

## Εισαγωγή στον Προγραμματισμό


# Εισαγωγή στον Προγραμματισμό

Φροντιστήριο

## Αριστείδης Ηλίας

- Κατανόηση της έννοιας του προβλήματος
- Εισαγωγή στους Αλγορίθμους

ΤΜΗΥΠ  
CEID



**Προβλήματα  
και Ανάλυση**

- Κατανόηση της έννοιας του προβλήματος
- Σαφήνεια στη διατύπωση
- Αναλυτική σκέψη στην προσέγγιση

ΤΜΗΜΑ ΗΥΠ  
CEID

- Πρόβλημα είναι μια κατάσταση η οποία χρήζει αντιμετώπισης, απαιτεί λύση, η δε λύση δεν είναι γνωστή, ούτε προφανής
- Για την αντιμετώπιση κάθε προβλήματος πρέπει προηγουμένως να έχει προηγηθεί η κατανόησή του. Αποτελεί συνάρτηση δυο παραγόντων:
  - Σωστή διατύπωση
  - Σωστή ερμηνεία από αυτόν που θα το επιλύσει
- Με τον όρο δομή προβλήματος αναφερόμαστε στα συστατικά του μέρη, στα επιμέρους τμήματα που το αποτελούν καθώς επίσης και στον τρόπο που αυτά συνδέονται μεταξύ τους.

- Ο Νίκος και η Δήμητρα είναι παντρεμένοι

- Πρώτη ερμηνεία: Ο Νίκος και η Δήμητρα είναι παντρεμένοι μεταξύ τους.
- Δεύτερη ερμηνεία: Ο Νίκος είναι παντρεμένος και η Δήμητρα είναι παντρεμένη, αλλά όχι μεταξύ τους.

ΤΜΗΜΑ  
CEID

- Η σωστή επίλυση ενός προβλήματος προϋποθέτει τον επακριβή προσδιορισμό των δεδομένων που παρέχει το πρόβλημα. Απαιτεί επίσης την λεπτομερειακή καταγραφή των ζητούμενων που αναμένονται σαν αποτελέσματα της επίλυσης του προβλήματος
- Τα στάδια αντιμετώπισης εντός προβλήματος είναι τρία
  - Κατανόηση: σωστή και πλήρης αποσαφήνιση των δεδομένων και ζητούμενων του προβλήματος
  - Ανάλυση: το αρχικό πρόβλημα διασπάται σε άλλα επί μέρους απλούστερα προβλήματα
  - Επίλυση: λύση του προβλήματος, με επίλυση των αρχικών προβλημάτων

- Οι υπολογιστές δρουν επικουρικά στην ανθρώπινη δραστηριότητα, δεν έχουν τις δυνατότητες του ανθρώπινου εγκεφάλου.
- Ο υπολογιστής μπορεί να πραγματοποιήσει τις εξής τρεις λειτουργίες
  - Πρόσθεση
  - Σύγκριση (λογικές πράξεις)
  - Μεταφορά δεδομένων

ΤΜΗΜΑ  
CEID



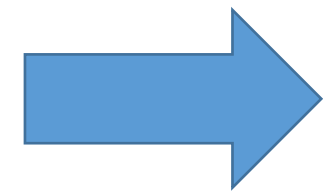


# Αλγόριθμοι και Αλγοριθμικές Δομές

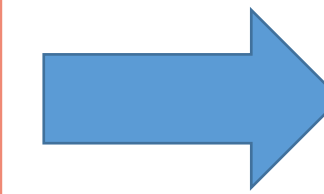
- Να διατυπώσουμε τον ορισμό για τον αλγόριθμο και να τον συσχετίσουμε με τη λύση ενός προβλήματος.
- Να περιγράψουμε τους τρόπους αναπαράστασης των αλγορίθμων
  - Περιφραστική διατύπωση
  - Λεκτική Περιγραφή βημάτων επίλυσης
  - Ψευδοκώδικας- Κωδικοποίηση (συγγραφή προγράμματος σε γλώσσα προγραμματισμού)

- Αλγόριθμος, είναι μια πεπερασμένη σειρά ενεργειών, αυστηρά καθορισμένων και εκτελέσιμων σε πεπερασμένο χρόνο, που στοχεύουν στην επίλυση ενός προβλήματος
- Κριτήρια που πρέπει να ικανοποιεί ένας αλγόριθμος είναι:
  - Είσοδος (**Καμιά, μία ή περισσότερες τιμές δεδομένων**)
  - Έξοδος (αποτελέσματα για τον χρήστη ή άλλον αλγόριθμο)
  - Περατότητα (πεπερασμένα βήματα, που εκτελούνται σε πεπερασμένο χρόνο)
  - Καθοριστικότητα (σαφώς καθορισμένες εντολές)
  - Αποτελεσματικότητα (κάθε εντολή καλώς ορισμένη είναι και εκτελέσιμη)
- Περιγραφή & αναπαράσταση αλγορίθμων
  - Ελεύθερο κείμενο (free text): αποτελεί τον πιο ανεπεξέργαστο και αδόμητο τρόπο παρουσίασης αλγορίθμου.
  - Διαγραμματικές τεχνικές: συνιστούν ένα γραφικό τρόπο παρουσίασης του αλγορίθμου (η πιο γνωστή είναι το διάγραμμα ροής- flow chart)
  - Βηματική εκτέλεση σε φυσική γλώσσα (natural language): περιγραφή κατά βήματα.
  - Κωδικοποίηση (coding): ένα πρόγραμμα που όταν εκτελεσθεί θα δώσει τα ίδια αποτελέσματα με τον αλγόριθμο (σε ψευδογλώσσα ή γλώσσα προγραμματισμού)

**Δεδομένα**  
(Δεδομένα Εισόδου)



**Επεξεργασία**  
(Αλγόριθμος)



**Αποτελέσματα**  
(Δεδομένα Εξόδου)

- **Μεταβλητές**

- Αριθμητικές: που αποθηκεύουν τιμές όπως 70, -32.5, κ.ο.κ.
- Αλφαριθμητικές: που αποθηκεύουν τιμές, όπως «Γεώργιος», «John», κ.ο.κ.
- Λογικές με τιμή αληθής ή ψευδής

- **Αποτελούν τον φορέα για τη διατήρηση των Δεδομένων και των Αποτελεσμάτων**

- Χρησιμοποιείτε όσες μεταβλητές νομίζετε και σας προκύπτει ότι χρειάζεστε!!!
- Άρα προσθέτετε μια κάθε φορά που προκύπτει η ανάγκη, χωρίς φόβο και πάθος και
- Αργότερα και με την αύξηση της εμπειρίας σας (ελπίζουμε σε αυτό το μάθημα και σε αυτό το εξάμηνο!!!) στον προγραμματισμό θα εντοπίζετε πόσες και ποιες πραγματικά χρειάζεστε

- Σημαντικά σημεία για τους αλγόριθμους που θα αναπτύσσετε:

- Οι εντολές εκτελούνται σειριακά όπως παρατίθενται στον αλγόριθμο/ πρόγραμμα (τουλάχιστον για την ώρα, επειδή αναφερόμαστε σε διαδικασιακό προγραμματισμό)
- Πρέπει να προβλέπεται κάποιας μορφής εισόδων/ εξόδων
  - Απόδοση τιμών εντός του αλγορίθμου (προγράμματος αργότερα)
  - Πληκτρολόγιο
  - Είσοδος από αρχείο
  - Είσοδος από άλλο τμήμα του προγράμματος
  - Είσοδος από άλλο πρόγραμμα
- Δεσμευμένες λέξεις (εντολές, αλγοριθμικές δομές, κ.λπ.)

- Πρόκειται για γνωστά σύμβολα πράξεων (αναλυτικά και στη C θα μιλήσουμε για όλους τους τελεστές σε μερικές εβδομάδες, εδώ εισαγάγουμε όσους χρειαζόμαστε για την αλγοριθμική σκέψη που θα δουλέψουμε σε αυτήν τη φάση)
  - Αριθμητικοί (+, -, \*, /, κ.ά. , στη C υπάρχουν μερικές διαφοροποιήσεις στον κλασικό τρόπο χρήσης των αριθμητικών- και όχι μόνο τελεστών- αλλά αυτές θα τις αντιμετωπίσουμε αργότερα)
  - Συγκριτικοί: >, <, >= , <=, = και <>
  - Λογικοί: **ΚΑΙ (AND, &&)**, **Η (OR, | |)**, **ΟΧΙ (NOT, !)** οι οποίοι συνδέουν κυρίως συνθήκες ελέγχου και το αποτέλεσμα είναι πάντα μια λογική τιμή **αληθής (true, 1, κ.λπ.)** ή **ψευδής (false, 0, κ.λπ.)**

ΤΜΗΜΑ ΗΥΠ  
CEID

## Εκφράσεις

- Συνδυασμοί μεταβλητών ή σταθερών και τελεστών
- Το αποτέλεσμα μιας έκφρασης αποδίδεται σε μια μεταβλητή με εκχώρηση (ή ανάθεση) τιμής

**π.χ. τιμή  $\leftarrow$   $\alpha + \beta$ ,**

- όπου το αποτέλεσμα του αθροίσματος των τιμών των μεταβλητών  $\alpha$  και  $\beta$  θα εκχωρηθεί (ανατεθεί) στη μεταβλητή τιμή (το βελάκι δείχνει την ενέργεια)
- Η τελική τιμή μια εκχώρησης εξαρτάται από την ιεραρχία των πράξεων και τις παρενθέσεις
- Αποδεκτές εκφράσεις

**τιμή  $\leftarrow$  5**

**τιμή  $\leftarrow$  «αρκετά»**

**τιμή  $\leftarrow$   $\alpha * \beta$**

**τιμή  $\leftarrow$  τιμή + 3**

- Μη αποδεκτές εκφράσεις

**τιμή  $\leftarrow$   $5 + \text{«}\chi\text{»}$**

**$\alpha +$  τιμή  $\leftarrow$  6**



Να διαβασθούν δύο αριθμοί, να υπολογισθεί και να εκτυπωθεί το άθροισμά τους

**Αλγόριθμος** Άθροισμα

**Διάβασε** a

**Διάβασε** b

$c \leftarrow a + b$

**Εκτύπωσε** c

**Τέλος** Άθροισμα

Ένας αθλητής πέτυχε στη σφαίρα τις επιδόσεις a, b, c. Να αναπτύξετε αλγόριθμο ο οποίος:

- α) θα διαβάζει τις τιμές των επιδόσεων a, b, c.
- β) θα υπολογίζει και θα εμφανίζει την μέση τιμή των παραπάνω τιμών.

**Αλγόριθμος** Επίδοση

**Διάβασε** a, b, c

Average  $\leftarrow (a + b + c) / 3$

**Εκτύπωσε** Average

**Τέλος** Επίδοση

- Είναι αναγκαίο (τουλάχιστον για τώρα) να σχεδιάζετε έναν αλγόριθμο στο «χαρτί», να εντοπίζετε τις αναμενόμενες τιμές των αποτελεσμάτων κι έπειτα να τον υλοποιείτε
- Υλοποιείτε σε κάποια γλώσσα
- Να επαληθεύετε τα αποτελέσματα και να τα συγκρίνετε με τα αναμενόμενα, π.χ. με πίνακες τιμών
  - πίνακας με τόσες στήλες, όσες και οι μεταβλητές που υπάρχουν και θέλουμε να παρακολουθήσουμε
  - εκτελούμε τις εντολές του αλγορίθμου μια προς μια και όταν μια μεταβλητή αλλάζει τιμή αλλάζουμε γραμμή στον πίνακα και τοποθετούμε τη νέα τιμή

## ...Σχεδιασμός- Ανάπτυξη- Επαλήθευση

$$\chi \leftarrow 2$$

$$A \leftarrow 5 * \chi$$

$$B \leftarrow 2 * \chi + 8$$

$$\Gamma \leftarrow (A + B) \bmod 2 + 10 \operatorname{div} 13$$

$$\Delta \leftarrow 5 * \Gamma - 4 \bmod 3$$

$$Y \leftarrow 2 * \chi + A / 2$$

**Εμφάνιση** A, B, Γ, Δ

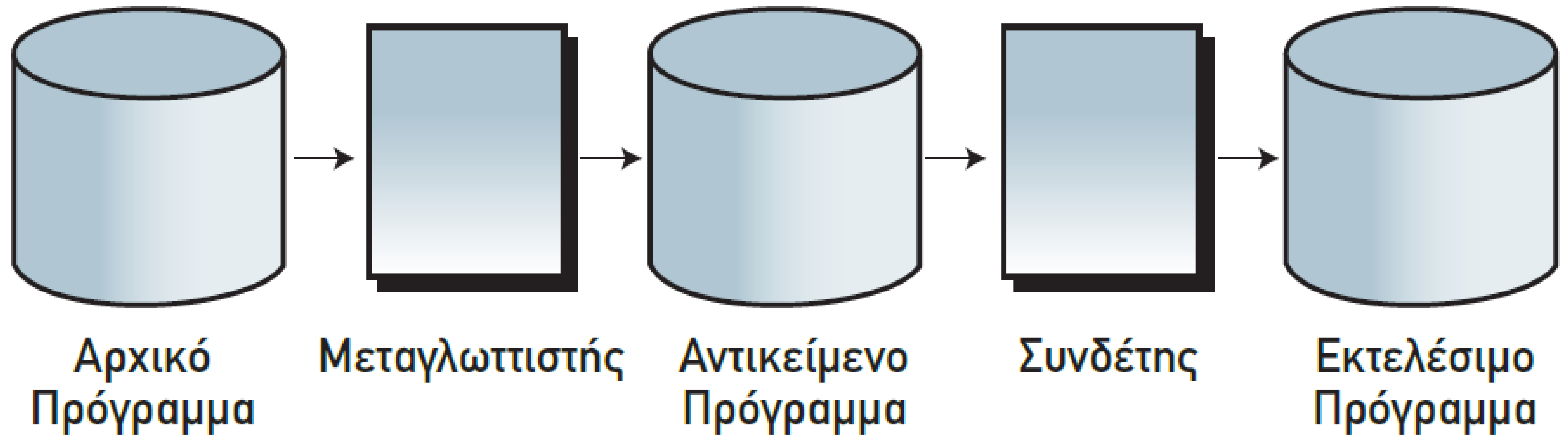
A	B	Γ	Δ	Y	χ
					2
10					2
10	12				2
10	12	0			2
10	12	0	-1		2
10	12	0	-1	9	2

- **Μεταγλωττιστής ή μεταφραστής (compiler) ονομάζεται ένα πρόγραμμα υπολογιστή που διαβάζει κώδικα γραμμένο σε μια γλώσσα προγραμματισμού (την πηγαία γλώσσα) και τον μεταφράζει σε ισοδύναμο κώδικα σε μια άλλη γλώσσα προγραμματισμού (τη γλώσσα στόχο). Το κείμενο της εισόδου ονομάζεται πηγαίος κώδικας (source code), ενώ η έξοδος του προγράμματος, η οποία συχνά έχει δυαδική μορφή, αντικειμενικός κώδικας (object code).**
- **Ο όρος «μεταγλωττιστής» χρησιμοποιείται κυρίως για προγράμματα που μεταφράζουν μια γλώσσα προγραμματισμού υψηλού επιπέδου σε μια γλώσσα χαμηλότερου επιπέδου (όπως η συμβολική γλώσσα ή η γλώσσα μηχανής).**

- Ένας μεταγλωττιστής μπορεί να περιλαμβάνει οποιαδήποτε από τις εξής λειτουργίες: λεκτική ανάλυση, προεπεξεργασία, συντακτική ανάλυση, σημασιολογική ανάλυση (μετάφραση καθοδηγούμενη από τη σύνταξη), παραγωγή κώδικα και βελτιστοποίηση κώδικα.
- Τα σφάλματα προγραμμάτων που προκύπτουν από λανθασμένη μεταγλώττιση είναι πολύ δύσκολο να εντοπιστούν και να αντιμετωπιστούν.

## ...Μεταγλωττιστής

---



## Αρχεία header που θέλουμε να συμπεριλάβουμε

τύπος-επιστρεφόμενης-τιμής όνομα-συνάρτησης  
(λίστα-παραμέτρων){

Κορμός- συνάρτησης/ εντολές

}



...Δομή ενός προγράμματος C...

```
#include<stdio.h>
```

Header αρχείο

```
int main(void){
```

Η συνάρτηση main()

```
printf("Hello World!");  
return 0;
```

Κυρίως πρόγραμμα

```
}
```

- Μέσα στην `main` γράφουμε το πρόγραμμα που θέλουμε.
- Ένα πρόγραμμα αποτελείται από διάφορες εντολές, όπως `printf("...")`, `scanf("...")`, αναθέσεις και δηλώσεις μεταβλητών,...

`int main(void)`

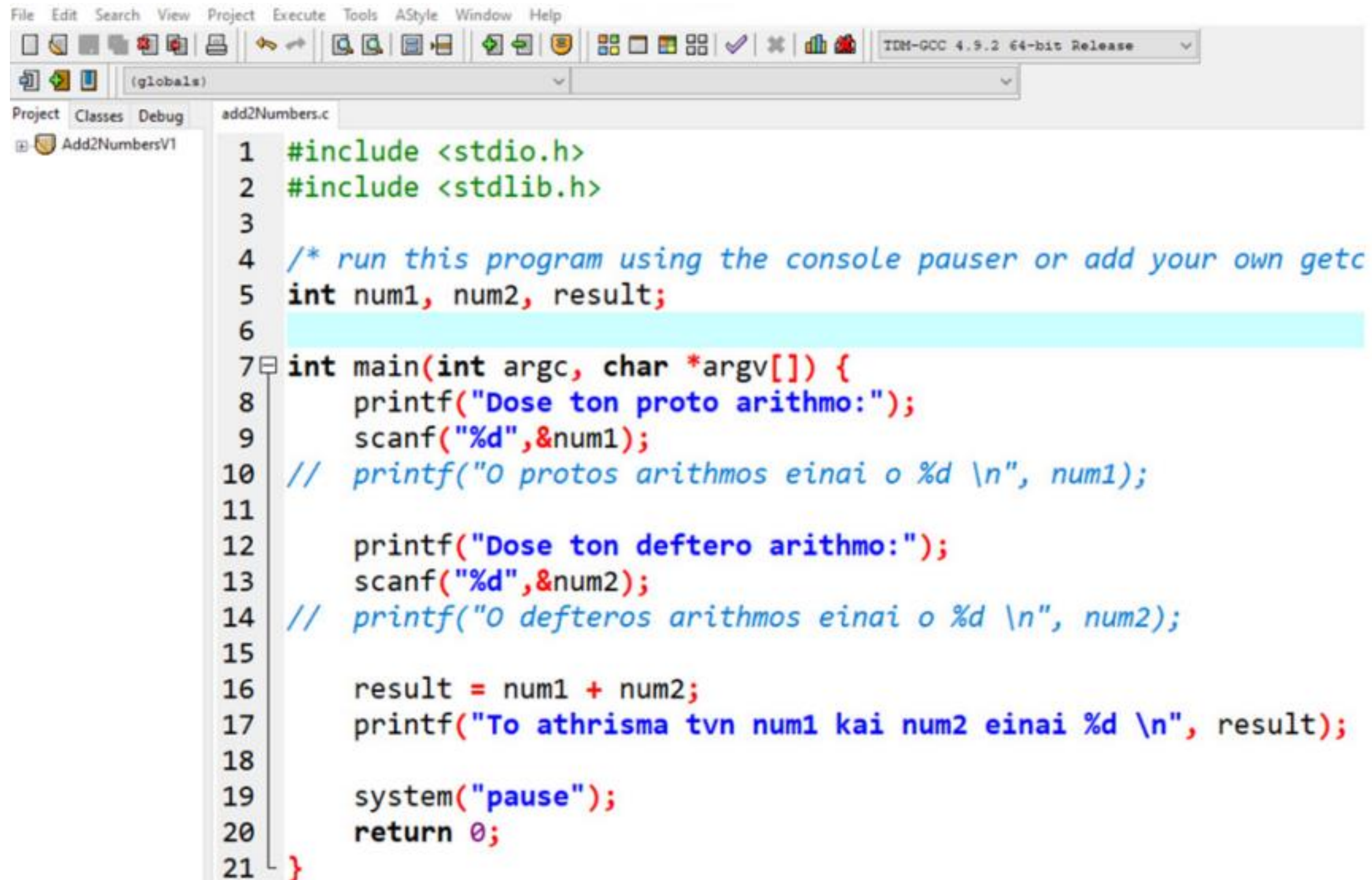
- Η συνάρτηση `main` είναι η βασική συνάρτηση του προγράμματος σας και η πρώτη που θα “τρέξει”
- Αρχικά το `int` προσδιορίζει τον τύπο μεταβλητής που θα επιστέψει η συνάρτηση (`main`)
- Μέσα στην παρένθεση της `main` προσδιορίζουμε τον τύπο των ορισμάτων.
- Στην περίπτωση μας δεν έχουμε ορίσματα γι' αυτό και έχουμε βάλει `void`.

## `#include<stdio.h>`

- Το αρχείο Header είναι ένα αρχείο με συγκεκριμένες δηλώσεις μεταβλητών και συναρτήσεων.
- Η ντιρεκτίβα `#include` λέει στον προεπεξεργαστή να διαβάσει ένα άλλο αρχείο (στην προκειμένη περίπτωση το `stdio.h`) και το συμπεριλάβει στο πρόγραμμα (`include`)


```
#include<stdio.h>  
int main(void){  
    printf("Hello World! This is our first C program!\n");  
    return 0;  
}
```

## ...Τα πρώτα μου πρόγραμμα σε C



```
1 #include <stdio.h>
2 #include <stdlib.h>
3
4 /* run this program using the console pauser or add your own getch
5 int num1, num2, result;
6
7 int main(int argc, char *argv[]) {
8     printf("Dose ton proto arithmo:");
9     scanf("%d",&num1);
10    // printf("O protos arithmos einai o %d \n", num1);
11
12    printf("Dose ton deftero arithmo:");
13    scanf("%d",&num2);
14    // printf("O defteros arithmos einai o %d \n", num2);
15
16    result = num1 + num2;
17    printf("To athrisma tvn num1 kai num2 einai %d \n", result);
18
19    system("pause");
20    return 0;
21 }
```

## Eclass και Εργαστήριο...



Επιλογές Μαθήματος

Ανακοινώσεις

**Έγγραφα** 1

Εργασίες

Ερωτηματολόγια

Ημερολόγιο

Ιστολόγιο






Μηνύματα

🏠 Χαρτοφυλάκιο / Εισαγωγή στον Προγραμματισμό - Ακ. έτος 2023-24 / Έγγραφα

### Εισαγωγή στον Προγραμματισμό - Ακ. έτος 2023-24

#### Έγγραφα

Αρχικός κατάλογος » Lab   
Υλικό Εργαστηρίων

Τύπος	Αρχείο ▾	Μέγεθος
	ComputerRoomZones.pdf	24.4 KB
	GoogleDrive.pdf	188.22 KB
	I2PLabStructure.pdf	194.79 KB
	I2PLabStudentsGroupNo1.pdf	42.3 KB
	I2PLabStudentsGroupNo2.pdf	43.38 KB

## ...Eclass και Εργαστήριο Ομάδες και Παρουσίες

Θέση	A.M.		A.M.	Θέση	
1		Ζώνη Α		14	
2				13	
3				12	
4	X			X	11
5					10
6					9
7					8

Θέση	A.M.		A.M.	Θέση	
1		Ζώνη Δ		14	
2				13	
3				12	
4	X			X	11
5					10
6					9
7					8

**I2PLabStudentsGroupNo1.pdf**

**Ομάδα Εργαστηρίου Νο 1**

Θέση	A.M.		A.M.	Θέση
1				18
2				17

Θέση	A.M.		A.M.	Θέση
1				18
2				17

**ComputerRoomZones.pdf**

A/A	Ζώνη	Θέση	Επώνυμο	Όνομα
1	Ζώνη Α	A1	ΠΑΝΙΤΣΑΣ	ΦΩΤΙΟΣ
2		A2	ΧΡΙΣΤΟΠΟΥΛΟΣ	ΓΕΩΡΓΙΟΣ
3		A3	ΚΕΠΕΝΟΣ	ΓΕΩΡΓΙΟΣ
4		A4		
5		A5	ΙΛΑΡΙΔΗΣ	ΙΩΑΝΝΗΣ
6		A6	ΣΟΦΙΑΝΟΠΟΥΛΟΣ	ΔΗΜΗΤΡΙΟΣ
7		A7	Αγγελή	Μαριάννα
8		A8	Δημητρουδιού	Παναγιώτης
9		A9	Ζουγραβού	Βασίλης
10		A10	Κουνούδη	Βασιλική
11		A11		
12		A12		
13		A13		

**Ευχαριστώ**

**Ερωτήσεις?**