



ΠΑΝΕΠΙΣΤΗΜΙΟ ΑΙΓΑΙΟΥ

Εισαγωγή στον Προγραμματισμό

Ενότητα 1

Χρήστος Καλλονιάτης
Τμήμα Πολιτισμικής Τεχνολογίας και
Επικοινωνίας



Ευρωπαϊκή Ένωση
Ευρωπαϊκό Κοινωνικό Ταμείο



ΥΠΟΥΡΓΕΙΟ ΠΑΙΔΕΙΑΣ & ΘΡΗΣΚΕΥΜΑΤΩΝ, ΠΟΛΙΤΙΣΜΟΥ & ΑΘΛΗΤΙΣΜΟΥ
ΕΙΔΙΚΗ ΥΠΗΡΕΣΙΑ ΔΙΑΧΕΙΡΙΣΗΣ

Με τη συγχρηματοδότηση της Ελλάδας και της Ευρωπαϊκής Ένωσης



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



ΕΙΣΑΓΩΓΗ ΣΤΟΝ ΠΡΟΓΡΑΜΜΑΤΙΣΜΟ

(ΒΑΣΙΚΕΣ ΕΝΝΟΙΕΣ)

Καλλονιάτης Χρήστος

Επίκουρος Καθηγητής

Τμήμα Πολιτισμικής Τεχνολογίας και

Επικοινωνίας,

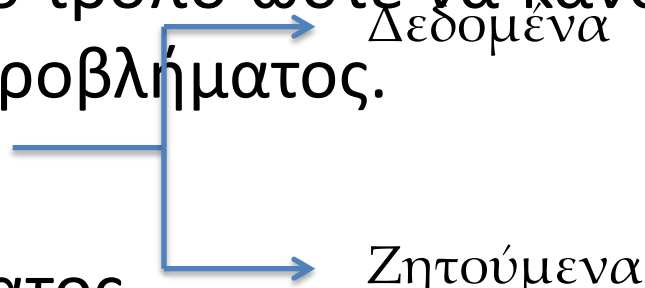
Πανεπιστήμιο Αιγαίου

<http://www.ct.aegean.gr/people/kalloniatis>

ΑΝΑΤΟΜΙΑ ΕΝΟΣ ΠΡΟΒΛΗΜΑΤΟΣ

- **Πρόβλημα:** Μια κατάσταση την οποία επιζητούμε να αλλάξουμε
- **Καθορισμός Προβλήματος:** Η διατύπωση του προβλήματος (δηλ. της κατάστασης που θέλουμε να αλλάξουμε) με τέτοιο τρόπο ώστε να κάνουμε εφικτή την κατανόηση του προβλήματος.

Καθορισμός Προβλήματος

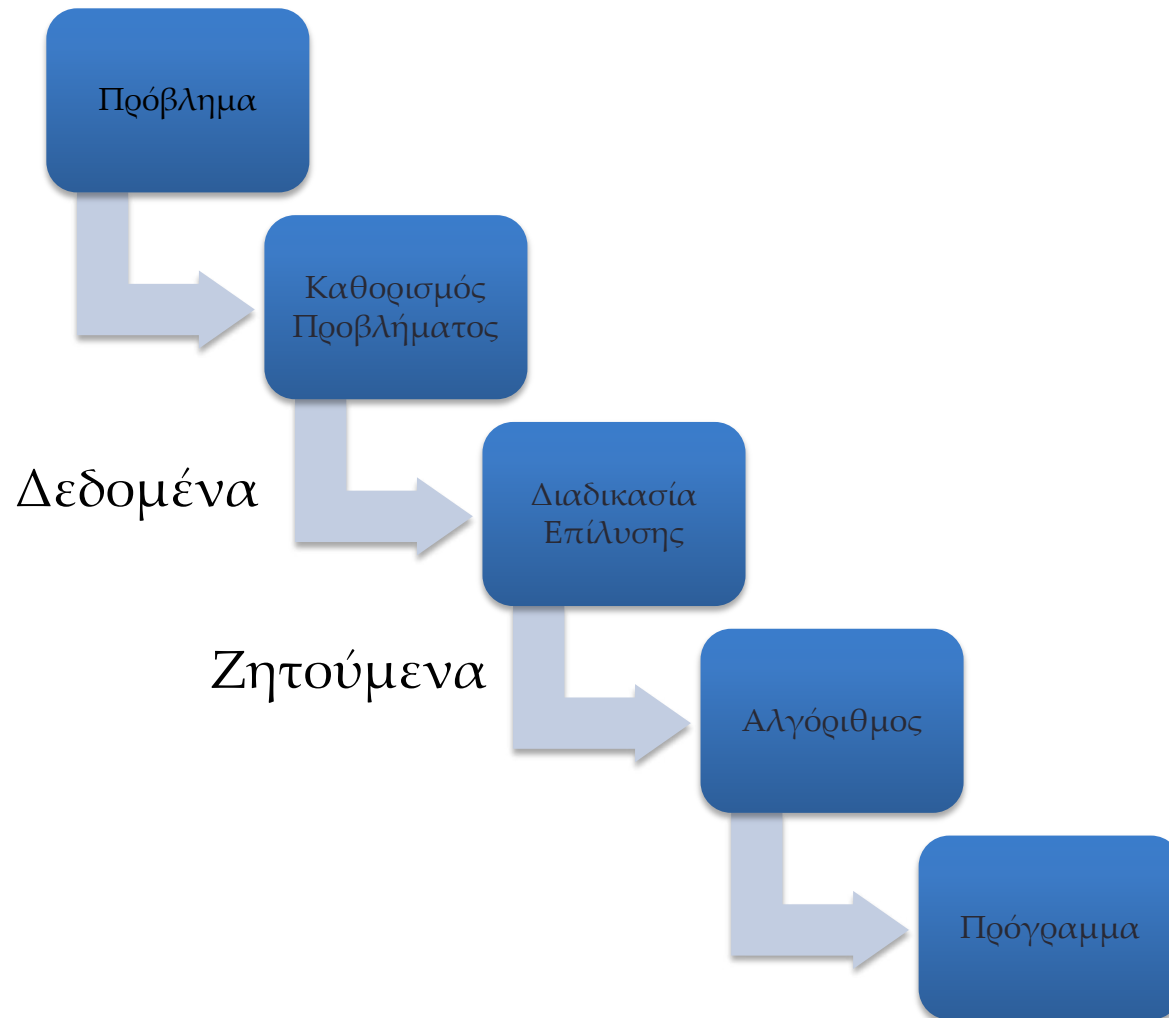


- **Διαδικασία Επίλυσης:** Η διαδικασία με βάση την οποία μεταβαίνουμε από τα δεδομένα στα₅

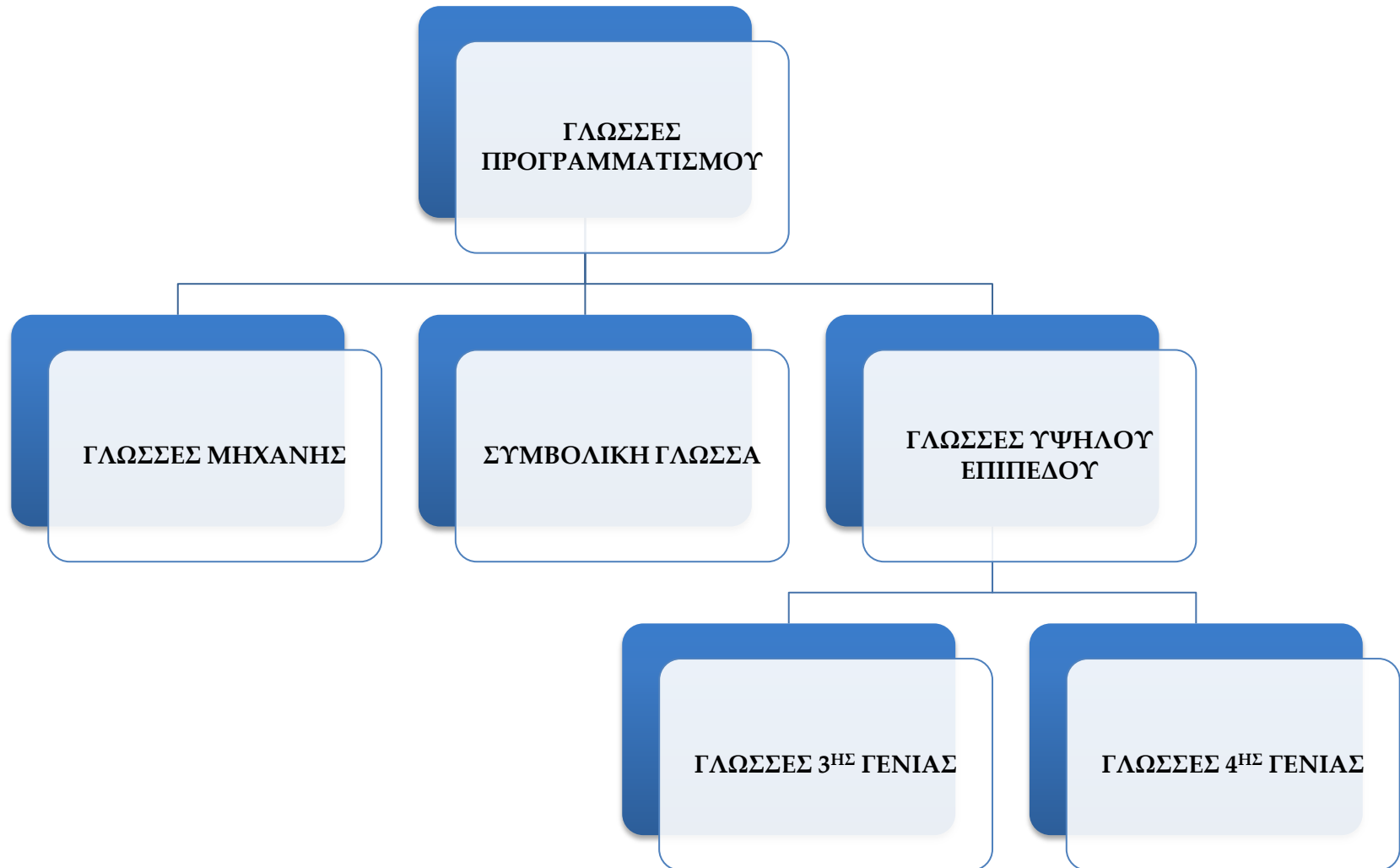
ΑΝΑΤΟΜΙΑ ΕΝΟΣ ΠΡΟΒΛΗΜΑΤΟΣ

- Η διαδικασία επίλυσης επιτυγχάνεται (υλοποιείται) με την χρήση αλγορίθμων. Οι αλγόριθμοι υλοποιούνται με τη χρήση Προγραμμάτων Η/Υ.
- Ένας αλγόριθμος υλοποιείται με τη χρήση Προγράμματος στον Η/Υ. Το πρόγραμμα υλοποιείται σε μια γλώσσα προγραμματισμού.
- Μια γλώσσα προγραμματισμού αποτελείται από ένα σύνολο κανόνων, συμβόλων και ειδικών λέξεων τα⁶

ΑΝΑΤΟΜΙΑ ΕΝΟΣ ΠΡΟΒΛΗΜΑΤΟΣ



ΓΛΩΣΣΕΣ ΠΡΟΓΡΑΜΜΑΤΙΣΜΟΥ



ΚΑΤΗΓΟΡΙΕΣ ΓΛΩΣΣΩΝ ΥΨΗΛΟΥ ΕΠΙΠΕΔΟΥ

- Επιστημονικές (C, Fortran, ADL)
- Εμπορικές (Cobol, RDG)
- Εκπαιδευτικές (Basic, Logo)
- Τεχνητής Νοημοσύνης για την κωδικοποίηση της λογικής (Lisp, Prolog)
- Επιστημονικές/Εμπορικές (P1-1, Pascal)
- Αντικειμενοστρεφής (C++, Java)
- Βάσεις Δεδομένων (SQL)
- Προγραμματισμού Πολυμέσων (Lingo)

ΜΕΤΑΓΛΩΤΤΙΣΗ – ΜΕΤΑΦΡΑΣΗ

Ο Η/Υ καταλαβαίνει μόνο τη γλώσσα μηχανής.
Για την μετατροπή ενός προγράμματος από
γλώσσα υψηλού επιπέδου σε γλώσσα μηχανής
πρέπει να γίνει η μεταγλώττιση.

ΠΡΟΓΡΑΜΜΑ ΣΕ
ΓΛΩΣΣΑ ΥΨΗΛΟΥ
ΕΠΙΠΕΔΟΥ

Μεταγλώττιση



ΠΡΟΓΡΑΜΜΑ ΣΕ
ΓΛΩΣΣΑ
ΜΗΧΑΝΗΣ

Source
file



Executable
file



COMPILER

- Κάνει συντακτική και γραμματική ανάλυση του πηγαίου κώδικα με συνολική εκτύπωση των συντακτικών λαθών
- Δεσμεύει τη μνήμη RAM που χρειάζεται για την εκτέλεση του προγράμματος
- Αποθηκεύει το Object αρχείο στο σκληρό δίσκο
- Το πρόγραμμα εκτελείται στο τέλος της όλης διαδικασίας

INTERPRETER

- Το πρόγραμμα εκτελείται ταυτόχρονα με τη μετάφραση. Δεν ελέγχεται η σύνταξη ολόκληρου του πηγαίου προγράμματος απευθείας
- Επαναλαμβάνει τη διαδικασία μετάφρασης-ελέγχου κάθε φορά που εκτελείται το πηγαίο πρόγραμμα
- Διενεργεί συντακτικό έλεγχο και μετάφραση σε κάθε μία εντολή ξεχωριστά (και όχι συνολικά σε όλο το πρόγραμμα)
- Αργεί εκτέλεση του προγράμματος

ΠΡΟΓΡΑΜΜΑΤΙΣΤΙΚΟ ΠΕΡΙΒΑΛΛΟΝ

- Το προγραμματιστικό περιβάλλον μιας γλώσσας είναι το λογισμικό της, το οποίο έχει τουλάχιστον τις παρακάτω λειτουργίες:
- **Editor:** Περιβάλλον στο οποίο συγγράφεται ο πηγαίος κώδικας. Αποθήκευση του αρχείου με το extension της γλώσσας
- **Compiler/Interpreter:** Για τη μετάφραση του πηγαίου κώδικα και τη δημιουργία του object αρχείου
- **Build:** Για τη δημιουργία του exe αρχείου
- **Run:** Για την εκτέλεση του exe αρχείου

ΔΟΜΗ ΠΡΟΓΡΑΜΜΑΤΟΣ ΣΤΗ PASCAL

Program *Όνομα Προγράμματος;*

Τμήμα Δηλώσεων, Ορισμοί Υποπρογραμμάτων

Const

Type

Var

Procedure/Function

Begin

Κυρίως Πρόγραμμα

End.

ΔΗΛΩΣΕΙΣ ΚΑΙ ΟΡΙΣΜΟΙ

- Στο τμήμα αυτό ορίζονται και δηλώνονται όλες οι μεταβλητές και οι σταθερές που χρησιμοποιούνται στο πρόγραμμα.
- *Ορισμός Μεταβλητής-Σταθερών* σημαίνει τη δήλωση του **ονόματος** της εκάστοτε μεταβλητής ή σταθεράς.
- *Δήλωση Μεταβλητών* σημαίνει να δηλωθεί ο **τύπος** της μεταβλητής. Στις σταθερές δεν δηλώνεται ο τύπος.

ΜΕΤΑΒΛΗΤΕΣ

- Κάθε μεταβλητή έχει ένα όνομα και έναν τύπο.
- Η τιμή της μεταβλητής μπορεί να αλλάζει μέσα στο πρόγραμμα.
- Το όνομα της μεταβλητής αποτελείται από μια ακολουθία γραμμάτων και αριθμών. Συνηθίζουμε να χρησιμοποιούμε ονόματα σχετικά με τη λειτουργία της εκάστοτε μεταβλητής.
- Κάθε μεταβλητή αντιστοιχεί σε μία συγκεκριμένη θέση μνήμης η οποία αντιστοιχεί στην τιμή που παίρνει η μεταβλητή κάθε φορά.
- Άρα, η τιμή της μεταβλητής είναι στην ουσία το ₁₆ περιεχόμενο της θέσης μνήμης που αντιστοιχεί στο

ΙΔΙΟΤΗΤΕΣ ΜΕΤΑΒΛΗΤΩΝ - ΣΤΑΘΕΡΩΝ

- Όνομα
- Τύπος
- Θέση Μνήμης
- Τιμή (Στις σταθερές δεν αλλάζει)

ΜΕΤΑΒΛΗΤΕΣ

$X := 5;$

5

$X := 12;$

12

ΤΥΠΟΙ ΔΕΔΟΜΕΝΩΝ

- Ακέραιοι (Integer)
 - 2, 3, 7, 8, -20
- Πραγματικοί (Real)
 - 3.5, 4.0, 7.3, -12.8
- Αλφαριθμητικοί (String)
 - ‘123’, ‘#abcd1234)’, ‘John’
- Λογικοί (Boolean)
 - True False
- Χαρακτήρες (Character)
 - ‘a’, ‘\$', ‘1’

ΔΗΛΩΣΗ ΣΤΑΘΕΡΩΝ

```
const pi=3.14;  
    o='*';  
    win=true;  
    i=false;  
    k=10;  
    i=20;  
    r=8.1727;  
    s='george';
```

ΔΗΛΩΣΗ ΜΕΤΑΒΛΗΤΩΝ

```
var    x: integer;  
      cost ,γ, z: real;  
      w, name: char;  
      condition, t: boolean;
```

ΕΝΤΟΛΕΣ

- Εντολές Εισόδου-Εξόδου
 - read/readln
 - write/writeln
- Εντολή Εκχώρησης
 - number := 50;
- Εντολές Ελέγχου
 - If..Then..Else
- Εντολές Επανάληψης
 - For, While, Repeat

ΕΠΙΤΡΕΠΤΑ ΣΥΜΒΟΛΑ

- Αριθμητικοί Τελεστές

—+, - , * , / , div, mod

- Συγκριτικοί Τελεστές

—= , < , > , <= , >= , <>

- Λοιπά Σύμβολα

—: , . , , , := , (,) , [,]

DIV και MOD

- $\text{Div}(ide)$: Ακέραιο πηλίκο μιας διαίρεσης μεταξύ δύο ακεραίων αριθμών
- $\text{Mod}(ulo)$: Το ακέραιο υπόλοιπο μίας διαίρεσης μεταξύ δύο ακεραίων αριθμών

- $20 \text{ div } 3 = 6$
- $20 \text{ mod } 3 = 2$

ΔΕΣΜΕΥΜΕΝΕΣ ΛΕΞΕΙΣ

AND, ABS, ARRAY, ARCTAN, BEGIN, BOOLEAN, CASE, CONST, CHAR, COS, DIV, DO, DISPOSE, DOWNTON, ELSE, EOF, EOWL, END, FILE, FALSE, FOR, FORWARD, FUNCTION, GET, GOTO, IF, IN, LABEL, INTEGER, INPUT, MOD, MAXINT, MININT, NIL, NOT, OF, NEW, OUTPUT, OR, PACKED, PRED, PACK, PAGE, PRECEDURE, PROGRAM ,REAL, REWRITE, REWIND, RESET, READ, READLN, ROUND, REPEAT, SET, SIN, SQR, SQRT, SUCC, TRUNC, THEN, TO, TYPE, TEXT, TRUE, UNTIL, UNPAC, VAR, WRITE, WRITELN, WHILE, WITH

ΙΕΡΑΡΧΙΑ ΠΡΑΞΕΩΝ

- ()
- *, /, div, mod
- +, -

ΠΑΡΑΔΕΙΓΜΑΤΑ ΠΡΑΞΕΩΝ

$(a+b)c$	$(a+b)*c$
$\frac{ab}{c}$	$a*b/c$
$\frac{a+b}{c}$	$(a+b)/c$
$x[y+w(3-v)]$	$y*(y+w*(3-v))$
$\frac{y+2}{x+4}$	$(y+2)/(x+4)$

ΣΗΜΕΙΩΣΕΙΣ - ΕΡΓΑΣΤΗΡΑ

- www.ct.aegean.gr/people/kalloniatis
- chkallon@aegean.gr
- <http://eclass.aegean.gr>
- Ώρες Γραφείου
 - Δευτέρα και Παρασκευή 10:00-12:00