

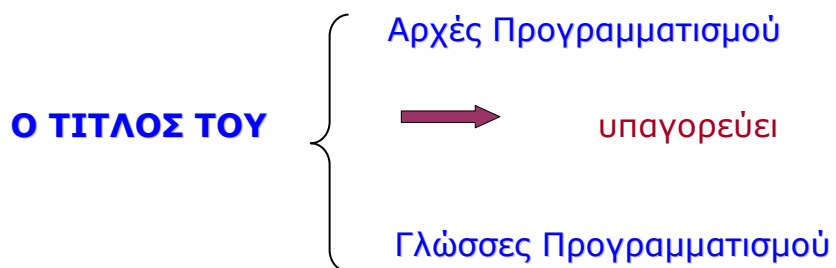
**Π. Σταθοπούλου**

psth@ece.upatras.gr ή  
psth@upatras.gr

Ομάδα Α' (Φοιτητές με μονό αριθμό Μητρώου )

- Διδασκαλία : Παρασκευή 11πμ-13μμ ΗΛ7
- Φροντιστήριο : Δευτέρα 11πμ-12πμ ΗΛ4

## **ΠΕΡΙΕΧΟΜΕΝΟ ΤΟΥ ΜΑΘΗΜΑΤΟΣ**



## Τι είναι όμως Προγραμματισμός ?

**Ορισμός:** Δραστηριότητα επίλυσης απλών ή σύνθετων προβλημάτων μέσω υπολογιστή

Ικανότητα της σωστής επίλυσης των προβλημάτων



Δυνατότητα σωστού Προγραμματισμού

### **ΟΜΩΣ**

- ❑ Ο Προγραμματισμός δεν είναι μόνο **Επιστήμη** αλλά και **ΤΕΧΝΗ**
- ❑ Προγραμματίζω σωστά δεν σημαίνει απομνημονεύω μερικούς κανόνες

## Επίλυση Προβλημάτων & Υπολογιστής

Λόγοι επίλυσης προβλημάτων μέσω υπολογιστή

- Πολυπλοκότητα προβλημάτων
- Επαναληπτικότητα διεργασιών
- Ταχύτητα των διεργασιών
- Μεγάλο πλήθος δεδομένων

## Στάδια επίλυσης προβλημάτων μέσω υπολογιστή

- Καθορισμός των προδιαγραφών του προβλήματος
- Ανάλυση του προβλήματος
- Σχεδιασμός του αλγορίθμου για την επίλυση του προβλήματος
- Υλοποίηση του αλγορίθμου
- Έλεγχος και επαλήθευση του

## Γιατί Βασικές Αρχές Προγραμματισμού ?

Τι σημαίνει βασικές αρχές προγραμματισμού



Στρατηγικές & μέθοδοι ανάπτυξης λογισμικού

## Γιατί Βασικές Αρχές Προγραμματισμού ?

- εφαρμογή βασικών αρχών προγραμματισμού συνεπάγεται:
  - σωστή διαχείριση της πολυπλοκότητας
- εκμάθηση βασικών αρχών προγραμματισμού:
  - πρακτική εξάσκηση & συστηματική ενασχόληση με τον προγραμματισμό
  - πειθαρχία κατά την ανάπτυξη & υλοποίηση προγραμμάτων

## Γιατί Βασικές Αρχές Προγραμματισμού ?

Στρατηγικές ή μέθοδοι προγραμματισμού:

- 1.αφαιρετικότητα (abstraction)
- 2.έλεγχος της ορθής λειτουργίας
- 3.δυνατότητα διασφάλισης ποιότητας
- 4.δυνατότητα επέκτασης ή τροποποίησης
- 5.εκμετάλλευση δομικών στοιχείων (reusability)

Στρατηγικές & μέθοδοι υλοποίησης οποιουδήποτε έργου μηχανικού

## Τρέχουσα Κατάσταση

Ποιες είναι οι μέχρι τώρα γνώσεις μας;

- 1<sup>ο</sup> εξάμηνο {
  - βασικά στοιχεία οργάνωσης των υπολογιστών
  - γενικά περι αλγορίθμων
  
- 2<sup>ο</sup> εξάμηνο {
  - συμφιλίωση με μια γλώσσα προγραμματισμού  
**(FORTRAN)**
  - ανάπτυξη & υλοποίηση προγραμμάτων

## Τρέχουσα Κατάσταση

➤ Το μάθημα τώρα στο 4<sup>ο</sup> εξάμηνο

Κατανόηση δύο βασικών στρατηγικών προγραμματισμού  
Διαδικαστικός (procedural) προγραμματισμός  
Αντικειμενοστραφής (object-oriented) προγραμματισμός  
Συμφιλίωση με σύγχρονες γλώσσες (C, JAVA)

Πως ;

- Διαδικαστικός (procedural) προγραμματισμός
  - Εφαρμογές με γλώσσα C
- Αντικειμενοστραφής (object-oriented) προγραμματισμός
  - Εφαρμογές με γλώσσα Java

## Σήμερα

### ➤ Εισαγωγικά

Διαδικαστικός προγραμματισμός, Διεργασία,  
Οργάνωση της Λύσης, Διάγραμμα ροής,  
Αλγόριθμος, Αναπαράσταση Δεδομένων,

### ➤ Δομή προγράμματος σε C

Συνάρτηση main, Δηλώσεις μεταβλητών,  
Εκτελέσιμες προτάσεις.

### ➤ Εργαστήριο 1ο

## Διαδικαστικός προγραμματισμός

Ο Διαδικαστικός (*procedural*) προγραμματισμός συχνά χρησιμοποιείται σαν συνώνυμο του προστακτικού (*imperative*) προγραμματισμού

{όσον αφορά τα βήματα που πρέπει να κάνει το πρόγραμμα για να οδηγηθεί στο τελικό αποτέλεσμα}

Βασίζεται, όμως κυρίως στην έννοια της

κλήσης μια Διεργασίας/Διαδικασίας  
(*Procedure Call*)

## Διαδικαστικός προγραμματισμός

- ✓ Οι Διεργασίες/Διαδικασίες, {γνωστές σαν ρουτίνες, υπορουτίνες, μέθοδοι, συναρτήσεις} αποτελούν ένα σύνολο από υπολογιστικά βήματα.
- ✓ Κάθε διεργασία/διαδικασία μπορεί να κληθεί από οποιοδήποτε σημείο της εκτέλεσης του προγράμματος και μπορεί να εμπεριέχεται σε άλλες διεργασίες ή ακόμα και στον ίδιο τον εαυτό της

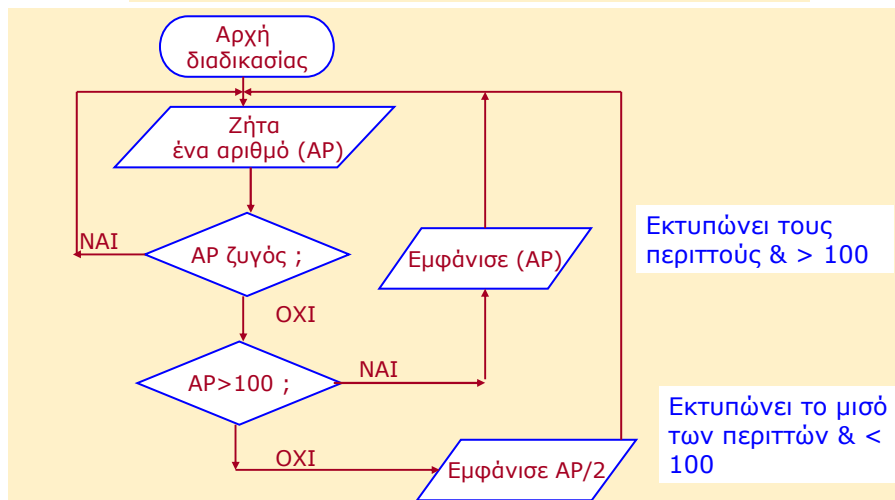
## Οργάνωση της Λύσης



## Τρόποι Περιγραφής/Απεικόνισης της Λύσης

- Δομημένη Φυσική Γλώσσα / Ψευδοκώδικας
- Διαγράμματα ροής
- UML, SDL, .....

## Διάγραμμα Ροής





## Σύνταξη Προγραμμάτων

Παλιότερα:

Αλγόριθμοι+ Δομές δεδομένων = Πρόγραμμα

Σήμερα:

Τεχνικές διαχείρισης της πολυπλοκότητας

## Τι είναι Αλγόριθμος

Αλγόριθμος είναι μια πεπερασμένη σειρά ενεργειών, αυστηρά καθορισμένων και εκτελέσιμων σε πεπερασμένο χρόνο που στοχεύουν στην επίλυση ενός προβλήματος.

## Τι είναι Αλγόριθμος

Κάθε αλγόριθμος ικανοποιεί τα παρακάτω κριτήρια:

- καμία ή περισσότερες εισόδους
- μία ή περισσότερες εξόδους
- Καθοριστικότητα
- Περαιτότητα
- Αποτελεσματικότητα

## Αναπαράσταση δεδομένων

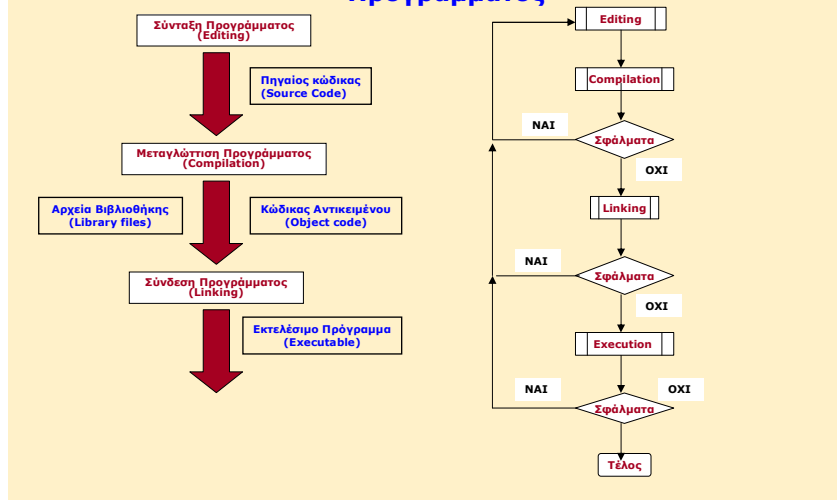
Τρόπος αναπαράστασης καθοριστικός :

- Απλοί και Σύνθετοι
- Σύνθετες δομές δεδομένων
  - Δένδρα , λίστες
- Αφαιρετικότητα στα δεδομένα

## Προετοιμασία για Προγραμματισμό

- Ορισμός του Προβλήματος
- Κύκλος Ανάπτυξης του Προγράμματος
  - Χρήση Προγράμματος Σύνταξης (Editing)
  - Δημιουργία Πηγαίου Κώδικα (Source Code)
  - Μεταγλώττιση Πηγαίου Κώδικα (Compilation)
  - Σύνδεση για την δημιουργία του εκτελέσιμου αρχείου (Linking)
  - Δημιουργία εκτελέσιμου Κώδικα (Executable Code)
- Ολοκλήρωση του Κύκλου Ανάπτυξης – Εκτέλεση

## Διαγραμματική Παράσταση του Κύκλου Ανάπτυξης του Προγράμματος



## Τι είναι γλώσσα Προγραμματισμού?

- Γενικά η **γλώσσα προγραμματισμού** είναι μία τεχνητή γλώσσα η οποία μπορεί να χρησιμοποιηθεί για να ελέγχει την συμπεριφορά μιας μηχανής, ειδικότερα ενός υπολογιστή.
- Οι γλώσσες προγραμματισμού, όπως και οι φυσικές γλώσσες, προσδιορίζονται από **ΣΥΝΤΑΚΤΙΚΟΥΣ** και **σημασιολογικούς** κανόνες.

## Τα χαρακτηριστικά της γλώσσας C

- Γλώσσα Προγραμματισμού υψηλού επιπέδου με αρκετά χαρακτηριστικά των γλωσσών χαμηλού επιπέδου.
- Δομημένη Γλώσσα Προγραμματισμού, επιτρέπει τον τμηματικό χειρισμό (modularity) των προγραμμάτων.
- Μεταφραστική γλώσσα

## Στοιχεία της Γλώσσας C

- Διαθέτει Λεξιλόγιο , Γραμματική & Συντακτικό
- Διαθέτει Τύπους Δεδομένων
  - Απλοί τύποι δεδομένων (double, int, char, float)
  - Σύνθετοι τύποι δεδομένων (πίνακες, δομές)
- Διαθέτει Βασική Βιβλιοθήκη
  - Έτοιμες συναρτήσεις (printf, scanf, abs, power,.....)

## Στοιχεία της Γλώσσας C

- Λεξιλόγιο
  - Σύνολο από σύμβολα & λέξεις που έχουν προκαθορισμένη σημασία
- Συντακτικό
  - Σύνολο κανόνων που καθορίζουν την σωστή σύνταξη προτάσεων

## Τα αλφάβητο της γλώσσας C

- αποτελείται σύμφωνα με το ANSI 88 πρότυπο από **96** **χαρακτήρες**:

- οι χαρακτήρες:

- **κενού**, ελέγχου οριζόντιου στηλοθέτη (**tab**), κάθετου στηλοθέτη,
- αλλαγής σελίδας (**form feed**), νέας γραμμής (**new-line**)

- οι παρακάτω 91 γραφικοί χαρακτήρες:

**a b c d e f g h i j k l m n o p q r s t u v w x y z**  
**A B C D E F G H I J K L M N O P Q R S T U V W X Y**  
**Z**  
**0 1 2 3 4 5 6 7 8 9**  
**\_ { } [ ] # ( ) < > % : ; . ? \* + - / ^ & | ~ ! = , \ " ' `**

## Το Λεξιλόγιο της Γλώσσας C

- Λέξεις κλειδιά (Keywords)
- Δεσμευμένες λέξεις (Reserved words)
- Τελεστές (Operators)
- Αναγνωριστές (Identifiers)

## Λέξεις Κλειδιά (keywords)

➤ Λεκτικές μονάδες που μόνες τους ή με άλλες λεκτικές μονάδες σχηματίζουν κάποια γλωσσική κατασκευή.

Π.χ.

- η λέξη κλειδί **int** της C, αναπαριστά τον ακέραιο τύπο δεδομένων,
- η **if** σε συνδυασμό με την λέξη κλειδί **else** δημιουργεί την βασική γλωσσική κατασκευή **if else** για τον έλεγχο της ροής του προγράμματος

## Λέξεις Κλειδιά στη C

|                 |               |                 |                 |
|-----------------|---------------|-----------------|-----------------|
| <b>auto</b>     | <b>double</b> | <b>int</b>      | <b>struct</b>   |
| <b>break</b>    | <b>else</b>   | <b>long</b>     | <b>switch</b>   |
| <b>case</b>     | <b>enum</b>   | <b>register</b> | <b>typedef</b>  |
| <b>char</b>     | <b>extern</b> | <b>return</b>   | <b>union</b>    |
| <b>const</b>    | <b>float</b>  | <b>short</b>    | <b>unsigned</b> |
| <b>continue</b> | <b>for</b>    | <b>signed</b>   | <b>void</b>     |
| <b>default</b>  | <b>goto</b>   | <b>sizeof</b>   | <b>volatile</b> |
| <b>do</b>       | <b>if</b>     | <b>static</b>   | <b>while</b>    |

καθοριστής τύπου - καθοριστής κατηγορίας αποθήκευσης - προσδιοριστής τύπου

## Δεσμευμένες λέξεις (Reserved words)

- Λέξεις κλειδιά (Keywords)
- Ονόματα συναρτήσεων της βασικής βιβλιοθήκης
- Ονόματα μακροεντολών που ορίζονται σε αρχεία επικεφαλίδας
- Ονόματα τύπων που ορίζει η βασική βιβλιοθήκη
- Ονόματα εντολών προεπεξεργαστών: `include`, `define`
- Ονόματα της μορφής `_DATE_`, `_FILE_`, κτλ.

## Τελεστής (Operators)

- Είναι ένα σύμβολο ή μία λέξη που αναπαριστά συγκεκριμένη διεργασία, που εκτελείται πάνω σε ένα ή περισσότερα δεδομένα

Π.χ.

- Ο τελεστής `*`, αναπαριστά την διεργασία του πολλαπλασιασμού.

( `/`, `+`, `=`, `==`, `!=` )



## Αναγνωριστές (Identifiers)

- λέξεις που κατασκευάζει ο προγραμματιστής
- χρησιμοποιούνται σαν ονόματα από τον προγραμματιστή
  - Μεταβλητές
  - Σταθερές
  - Συναρτήσεις
  - δικούς του τύπους δεδομένων
- δεν θα πρέπει να είναι δεσμευμένες

## Παραδείγματα Αναγνωριστών

|              |        |        |
|--------------|--------|--------|
| ■ J          | —————> | έγκυρο |
| ■ MyName     | —————> | έγκυρο |
| ■ Int        | —————> | έγκυρο |
| ■ Get_word() | —————> | έγκυρο |
| ■ Float      | —————> | έγκυρο |
| ■ char       | —————> | άκυρο  |

## Οι τύποι Δεδομένων στη C

- char - χαρακτήρας
- int - ακέραιος
- float - αριθμός κινητής υποδιαστολής απλής ακρίβειας(32 bits)
- double - αριθμός κινητής υποδιαστολής απλής ακρίβειας(64 bits)
- Απαριθμητικός τύπος
  - enum boolean {false, true}
- Σύνθετοι τύποι
  - Πίνακες & δομές

## Διαδικασία απόδοσης ονομάτων

- Ένας από τους σημαντικότερους παράγοντες που καθορίζουν την λειτουργεί του προγράμματος είναι η **αναγνωσιμότητα**.
- Καλό πρόγραμμα σημαίνει:
  - Σωστή λειτουργία
  - Εύκολη αναγνωσιμότητα
  - Συντήρηση

Μην παραβλέπετε σε καμία περίπτωση την αναγνωσιμότητα ενός προγράμματος.

## Αναγνωσιμότητα Κώδικα

```
i= 120;  
if (i>j)  
func1();  
else  
func2();
```

```
velocity = 120;  
if (velocity > max_velocity)  
    decrease_velocity( );  
else  
    increase_velocity( );
```

## Απλό πρόγραμμα σε C

```
/* A simple C program */  
#include <stdio.h>  
  
main()  
{  
    printf("hello, world");  
}
```

## 2<sup>ο</sup> απλό πρόγραμμα σε C

```
/* A simple C program */  
#include <stdio.h>  
main()  
{  
int a, b, c, sum;  
a=1;  
b=2; c=3;  
sum = a + b + c;  
printf("sum is %d",sum);  
}
```