



Πανεπιστήμιο Αιγαίου

Εισαγωγή στην Πληροφορική

Ενότητα 3: Προγραμματισμός ηλεκτρονικών υπολογιστών

Μιχάλης Βαΐτης
Τμήμα Γεωγραφίας



Ευρωπαϊκή Ένωση
Ευρωπαϊκό Κοινωνικό Ταμείο



ΥΠΟΥΡΓΕΙΟ ΠΑΙΔΕΙΑΣ & ΘΡΗΣΚΕΥΜΑΤΩΝ, ΠΟΛΙΤΙΣΜΟΥ & ΑΘΛΗΤΙΣΜΟΥ
ΕΙΔΙΚΗ ΥΠΗΡΕΣΙΑ ΔΙΑΧΕΙΡΙΣΗΣ

Με τη συγχρηματοδότηση της Ελλάδας και της Ευρωπαϊκής Ένωσης



ΕΥΡΩΠΑΪΚΟ ΚΟΙΝΩΝΙΚΟ ΤΑΜΕΙΟ

Άδειες Χρήσης

- Το παρόν εκπαιδευτικό υλικό υπόκειται σε άδειες χρήσης Creative Commons.
- Για εκπαιδευτικό υλικό, όπως εικόνες, που υπόκειται σε άλλου τύπου άδειας χρήσης, η άδεια χρήσης αναφέρεται ρητώς.



Χρηματοδότηση

- Το παρόν εκπαιδευτικό υλικό έχει αναπτυχθεί στα πλαίσια του εκπαιδευτικού έργου του διδάσκοντα.
- Το έργο «**Ανοικτά Ακαδημαϊκά Μαθήματα στο Πανεπιστήμιο Αιγαίου**» έχει χρηματοδοτήσει μόνο τη αναδιαμόρφωση του εκπαιδευτικού υλικού.
- Το έργο υλοποιείται στο πλαίσιο του Επιχειρησιακού Προγράμματος «Εκπαίδευση και Δια Βίου Μάθηση» και συγχρηματοδοτείται από την Ευρωπαϊκή Ένωση (Ευρωπαϊκό Κοινωνικό Ταμείο) και από εθνικούς πόρους.



Ευρωπαϊκή Ένωση
Ευρωπαϊκό Κοινωνικό Ταμείο



ΥΠΟΥΡΓΕΙΟ ΠΑΙΔΕΙΑΣ & ΘΡΗΣΚΕΥΜΑΤΩΝ, ΠΟΛΙΤΙΣΜΟΥ & ΑΘΛΗΤΙΣΜΟΥ
ΕΙΔΙΚΗ ΥΠΗΡΕΣΙΑ ΔΙΑΧΕΙΡΙΣΗΣ

Με τη συγχρηματοδότηση της Ελλάδας και της Ευρωπαϊκής Ένωσης



Σκοποί ενότητας

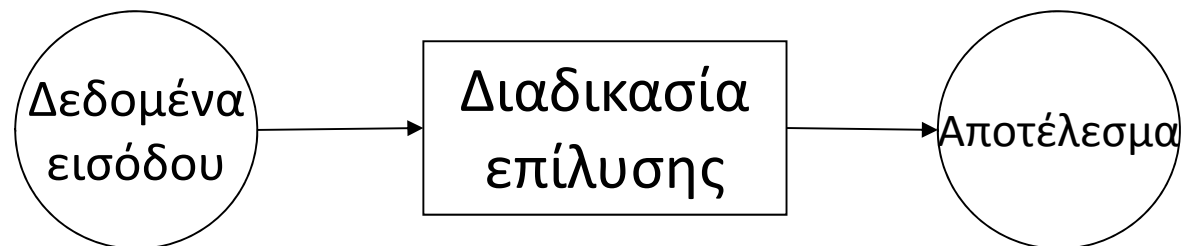
- Να γνωρίζετε την έννοια του αλγορίθμου.
- Να μπορείτε να χρησιμοποιείτε τις αλγοριθμικές δομές της διαδοχικότητας, της απόφασης και της επανάληψης για την επίλυση απλών υπολογιστικών προβλημάτων.
- Να μπορείτε να καταγράφετε έναν αλγόριθμό με λογικό διάγραμμα.
- Να γνωρίζετε τι είναι γλώσσα προγραμματισμού και τη διαδικασία κατασκευής ενός προγράμματος.
- Να μπορείτε να κατασκευάζεται προγράμματα για την επίλυση απλών υπολογιστικών προβλημάτων με τη γλώσσα προγραμματισμού R.

Αλγόριθμοι

Η Ανατομία ενός Προβλήματος

Πρόβλημα: Μια δυσάρεστη κατάσταση, η οποία επιθυμούμε να αλλάξει, ή ένα θέμα για την επίλυση του οποίου απαιτείται σκέψη

- 1. Καθορισμός απαιτήσεων προβλήματος:** Η διατύπωση του προβλήματος με τέτοιο τρόπο ώστε να βοηθήσουμε κάποιον να το κατανοήσει.
- 2. Διαδικασία επίλυσης:** Η διαδικασία μέσω της οποίας επιτυγχάνεται το αποτέλεσμα



1. Καθορισμός Απαιτήσεων του Προβλήματος

- **Η καταγραφή των παραμέτρων και των χαρακτηριστικών του προβλήματος**
 - **Δεδομένα εισόδου:** τα στοιχεία που δίνονται και είναι απαραίτητα για να λύσουμε το πρόβλημα
 - Περιέχονται στην εκφώνηση του προβλήματος
 - Είμαστε αναγκασμένοι να αναζητήσουμε από άλλες πηγές (αρχεία, βάσεις δεδομένων)
 - **Αποτέλεσμα:** το ζητούμενο ενός προβλήματος, αυτό που ψάχνουμε, το σημείο που πρέπει να φτάσουμε για να βγούμε από την προβληματική κατάσταση
 - **Μεταξύ τους σχέσεις**
- **Παράγοντες που επηρεάζουν τον καθορισμό των δεδομένων:**
 - Επιλογή των δεδομένων που έχουν άμεση σχέση με το πρόβλημα
 - Έλεγχος της ορθότητας των δεδομένων του προβλήματος
 - Πληρότητα των δεδομένων του προβλήματος

Στόχος της διαδικασίας αυτής είναι η πλήρης καταγραφή των δεδομένων και ζητούμενων του προβλήματος και η διατύπωσή τους με τέτοιο τρόπο ώστε να μην δημιουργεί παρερμηνείες και συγχύσεις σε κάποιον που θα κληθεί να το επιλύσει.

2. Διαδικασία Επίλυσης

- Προϋπόθεση: Καθορισμός απαιτήσεων προβλήματος
- Αναδιατύπωση του προβλήματος πιθανά σε «μαθηματική γλώσσα»
- Ανάλυση του προβλήματος σε απλούστερα διακριτά μέρη [υποπροβλήματα]
- Περιγραφή του τρόπου σύνδεσης των επιμέρους μερών
- Η σύνθεση των επιμέρους μερών θα πρέπει να επιλύει το όλο πρόβλημα
- Φραστική ή σχηματική παρουσίαση της δομής του προβλήματος

Στην περίπτωση επίλυσης ενός προβλήματος με τη χρήση Η/Υ, η καταγραφή του τρόπου σκέψης σε διαδοχικά βήματα είναι επιτακτική [ο υπολογιστής δεν έχει κοινή λογική!]

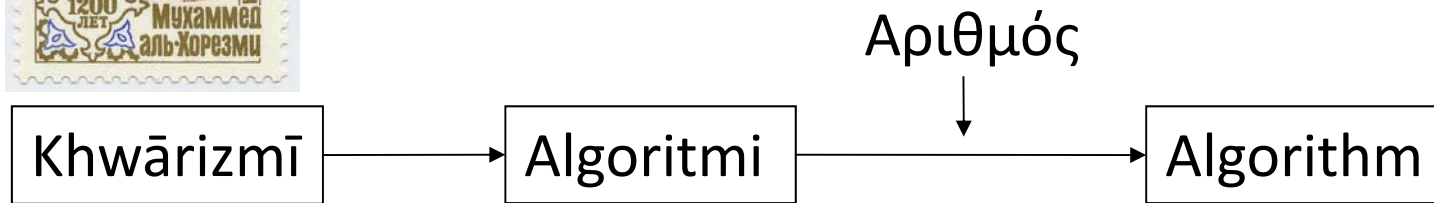
Αλγόριθμος

Πεπερασμένο σύνολο βημάτων που καθορίζουν μια σειρά από πράξεις που εκτελούνται σε πεπερασμένο χρόνο για την επίλυση μιας συγκεκριμένης κατηγορίας προβλημάτων

Αλγόριθμος



Mohammad ebne Mūsā Khwārizmī
(محمد بن موسى خوارزمی)



Ο Khwārizmī ήταν Πέρσης αστρονόμος και μαθηματικός. Γεννήθηκε περί το 780 in Khwārizm, στο σημερινό Ουζμπεκιστάν, και πέθανε περί το 850. Ο Khwārizmī αναθεώρησε τη «Γεωγραφία» του Πτολεμαίου και έγραψε πολλές εργασίες σχετικά με την αστρονομία και την αστρολογία.

Το 825 έγραψε τη διατριβή «*On Calculation with Hindu Numerals*». Η διατριβή μεταφράστηκε στα λατινικά τον 12 αι. με τον τίτλο «*Algoritmi de numero Indorum*», δηλαδή «*Ο Algoritmi για τους αριθμούς των Indians*», όπου "Algoritmi" ήταν η μετάφραση του ονόματος του συγγραφέα. Όμως από παρερμηνεία η λέξη "Algoritmi" θεωρήθηκε ο πληθυντικός της λέξης "algorithm", η οποία επικράτησε να σημαίνει «υπολογισμός». Η παρείσφρηση του "th" μάλλον οφείλεται σε λανθασμένη συσχέτιση με την ελληνική λέξη «αριθμός».

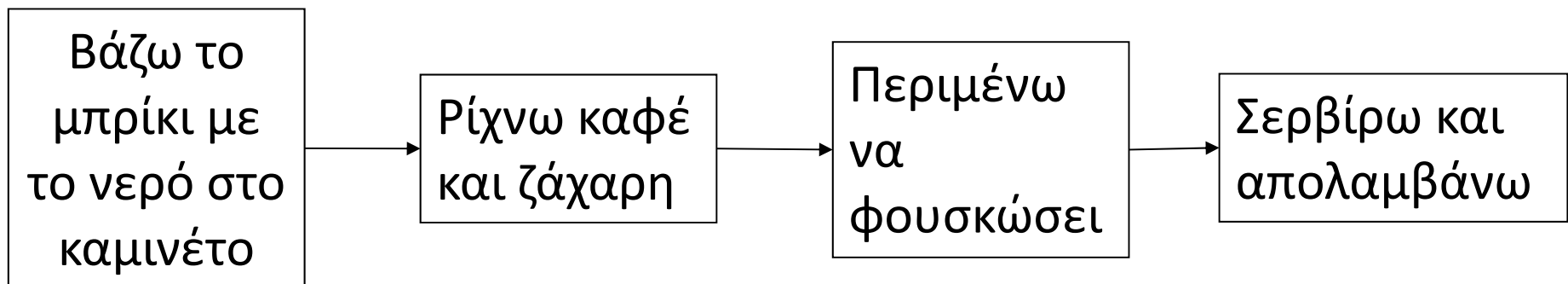
Τα δομικά στοιχεία των αλγορίθμων

- Σειριακή εκτέλεση πράξεων
- Επιλογή (ή απόφαση) ροής εκτέλεσης πράξεων
- Επανάληψη εκτέλεσης πράξεων

Τα δομικά στοιχεία των αλγορίθμων

- Σειριακή εκτέλεση πράξεων
- Επιλογή (ή απόφαση) ροής εκτέλεσης πράξεων
- Επανάληψη εκτέλεσης πράξεων

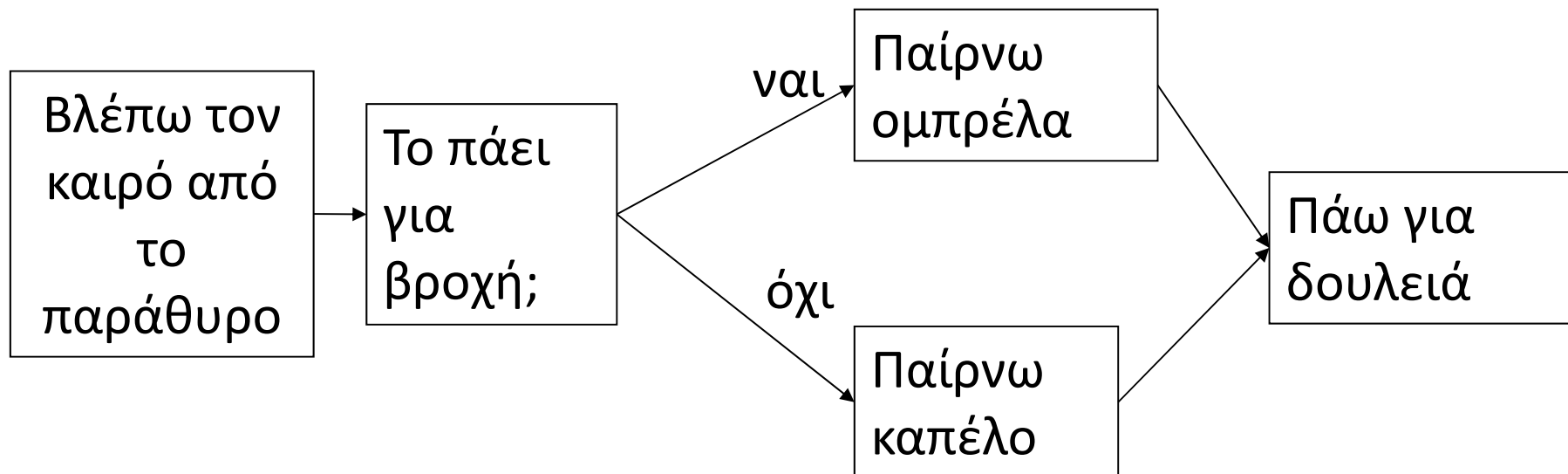
Ο αλγόριθμος του ελληνικού καφέ



Τα δομικά στοιχεία των αλγορίθμων

- Σειριακή εκτέλεση πράξεων
- Επιλογή (ή απόφαση) ροής εκτέλεσης πράξεων
- Επανάληψη εκτέλεσης πράξεων

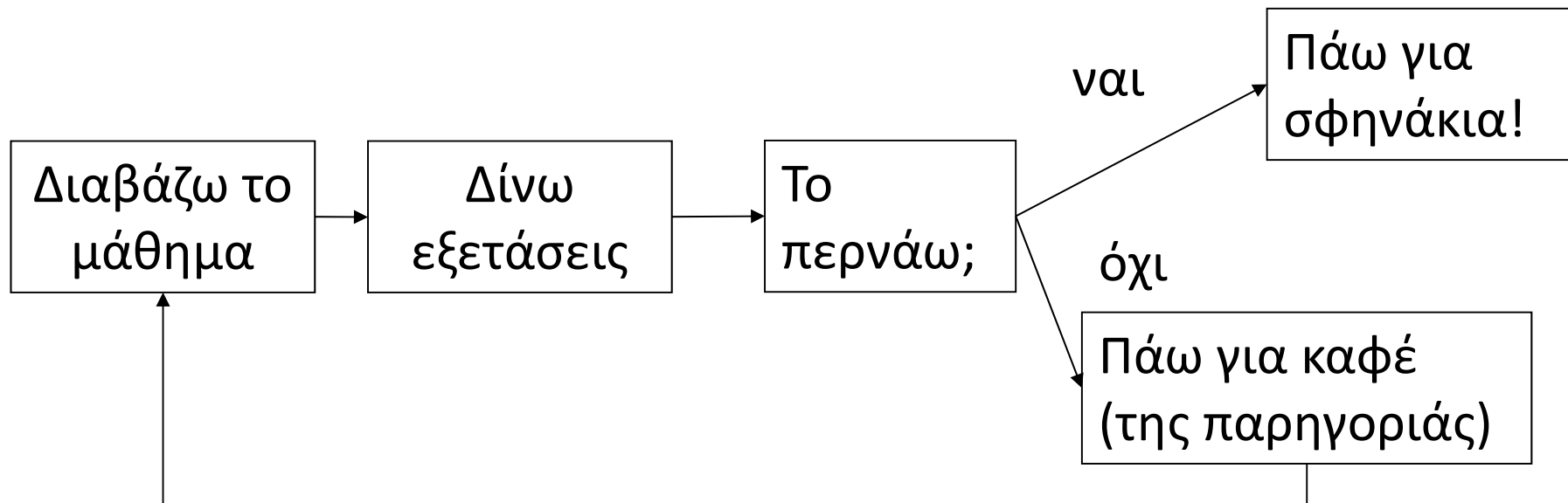
Ο αλγόριθμος της ομπρέλας



Τα δομικά στοιχεία των αλγορίθμων

- Σειριακή εκτέλεση πράξεων
- Επιλογή (ή απόφαση) ροής εκτέλεσης πράξεων
- Επανάληψη εκτέλεσης πράξεων

Ο αλγόριθμος της εξεταστικής



Εύρεση του μέγιστου κοινού διαιρέτη

Ζητείται η εύρεση του μκδ δύο θετικών αριθμών **α_1** και **α_2**

Βήμα1: Έστω **μ** ο μικρότερος εκ των α_1 και α_2

Βήμα2: Βρες τα **υπόλοιπα u_1** και **u_2** των διαιρέσεων $\alpha_1 \setminus \mu$ και $\alpha_2 \setminus \mu$

Βήμα3α: Αν τα u_1 και u_2 είναι μηδέν, τότε το ζητούμενο είναι ο αριθμός μ

Βήμα3β: Αλλιώς, μείωσε τον μ κατά 1 και επανέλαβε το Βήμα2

Παράδειγμα 1:

$$\alpha_1 = 25$$

$$\alpha_2 = 5$$

$$\mu = 5$$

$$u_1 = \text{mod}(25/5) = 0, u_2 = \text{mod}(5/5) = 0$$

$$\text{Άρα } \mu\text{κδ} = 5$$

Εύρεση του μέγιστου κοινού διαιρέτη

Παράδειγμα 2:

$$\alpha_1 = 25$$

$$\alpha_2 = 15$$

#	μ		$u_1 = \text{mod}(\alpha_1/\mu)$	$u_2 = \text{mod}(\alpha_2/\mu)$
1	$\mu = \min(\alpha_1, \alpha_2)$	15	$\text{mod}(25/15)=10$	$\text{mod}(15/15)=0$
2	$\mu = \mu - 1$	14	$\text{mod}(25/14)=11$	$\text{mod}(15/14)=1$
3	$\mu = \mu - 1$	13	$\text{mod}(25/13)=12$	$\text{mod}(15/13)=2$
...
...
10	$\mu = \mu - 1$	6	$\text{mod}(25/6)=1$	$\text{mod}(15/6)=3$
11	$\mu = \mu - 1$	5	$\text{mod}(25/5)=0$	$\text{mod}(15/5)=0$

Αλγόριθμος του Ευκλείδη

Βήμα1: Διαίρεσε τον α_1 δια α_2 και βρες το υπόλοιπο u

Βήμα2α: Αν το u είναι μηδέν, τότε το ζητούμενο είναι ο αριθμός α_2

Βήμα2β: Αλλιώς, θέσε $\alpha_1 \leftarrow \alpha_2$ και $\alpha_2 \leftarrow u$ και επανέλαβε το Βήμα1

Παράδειγμα 1:

$$\alpha_1 = 25, \alpha_2 = 5,$$
$$u = \text{mod}(25/5) = 0$$

Άρα $\text{mκδ} = 5$

Παράδειγμα 2:

$\alpha_1=25$

$\alpha_2=15$

#	α_1	α_2	u
1	25	15	$\text{mod}(25/15)=10$
2	15	10	$\text{mod}(15/10)=5$
3	10	5	$\text{mod}(10/5)=0$

Γλώσσες προγραμματισμού

Προγράμματα

- Η φυσική γλώσσα δεν είναι κατάλληλη για την περιγραφή αλγορίθμων που θα εκτελεσθούν από έναν Η/Υ, γιατί:
 - ίδιες λέξεις και ίδιες εκφράσεις επιδέχονται διαφορετικές ερμηνείες
 - μια λύση είναι περιορισμός του λεξιλογίου και χρήση καλά ορισμένων συμβόλων
- **Πρόγραμμα:** Ένας αλγόριθμος διατυπωμένος σε γλώσσα που μπορεί να καταλάβει ένας Η/Υ
- **Εντολή:** Κάθε βήμα του προγράμματος
- **Γλώσσα προγραμματισμού:** μια γλώσσα που χρησιμοποιούμε για τη συγγραφή προγραμμάτων

Γλώσσες Προγραμματισμού Χαμηλού Επιπέδου

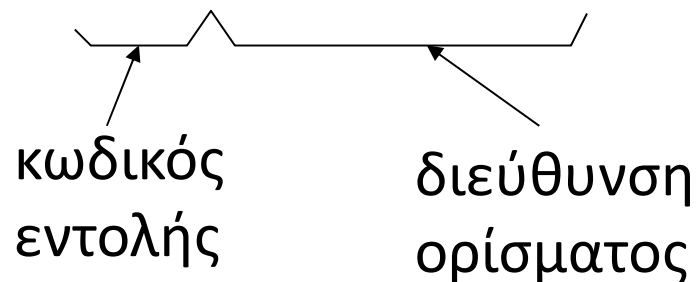
Γλώσσα Μηχανής

Συμβολική Γλώσσα

1000000000000111
1000000000001000
0100000000000111
0001000000001000
0101000000001001
1001000000001001
0000000000000000



READ 7
READ 8
LOAD 7
ADD 8
STORE 9
WRITE 9
JUMPS 0



Γλώσσες Προγραμματισμού Υψηλού Επιπέδου

- Δίνουν τη δυνατότητα να γράψουμε αλγόριθμους με τρόπο πιο 'φιλικό' και οικείο για τον άνθρωπο
- Χαρακτηριστικά των γλωσσών αυτών είναι:
 - Αποτελούνται από ένα σύνολο λέξεων, συμβόλων και προτάσεων
 - Οι εντολές που γράφονται σε γλώσσα υψηλού επιπέδου μεταφράζονται σε πολλές εντολές κώδικα μηχανής
 - Έχουν ορισμένους γραμματικούς και συντακτικούς κανόνες που πρέπει να γνωρίζει ο προγραμματιστής
 - Συνήθως η γλώσσα είναι ανεξάρτητη από τον υπολογιστή
 - Η γλώσσα περιλαμβάνει συνήθως ένα αριθμό υποπρογραμμάτων που αποτελούν τη **βιβλιοθήκη της γλώσσας**. Από τη βιβλιοθήκη αυτή ο προγραμματιστής ενσωματώνει όσα υποπρογράμματα χρειάζεται. Επίσης, μπορεί να προσθέσει δικά του υποπρογράμματα στη βιβλιοθήκη αυτή.
- Οι περισσότερο δημοφιλείς γλώσσες υψηλού επιπέδου κατατάσσονται ανάλογα με τις λειτουργικές τους ιδιότητες. Μερικές έχουν σχεδιαστεί και λειτουργούν καλύτερα σε:
 - **Επιστημονικές** εφαρμογές, όπως η FORTRAN, C και η APL
 - **Εμπορικές** εφαρμογές, όπως η COBOL και η RPG
 - **Εκπαιδευτικές** εφαρμογές, όπως η BASIC και η LOGO
 - **Ειδικές** εφαρμογές (τεχνητή νοημοσύνη), όπως η LISP και η PROLOG
 - **Επιστημονικές όσο και εμπορικές** εφαρμογές, όπως η PASCAL, η C, η C++, η Visual Basic
 - **Εφαρμογές διαδικτύου**, όπως η Java, η JavaScript, η VBscript

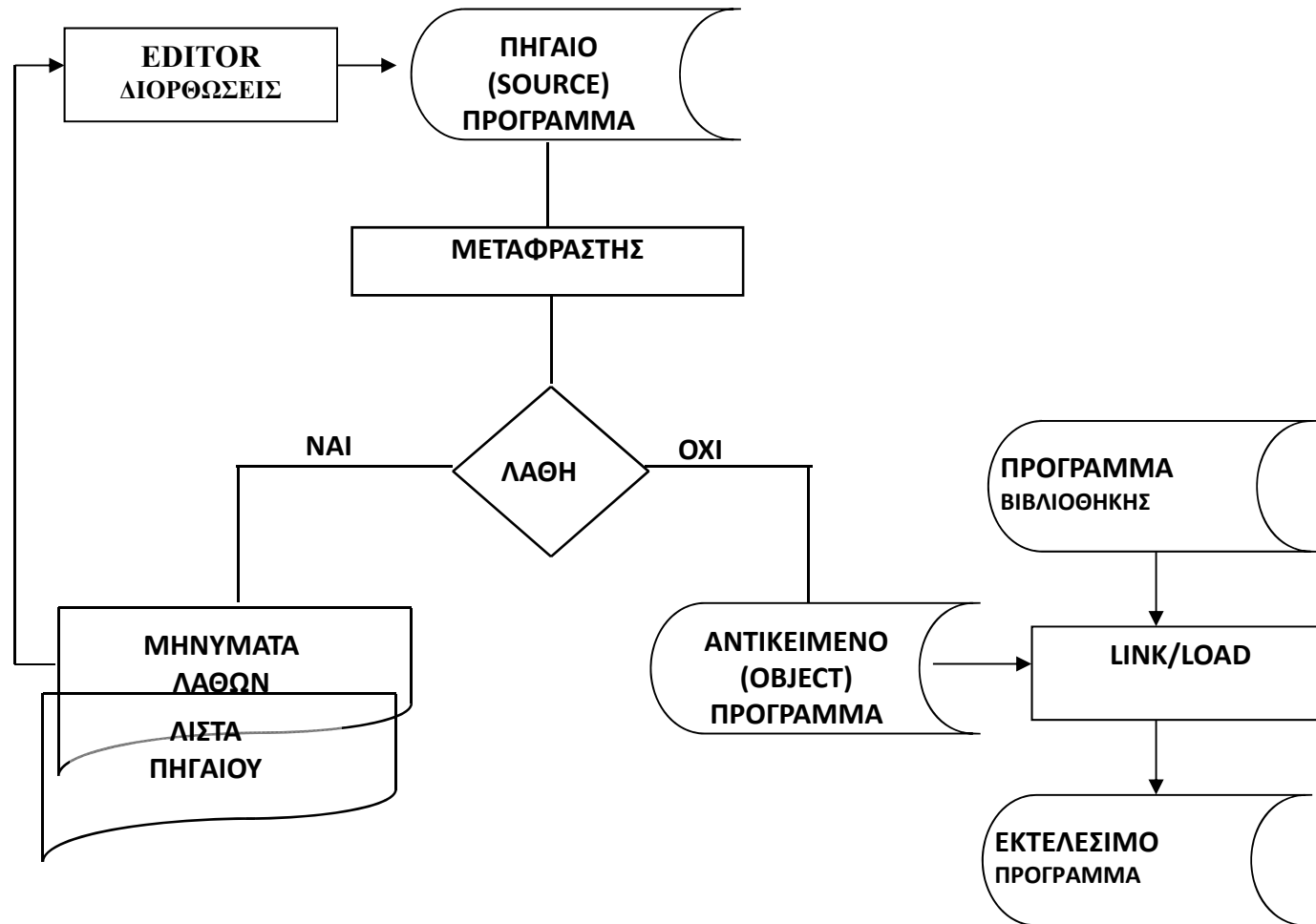
Μετάφραση

- Ο υπολογιστής μπορεί να εκτελέσει ένα πρόγραμμα τότε μόνον, εάν αυτό έχει γραφτεί σε γλώσσα μηχανής
- Για τη μετατροπή ενός προγράμματος από μια γλώσσα προγραμματισμού σε γλώσσα μηχανής απαιτείται η χρήση κάποιου μεταφραστικού προγράμματος
- Το πρόγραμμα προς μετάφραση ονομάζεται πηγαίο (source) πρόγραμμα
- Το αποτέλεσμα της μετάφρασης ονομάζεται *αντικείμενο* (object) πρόγραμμα
- **Compiler:** Μεταγλωττιστής γλώσσας υψηλού επιπέδου
 - κάνει γραμματική και συντακτική ανάλυση του πηγαίου προγράμματος με εκτύπωση των λαθών
 - κάνει την κατάληψη του απαραίτητου χώρου στην κύρια μνήμη για να μπορεί να εκτελεστεί το πρόγραμμα
 - φυλάσσει το αντικείμενο πρόγραμμα στο δίσκο
- **Interpreter:** Διερμηνέας, δηλ. μεταγλωττιστής με ταυτόχρονη εκτέλεση του προγράμματος
 - δεν ελέγχει τη σύνταξη ολόκληρου του πηγαίου προγράμματος πριν την εκτέλεση
 - επαναλαμβάνει τη διαδικασία μετάφρασης – ελέγχου κάθε φορά που εκτελείται το πηγαίο πρόγραμμα
 - κάνει γραμματικό και συντακτικό έλεγχο και μετάφραση σε κάθε εντολή που συναντάται
 - ο έλεγχος και η διόρθωση των λαθών γίνεται ευκολότερα
 - η εκτέλεση του προγράμματος είναι πιο αργή σε σχέση με την εκτέλεση του ίδιου προγράμματος, μεταφρασμένου με compiler

Συγγραφή - Εκτέλεση Προγραμμάτων

- Για την συγγραφή του πηγαίου προγράμματος αρκεί ένας απλός κειμενογράφος (editor)
- Στη συνέχεια ο μεταφραστής:
 - ελέγχει το πηγαίο πρόγραμμα για ορθογραφικά και συντακτικά λάθη
 - εφόσον υπάρχουν λάθη, ο μεταφραστής δίνει τα αντίστοιχα μηνύματα και ενεργοποιείται ο editor για τη διόρθωσή τους
 - η διαδικασία διόρθωσης επαναλαμβάνεται όσες φορές χρειαστεί μέχρι να μηδενιστούν τα λάθη
 - όταν δεν υπάρχουν λάθη, ο μεταφραστής παράγει το *αντικείμενο* πρόγραμμα
- Το *αντικείμενο* πρόγραμμα δεν είναι κατ' ανάγκη εκτελέσιμο
 - είναι απαραίτητο το αντικείμενο πρόγραμμα να συνδεθεί με υποπρογράμματα της βιβλιοθήκης της γλώσσας ή του προγραμματιστή
 - το πρόγραμμα που εκτελεί τη σύνδεση αυτή ονομάζεται *συνδέτης* (*linker*)

Προγραμματιστικό Περιβάλλον



Ολοκληρωμένο (γραφικό) περιβάλλον για την συγγραφή – μετάφραση – σύνδεση και εκτέλεση ενός προγράμματος

Τέλος Διάλεξης



Ευρωπαϊκή Ένωση
Ευρωπαϊκό Κοινωνικό Ταμείο



Με τη συγχρηματοδότηση της Ελλάδας και της Ευρωπαϊκής Ένωσης

