

# Εισαγωγικά στοιχεία στη γλώσσα C

Μάθημα 1

# Βασικά στοιχεία της Γλώσσας

```
/*prog10.c*/  
  
#include <stdio.h>  
  
/*to prwto mou programma*/  
  
main()  
{  
    int a=10;  
    int b=5;  
    printf("Hello ");  
    printf("George\n");  
    printf("%d", a+b);  
    getchar();  
}
```

**Σύμβολα** (#, %, //, /\*, ``,>,{ κλπ)

**Λεξιλόγιο** (include, main, int, printf, getchar, κλπ)

**Συντακτικό** : κανόνες γλώσσας

Πχ κάθε πρόταση-εντολή τελειώνει σε ";" (semicolon)

**Σημασιολογία** : Τι κάνουν οι εντολές όπως συντάσσονται, πχ

Printf ("%d", a+b) : εκτελεί την πρόσθεση των αριθμών 10 και 5, μορφοποιεί το αποτέλεσμα ως ακέραιο στο δεκαδικό σύστημα και τον παρουσιάζει στην οθόνη

**Σχόλια** : ανάμεσα στα σύμβολα /\* . . . \*/

**Παρουσίαση** : Εξαρτώμενη από τον κειμενογράφο (editor)

**Χρώμα** : Ξεχωρίζουν τα δεδομένα, εντολές, τύπους κλπ

**Στοίχιση** : Για καλύτερη κατανόηση της σύνταξης

Η αλλαγή γραμμής ή τα κενά δεν επηρεάζουν το πρόγραμμα

Δεν ενδιαφέρουν τον μεταγλωττιστή (compiler) της γλώσσας

Εκτός αν κόβουν μια εντολή ή μια παράσταση πριν ολοκληρωθεί

# Βασικά στοιχεία της Γλώσσας

## (κεφαλίδες)

```
/*prog10.c*/  
  
#include <stdio.h>  
  
/*to prwto mou programma*/  
  
main()  
{  
    int a=10;  
    int b=5;  
    printf("Hello ");  
    printf("George\n");  
    printf("%d", a+b);  
    getchar();  
}
```

#: σύμβολο πριν την εντολή προ-επεξεργαστή

**include** : Εντολή προ-επεξεργαστή που ενσωματώνει στον κώδικά μας ένα αρχείο «κεφαλίδα» (**header**)

**<stdio.h>** : Το αρχείο-κεφαλίδα προς ενσωμάτωση στο κώδικα το οποίο βρίσκεται σε προκαθορισμένο φάκελο που ορίστηκε με την εγκατάσταση της γλώσσας στον υπολογιστή μας

Μπορούμε να δώσουμε το πλήρες όνομα του αρχείου-κεφαλίδας  
Πχ "c:\Dev-Cpp\include\stdio.h"

Ένα αρχείο «κεφαλίδα» περιέχει ένα σύνολο δηλώσεων «σταθερών» και συναρτήσεων τις οποίες (μετά τη δήλωσή τους) μπορούμε να χρησιμοποιούμε

Πχ η εντολή `printf` είναι μια συνάρτηση η οποία δηλώνεται στην κεφαλίδα `stdio.h`. Ενσωματώνοντας την `stdio.h` στο πρόγραμμά μας μόνο τότε μπορούμε να κάνουμε χρήση της `printf`. Ο κώδικας, ο οποίος υλοποιείτε όταν εκτελείτε η εντολή βρίσκεται σε μια βιβλιοθήκη της γλώσσας (`library`) η οποία ενσωματώνεται στον τελικό κώδικα `exe` με τον **συνδέτη** (`Linker`) της γλώσσας.

Η κεφαλίδα `stdio.h` (**standard input-output**) ενσωματώνεται σχεδόν σε κάθε πρόγραμμα γιατί περιέχει τις βασικές εντολές για την είσοδο και έξοδο δεδομένων από και προς την **κονσόλα** ή ένα **text-αρχείο**.

# Βασικά στοιχεία της Γλώσσας

(η συνάρτηση main( ) )

```
/*prog10.c*/  
  
#include <stdio.h>  
  
/*to prwto mou programma*/  
  
main()  
{  
    int a=10;  
    int b=5;  
    printf("Hello ");  
    printf("George\n");  
    printf("%d", a+b);  
    getchar();  
}
```

Συνάρτηση **main( )**

Η εκτέλεση του προγράμματος ξεκινάει από αυτόν τον κώδικα

- Κάθε πρόγραμμα (project) της C πρέπει να περιέχει μια και μοναδική συνάρτηση main( )
- Όταν κάνουμε ένα απλό πρόγραμμα τότε όλος ο κώδικας μπορεί να περιέχεται μέσα στη main( ).
- Οι εντολές μέσα στο μπλοκ { ... } εκτελούνται με τη σειρά – στο συγκεκριμένο κώδικα δεν υπάρχουν εντολές που αλλάζουν τη ροή του προγράμματος.

Η `getchar( )` αποτελεί την τελευταία εντολή του προγράμματος. Η λειτουργία της είναι να **περιμένει** να πάρει έναν χαρακτήρα απ' το πληκτρολόγιο (μετά το πάτημα του «Enter»). Ο χαρακτήρας αυτός δεν χρησιμοποιείται από το πληκτρολόγιο. Απλά εκμεταλλευόμαστε τη λειτουργία της `getchar` για να **σταματήσουμε το πρόγραμμα** πριν το τέλος του και **πριν σβήσει η κονσόλα\***.

\* Πριν την `getchar` μπορούμε για ενημέρωση του χρήστη να προσθέσουμε την εντολή `printf("press <Enter> to end the program");`

# Βασικά στοιχεία της Γλώσσας

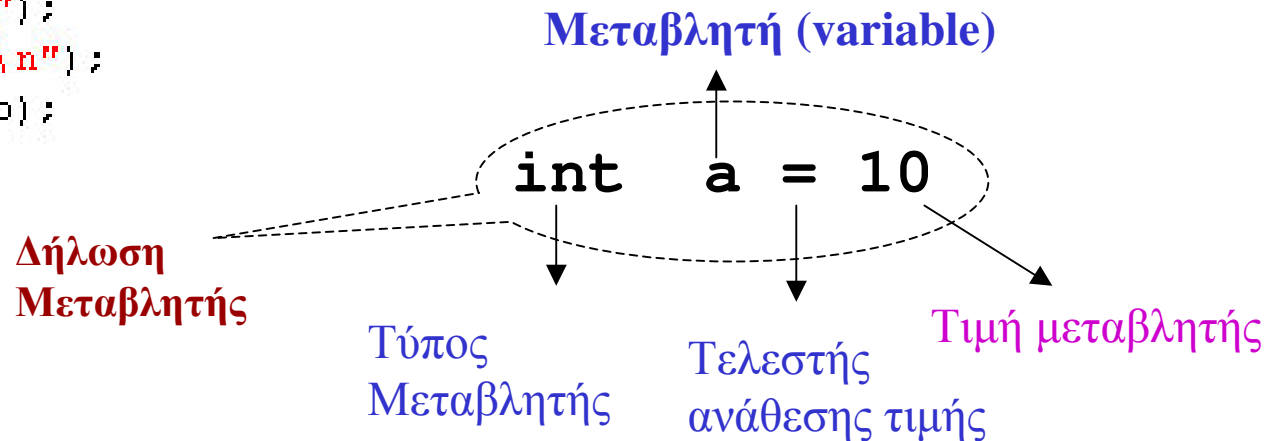
(δεδομένα & Μεταβλητές)

```
/*prog10.c*/  
  
#include <stdio.h>  
  
/*to prwto mou programma*/  
  
main()  
{  
    int a=10;  
    int b=5;  
    printf("Hello ");  
    printf("George\n");  
    printf("%d", a+b);  
    getchar();  
}
```

## 1. σταθερά δεδομένα

- Αριθμοί (10,5)
- Αλφαριθμητικά (strings)  
( "Hello", "George\n" )
- Χαρακτήρες (πχ 'A', 'c')

**2. Μεταβλητές :** σύμβολα-λέξεις του προγραμματιστή που αντιστοιχούν σε θέσεις μνήμης όπου αποθηκεύονται δεδομένα



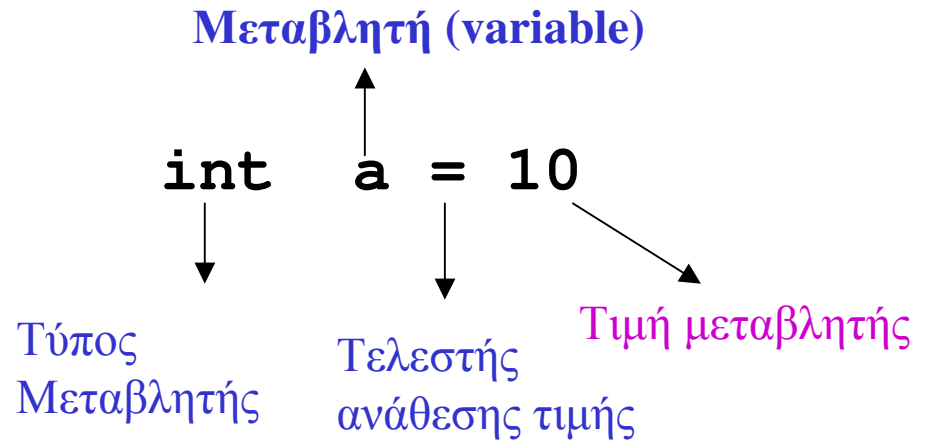
- Όλες οι μεταβλητές πρέπει να δηλώνονται
- τα δεδομένα της μεταβλητής μπορεί να αλλάζουν – το όνομα της μεταβλητής όχι!
- Η ποσότητα που υπάρχει δεξιά του = πρέπει πάντα να ανάγεται σε τιμή

# Βασικά στοιχεία της Γλώσσας

## (Δηλώσεις Μεταβλητών)

Με την Δήλωση μιας μεταβλητής έχουμε τις ακόλουθες ενέργειες του μεταγλωττιστή :

- Κράτηση θέσης στη μνήμη, με μέγεθος όσο απαιτείται από τον τύπο της μεταβλητής
- Το όνομα της μεταβλητής “a” συνδέεται με μια διεύθυνση μνήμης στην οποία θα αποθηκεύονται οι τιμές της
- Με τον τελεστή ανάθεσης τιμής γράφεται η τιμή (δεξιά του =) στον χώρο μνήμης που ορίστηκε.



- Οι μεταβλητές μιας συνάρτησης δηλώνονται στην αρχή της συνάρτησης, πχ της `main()`
- Μετά τον τύπο της μεταβλητής μπορούμε να δηλώσουμε πολλές μεταβλητές ταυτόχρονα πχ

```
int a=100, b=50, n;
```

**Σημείωση :** οι περισσότεροι μεταγλωττιστές καταλαβαίνουν τη δήλωση μιας μεταβλητής όπου δηλώνεται (αυτό ισχύει πάντα στη C++)

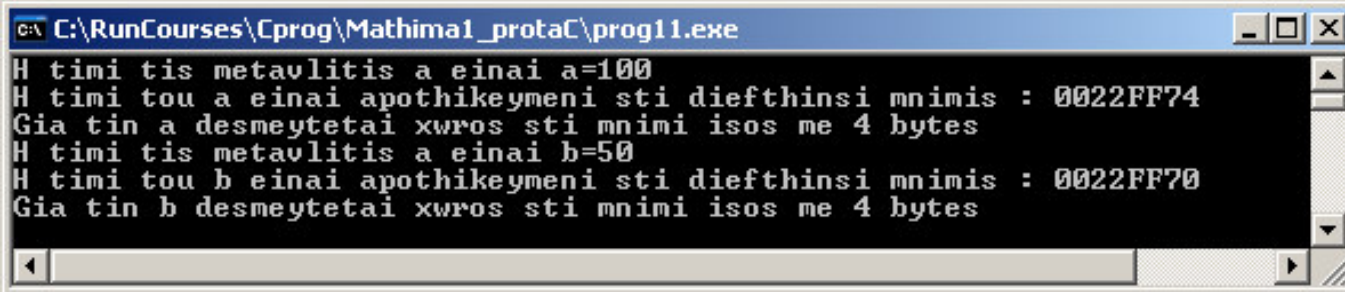
- Μια μεταβλητή μπορεί να πάρει τιμή κατά την δήλωσή της (αρχικοποίηση) ή αργότερα κάποια στιγμή μέσα στο πρόγραμμα.
- Μέχρι να πάρει μια τιμή έχει μια τυχαία τιμή (τιμή σκουπίδι!)
- Όταν ξαναπάρει μια τιμή, αντικαθιστά την προηγούμενη.

# Βασικά στοιχεία της Γλώσσας

(Μεταβλητές και Διευθύνσεις μνήμης)

```
/*prog11.c*/  
//variables and memory addresses  
#include <stdio.h>  
  
void main()  
{  
    int a=100;  
    int b=50;  
    int n;  
    printf("H timi tis metavlitis a einai a=%d \n",a);  
    printf("H timi tou a einai apothikeymeni sti diefthinsi mnimis : %p \n",&a);  
    n=sizeof(a);  
    printf("Gia tin a desmeytetai xwros sti mnimi isos me %d bytes\n",n);  
    printf("H timi tis metavlitis a einai b=%d \n",b);  
    printf("H timi tou b einai apothikeymeni sti diefthinsi mnimis : %p \n",&b);  
    n=sizeof(b);  
    printf("Gia tin b desmeytetai xwros sti mnimi isos me %d bytes\n",n);  
    getchar();  
}
```

**Αναφορά:** Αναφερόμαστε στη διεύθυνση μιας μεταβλητής με το σύμβολο **&** μπροστά από το όνομά της



```
c:\RunCourses\Cprog\Mathimal_protaC\prog11.exe  
H timi tis metavlitis a einai a=100  
H timi tou a einai apothikeymeni sti diefthinsi mnimis : 0022FF74  
Gia tin a desmeytetai xwros sti mnimi isos me 4 bytes  
H timi tis metavlitis a einai b=50  
H timi tou b einai apothikeymeni sti diefthinsi mnimis : 0022FF70  
Gia tin b desmeytetai xwros sti mnimi isos me 4 bytes
```

0022FF74

0022FF70

|00000000|00000000|00000000|00110100|00000000|00000000|00000000|00011010|

b

a

# Βασικά στοιχεία της Γλώσσας

(Εξοδος στη Κονσόλα)

```
/*prog10.c*/
```

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

```
/*to prwto mou programma*/
```

```
main()
```

```
{
```

```
    int a=10;
```

```
    int b=5;
```

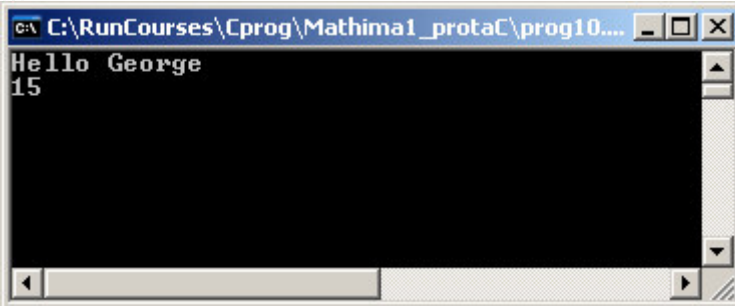
```
    printf("Hello ");
```

```
    printf("George\n");
```

```
    printf("%d", a+b);
```

```
    getchar();
```

```
}
```



```
C:\RunCourses\Cprog\Mathima1_protac\prog10...
Hello George
15
```

**printf** : συνάρτηση-εντολή που εκτυπώνει στην κονσόλα.

- Τυπώνει μόνο αλφαριθμητικές σταθερές
- Διαμορφώνει-κατασκευάζει ένα αλφαριθμητικό χρησιμοποιώντας αριθμητικά δεδομένα (χρήση του τελεστή μορφοποίησης %)

\* Οι τρεις εντολές εκτύπωσης θα μπορούσαν να εκτελεστούν με μία εντολή εκτύπωσης

```
printf("Hello George\n%d", a+b);
```



# Λάθη και Προειδοποιήσεις (errors & warnings)

**Λεκτικά λάθη :** Λάθος γραφή μιας εντολής ή μεταβλητής

**Συντακτικά λάθη:** Λάθος στη σύνταξη μιας συνάρτησης-εντολής ή σε ένα μπλοκ εντολών

**Συγκρουόμενες δηλώσεις (conflicts):** Πολλαπλές δηλώσεις μεταβλητών ή συναρτήσεων με ίδιο όνομα ή όνομα που έχει δεσμευμένο η C

**Δομικά-Λογικά λάθη :** Λάθος χρήση μεταβλητών και Συναρτήσεων

Αναγνωρίζονται από τον μεταγλωττιστή (ή συνδετή) και παίρνουμε μήνυμα λάθους  
Το πρόγραμμα ΔΕΝ εκτελείται

Κάποιες φορές αναγνωρίζονται και παίρνουμε προειδοποίηση  
Το πρόγραμμα μπορεί να εκτελεστεί

```
/*prog10e.c*/  
//errors and warnings  
#include "stdio.h"  
  
void main()  
{  
    int a=10;  
    ind b=5;  
    printf("Hello ");  
    printf("George\n");  
    printf("%d", a+b+0.5);  
    getchar();  
}
```

The screenshot shows the Dev-C++ IDE with a C program. The variable 'ind' is undeclared. The error message in the bottom pane reads: 'In function 'main': 'ind' undeclared (first use in this function) (Each undeclared identifier is reported only once)'. The error is highlighted in red in the original image.

Όλα τα λεκτικά και συντακτικά λάθη ανιχνεύονται από όλους τους μεταγλωττιστές.  
Το πόσο αξιόπιστη είναι η περιγραφή ενός λάθους εξαρτάται από τον μεταγλωττιστή.

Το πρώτο λάθος που βρέθηκε

Λίστα λαθών μετά την μεταγλώττιση

# Λάθη και Προειδοποιήσεις (errors & warnings)

```
/*prog10e.c*/
```

```
//errors and warnings
```

```
#include "stdio.h"
```

Σχόλιο με // : Λάθος για την ANSI C, αλλά δεκτό για νεότερους μεταγλωττιστές

```
void main()
```

Πιθανό δομικό λάθος - προειδοποίηση

```
{
```

```
int a=10;
```

Λεκτικό λάθος

```
ind b=5;
```

Συντακτικό λάθος

```
printf("Hello ")
```

```
print("George\n");
```

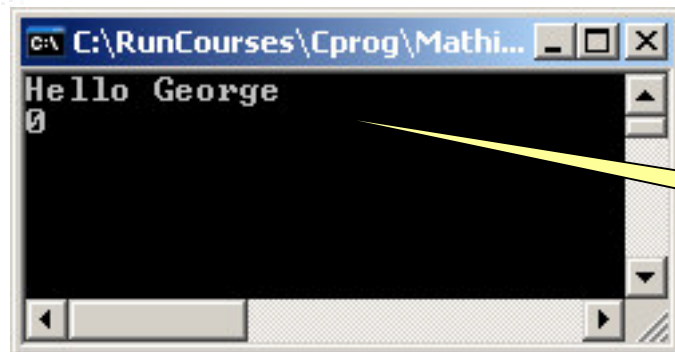
Λεκτικό λάθος – Αναγνώριση από τον συνδετή

```
printf("%d", a+b+0.5);
```

```
getchar();
```

```
}
```

Δομικό λάθος : Δεν αναγνωρίζεται



Έξοδος μετά τη διόρθωση των λεκτικών και συντακτικών λαθών

# Λεξιλόγιο της C

- Λέξεις-κλειδιά (keywords) της γλώσσας

Αποτελούν δεσμευμένες λέξεις της γλώσσας και δεν επιτρέπεται να τις δηλώσει διαφορετικά ο προγραμματιστής

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

•Οι βιβλιοθήκες της C που ενσωματώνονται στο πρόγραμμά μας περιέχουν ένα σύνολο συναρτήσεων-εντολών με προκαθορισμένα ονόματα. Αν χρησιμοποιήσουμε ένα ίδιο όνομα *ενδεχομένως* θα πάρουμε από τον συνδετή μήνυμα λάθους **συγκρουόμενης δηλώσης** (conflict)

Πχ η "printf" αποτελεί συνάρτηση βασικής βιβλιοθήκης – δεν πρέπει ο προγραμματιστής να τη δηλώσει για άλλη χρήση (πχ ως όνομα μεταβλητής)

# Λεξιλόγιο της C

## ονόματα μεταβλητών & συναρτήσεων

Τα ονόματα μεταβλητών και συναρτήσεων αποτελούνται από γράμματα (λατινικούς χαρακτήρες), αριθμούς 0..9 και τον χαρακτήρα \_

- Μπορούν να έχουν πολλούς χαρακτήρες (εκτός ειδικών περιπτώσεων ) αλλά μόνο οι πρώτοι 31 αναγνωρίζονται.
- Δεν πρέπει να περιέχουν κενά ή να ξεκινούν με αριθμό
- Χαρακτήρες πεζοί ή κεφαλαίοι αναγνωρίζονται ως διαφορετικοί

### Σωστά ονόματα

myfile  
myNumber  
Salary\_May  
x1  
x2  
x001  
YYx  
George25x  
Temperature\_of\_oven  
Thermokrasia\_Fournou

### Λάθος ονόματα

My file  
myNumber/2  
Salary-May  
1x  
x(2)  
001  
X%  
Γιώργος25  
Temperature#1  
const

Τα διάφορα σύμβολα (+,/,>,^ κλπ), εκτός της κάτω παύλας ( \_ ), χρησιμοποιούνται από τη C. Η χρήση τους μπορεί να είναι διαφορετική ανάλογα της σύνταξης.

# Άσκηση C 1.1

Να γραφεί το παρακάτω πρόγραμμα στην Dev-C++ και να διορθωθούν τα λάθη

```
#include <stdio>

int main()
{
    int a=150;
    int b=135.4;
    unsigned int c=15
    unsigned int d=-100;
    char e=150;
    printf("Program C11\n");
    printf("Data :\n");
    printf("%d\n",a);
    printf("%d",b);
    printf("%d\n",c);
    printf("%d\n",d);
    printf("%d\n",e);
    /* Παρακάτω εκτυπώνεται το άθροισμα
    print("Αποτελεσμα (athrisma) = a+b+c+d+e");
    getch();
```