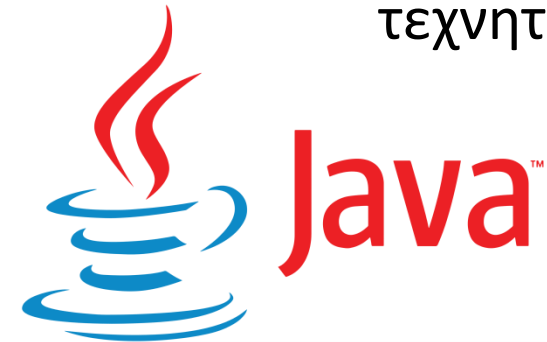
Πόσο εύκολο θα ήταν να γράφονται τα προγράμματα σε γλώσσα μηχανής;

- Δύσκολη εκμάθηση
- Αρκετά διαφορετική από τη φυσική μας γλώσσα
- Δεν είναι ενιαία σε όλους τους υπολογιστές, μια και κάθε τύπος υπολογιστή (με διαφορετικό επεξεργαστή) έχει τη δική του γλώσσα μηχανής.

# ΓΛΩΣΣΕΣ ΠΡΟΓΡΑΜΜΑΤΙΣΜΟΥ

Οι γλώσσες που «καταλαβαίνουν» οι υπολογιστές είναι τεχνητές γλώσσες που ονομάζονται **γλώσσες**

**προγραμματισμού**



PYTHON VISUAL BASIC LOGO PASCAL C++  
JAVA C COBOL ADA PASCAL LISP

WHAT IS  
PYTHON ?





# ΠΡΟΓΡΑΜΜΑ ΣΕ LOGO

ΓΙΑ ΤΕΤΡΑΓΩΝΟ

ΕΠΑΝΑΛΑΒΕ 4 [μπ 100 δε 90]

ΤΕΛΟΣ

# ΧΑΡΑΚΤΗΡΙΣΤΙΚΑ ΓΛΩΣΣΩΝ ΠΡΟΓΡΑΜΜΑΤΙΣΜΟΥ

- **Αλφάβητο** (σύνολο των χαρακτήρων που χρησιμοποιούνται από τη γλώσσα)
- **Λεξιλόγιο** (σύνολο των λέξεων που αναγνωρίζει η γλώσσα και έχουν συγκεκριμένη και μοναδική σημασία)
- **Συντακτικό** (το σύνολο των κανόνων που πρέπει να ακολουθούμε, για να συνδέουμε λέξεις σε προτάσεις)

# ΠΡΟΣΟΧΗ

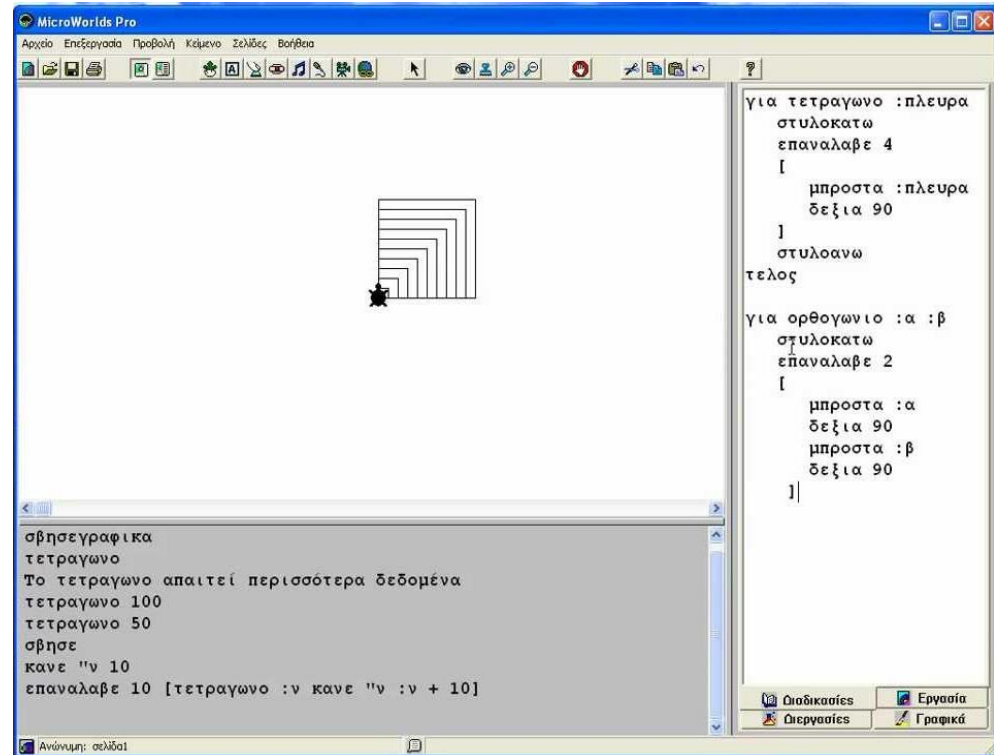
Σε μια γλώσσα προγραμματισμού η σύνδεση λέξεων δημιουργεί ολοκληρωμένες εντολές προς τον υπολογιστή.

Αν δεν ακολουθήσουμε **αυστηρά το συντακτικό** μιας γλώσσας, είναι αδύνατο για τον υπολογιστή να καταλάβει ποια εντολή του δίνουμε.

# ΟΛΟΚΛΗΡΩΜΕΝΟ ΠΡΟΓΡΑΜΜΑΤΙΣΤΙΚΟ ΠΕΡΙΒΑΛΛΟΝ

## Εργαλεία:

- **εξειδικευμένος κειμενογράφος** (σύνταξη προγράμματος)
- **πρόγραμμα-μεταφραστής** που μετατρέπει τις οδηγίες μας στη μορφή που τις καταλαβαίνει ο επεξεργαστής, δηλαδή σε μια σειρά από 0 και 1



*Η Logo χρησιμοποιεί διερμηνέα*

# ΜΕΤΑΦΡΑΣΤΕΣ

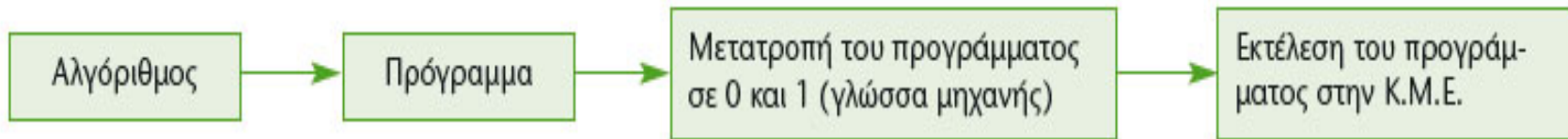
- **Μεταγλωττιστής (compilers):** ελέγχει όλο το πρόγραμμα για συντακτικά λάθη και μετά το μετατρέπει όλο σε μια κατάλληλη σειρά από 0 και 1, ώστε να μπορεί να εκτελεστεί από τον επεξεργαστή του υπολογιστή.
- **Διερμηνέας:** οι διερμηνείς (interpreters) ελέγχουν μία οδηγία κάθε φορά, την εκτελούν και μετά ελέγχουν την επόμενη οδηγία.

# ΛΑΘΗ

- **Λογικά:** Αν το αποτέλεσμα, που τελικά προκύπτει από την εκτέλεση του προγράμματος, δεν είναι το αναμενόμενο, τότε το πρόβλημα δε βρίσκεται στον τρόπο εκτέλεσης, αλλά στον αλγόριθμο που κατασκευάσαμε για τη λύση του προβλήματός μας.
- **Συντακτικά:** Αν σε κάποια οδηγία έχουμε κάνει λάθος στο αλφάβητο, στο λεξιλόγιο ή στο συντακτικό τότε το πρόγραμμα που μετατρέπει τις οδηγίες μας σε σειρά από 0 και 1 θα μας δώσει ένα κατάλληλο μήνυμα λάθους, ώστε να μας βοηθήσει να διορθώσουμε το λάθος μας



# ΣΤΑΔΙΑ ΓΙΑ ΤΗΝ ΕΚΤΕΛΕΣΗ ΕΝΟΣ ΑΛΓΟΡΙΘΜΟΥ ΑΠΟ ΤΗΝ ΚΜΕ



**Σχήμα 1.3.** Στάδια για την εκτέλεση ενός αλγορίθμου από την Κ.Μ.Ε. του υπολογιστή