

# Δομημένος Προγραμματισμός

Τμήμα Επιχειρηματικού Σχεδιασμού και  
Πληροφοριακών Συστημάτων

**[www.bpis.teicrete.gr](http://www.bpis.teicrete.gr)**

# Code::Blocks

- Γράψτε το παρακάτω πρόγραμμα μεταγλωττίστε το και τρέξτε το εκτελέσιμο

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

main{
{
    float s = 0, a, mo;
    int i, N = 6;
    for ( i=1; i<=N; i++ )
    {
        printf( " Type a number: " );
        scanf( " %f " , &a );
        s += a;
    }
    mo = s / N;
    printf( " MO = %f " , mo );
}
```

# Code::Blocks

- Πληκτρολογήστε τους παρακάτω αριθμούς
  - 5
  - 2
  - 4
  - 5
  - 8
  - 12

Ποιο είναι το αποτέλεσμα;

# Συνάρτηση main()

- Βασική ή Κύρια Συνάρτηση
- Δυνατότητα τοποθέτησης εντολών και κλήση άλλων συναρτήσεων
- Οι εντολές εντός της συνάρτησης αποτελούν το σώμα αυτής και περικλείονται με άγκιστρα { }
- ; Το σύμβολο του τερματιστή προτάσεων ή απλά τέλος εντολής ή κενή εντολή

# Δομή Προγράμματος

- Σχόλια προλόγου [προαιρετικό]
- Οδηγίες αναφοράς σε βιβλιοθήκες [προαιρετικό]
- Συναρτήσεις [προαιρετικό]
- Κύρια Συνάρτηση (main)
  - {
  - Δηλώσεις [προαιρετικό]
  - Σώμα προγράμματος
  - }

# Σχόλια προγράμματος

- // για μία γραμμή ή /\* \*/ για πολλές γραμμές
- Δεν αποτελούν τμήμα κώδικα
- Δεν λαμβάνονται υπόψη από τον μεταγλωττιστή
- Τοποθετούνται σε οποιοδήποτε σημείο του προγράμματος

**Η εισαγωγή σχολίων αποτελεί ένα εργαλείο δημιουργίας ευανάγνωστων προγραμμάτων και εύκολης κατανόησης**

# Σχόλια

```
/******  
Eyresh mesou orou 5 arithmwn  
*****/  
#include <stdlib.h>  
#include <stdio.h>  
  
main{  
{  
    float s=0, a, mo; // Orismos metablhtwn mo: mesos oros, s: a8roisma  
    int i,N=5; // Plh8os ari8mwn  
    for (i=1;i<=N;i++)  
    {  
        printf("Type a number: ");  
        scanf("%f",&a); // Eisodos ari8mwn  
        s += a; //Pros8esh ari8mou sto a8roisma  
    }  
    mo=s/N; //Ypologismos Mesou Orou  
    printf("MO = %f \n",mo); //Emfanish apotelesmatos  
}
```

# Αρχεία Κεφαλίδας

- Δηλώνονται πάντα στην αρχή του προγράμματος και πριν από την main με χρήση της οδηγίας προς τον προεπεξεργαστή

**#include <όνομα\_αρχείου\_κεφαλίδας.h>**

Οτιδήποτε ξεκινάει με # είναι μία οδηγία για τον προεπεξεργαστή

- Τα αρχεία κεφαλίδας περιλαμβάνουν συναρτήσεις συναφούς λειτουργίας
- Οι εντολές include δεν δέχονται ;

# Αρχεία Κεφαλίδας

```
/*  
Eyresh mesou orou 5 arithmwn  
*/  
#include <stdlib.h>  
#include <stdio.h>  
  
main{  
{  
    float s=0, a, mo;          // Orismos metablhtwn mo: mesos oros, s: a8roisma  
    int i,N=5;                // Plh8os ari8mwn  
    for (i=1;i<=N;i++)  
    {  
        printf("Type a number: ");  
        scanf("%f",&a);      // Eisodos ari8mwn  
        s += a;              //Pros8esh ari8mou sto a8roisma  
    }  
    mo=s/N;                   //Ypologismos Mesou Orou  
    printf("MO = %f \n",mo);  //Emfanish apotelesmatos  
}
```

## Κανόνες Δημιουργίας Ευανάγνωστων Προγραμμάτων

Η **Σωστή Στοίχιση** είναι το σημαντικότερο επίτευγμα για ευανάγνωστο κώδικα.

Δίνει έμφαση στη δομή του κώδικα σας

## Κανόνες Δημιουργίας Ευανάγνωστων Προγραμμάτων

```
/******  
Eyresh mesou orou 5 arithmwn  
*****/  
  
#include <stdlib.h>  
#include <stdio.h>  
  
main{  
{  
    float s=0, a, mo;  
    int i,N=5;  
    for (i=1;i<=N;i++)  
    {  
        printf("Type a number: ");  
        scanf("%f",&a);  
        s += a;  
    }  
    mo=s/N;  
    printf("MO = %f \n",mo);  
}
```

## Σωστή στοίχιση

## Κανόνες Δημιουργίας Ευανάγνωστων Προγραμμάτων

```
/******  
Eyresh mesou orou 5 arithmwn  
*****/  
  
#include <stdlib.h>  
#include <stdio.h>  
  
main(){  
float s=0, a, mo;  
int i,N=5;  
for (i=1;i<=N;i++)  
{printf("Type a number: ");  
scanf("%f",&a);  
s += a;}  
mo=s/N; printf("MO = %f \n",mo);}
```

## Ανύπαρκτη στοίχιση

**The International Obfuscated C Code Contest**<http://www.de.ioccc.org>

```
int i;main(){for(;i["]<i;++i){--i;}";read('-'-'-',i+++ "hell\
o, world!\n",'/'/'/')));}read(j,i,p){write(j/p+p,i---j,i/i);}
```

**The International Obfuscated C Code Contest**<http://www.de.ioccc.org>

```

#define iv 4
#define v ;(void
#define XI(xi)int xi[iv*'V'];
#define L(c,l,i)c(){d(l);m(i);}
#include <stdio.h>
int*cc,c,i,ix='\t',exit(),X='\n'*'\d';XI(VI)XI(xi)extern(*vi[]),(*
signal());char*V,cm,D['x'],M='\n',l,*gets();L(MV,V,(c+='\d',ix))m(x){v
signal(X/'l',vi[x]);}d(x)char*x;{v}write(i,x,i);}L(MC,V,M+1)xv(){c>=i?m(
c/M/M+M):(d(&M),m(cm));}L(mi,V+cm,M)L(md,V,M)MM(){c=c*M%X;V-=cm;m(ix);}
LXX(){gets(D)||vi[iv]();c=atoi(D);while(c>=X){c-=X;d("m");}V="ivxlcdm"
+iv;m(ix);}LV(){c-=c;while((i=cc[*D=getchar()])>-1)i?(c?(c<i&&l(-c-c,
"%d"),l(i,"+%d")):l(i,"%d")):(c&&l(M,""),l(*D,"%c")),c=i;c&&l(X,""),l
(-i,"%c");m(iv-!(i&l));}L(ml,V,'\f')li(){m(cm+!isatty(i=l));}ii(){m(c=cm
=++l)v)pipe(VI);cc=xi+cm++;for(V="jWYmDEnX";*V;V++)xi[*V^ ']=c,xi[*V++
]=c,c*=M,xi[*V^ ']=xi[*V]=c>>l;cc[-l]-=ix v)close(*VI);cc[M]-=M;}main(){
(*vi)();for(;v)write(VI[l],V,M);}l(xl,lx)char*lx;{v}printf(lx,xl)v
fflush(stdout);}L(xx,V+1,(c-=X/cm,ix))int(*vi[])=ii,li,LXX,LV,exit,l,
d,l,d,xv,MM,md,MC,ml,MV,xx,xx,xx,xx,MV,mi};

```

## Κανόνες Δημιουργίας Ευανάγνωστων Προγραμμάτων

- Όχι ονόματα ενός χαρακτήρα i,j,x,y  
(εκτός από μετρητές... i,j,k)
- Χρήση ονομάτων που δηλώνουν χαρακτήρα  
(εκφραστικά ονόματα μέχρι 31 χαρακτήρες)
- Χρήση μικρών γραμμάτων για μεταβλητές ή  
συνδυασμός μικρών κεφαλαίων για εκφραστικά  
ονόματα

**synolo**

**mesosOros ή mesos\_oros**

## Κανόνες Δημιουργίας Ευανάγνωστων Προγραμμάτων

```
/******  
Eyresh mesou orou 5 arithmwn  
*****/  
#include <stdlib.h>  
#include <stdio.h>  
main{  
{  
    float synolo=0, neoNoumero, mesos_oros;  
    int i,N=5;  
    for (i=1;i<=N;i++)  
    {  
        printf("Type a number: ");  
        scanf("%f",& neoNoumero);  
        synolo += neoNoumero;  
    }  
    mesos_oros =synolo/N;  
    printf("Mesos Oros = %f \n", mesos_oros);  
}
```

## Πρακτικές Καλού Προγραμματισμού

- Γράψτε τα προγράμματα σας με απλό και σαφή τρόπο – KIS ... Keep It Simple
- Διαβάστε τα εγχειρίδια που συνοδεύουν κάθε συνάρτηση της C
  - man pages του Linux ή αναζητήστε τα στο web
  - <http://linux.die.net/man/>
  - <http://man.cx/>
  - <http://man.he.net/>
  - <http://www.freebsd.org/cgi/man.cgi>
- Πειραματιστείτε με ένα άγνωστο χαρακτηριστικό της C προτού το συμπεριλάβετε σε ένα πρόγραμμα
- Χρησιμοποιείτε τις συναρτήσεις της τυπικής βιβλιοθήκης της C (είναι δοκιμασμένες...)

# Λεξιλόγιο της γλώσσας C

- Δεσμευμένες Λέξεις
  - Ονόματα συναρτήσεων
  - Ονόματα σε αρχεία κεφαλίδας EOF, INT\_MAX
  - Ονόματα τύπων
  - Ονόματα εντολών προ επεξεργαστή
  - Ονόματα που αρχίζουν με τον χαρακτήρα \_ και έχουν δεύτερο χαρακτήρα τον ίδιο ή κεφαλαίο γράμμα \_DATE, \_FILE
  - Λέξεις κλειδιά if else for goto int long struct etc
  - Τελεστές

# Λεξιλόγιο της γλώσσας C

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
( )	* / %	+ -	< <= == > >
==	!=	=	& *

# Εντολή printf

- Ανήκει στην κατηγορία εντολών εσόδου εξόδου
- Ορίζεται στην βιβλιοθήκη stdio.h
- Ονομάζεται μορφοποιούμενη διότι δίνει την δυνατότητα στον χρήστη να μορφοποιήσει την έξοδο της με ακολουθίες διαφυγής
- Χρησιμοποιείται ως προς την εκτύπωση μηνυμάτων στην οθόνη (στη ροή stdout)
- Εκτυπώνει ότι υπάρχει εντός του ορίσματος μορφοποιημένο ανάλογα με τις ακολουθίες διαφυγής
- \n : χαρακτήρας διαφυγής ακολουθούμενος από n : **Νέα Γραμμή**
- %f : εκτύπωση σε αυτή τη θέση της τιμής της μεταβλητής με μορφή αριθμού τύπου float

```
printf("MO = %f \n",mo);
```

# Προσοχή

- Στη C υπάρχει διαχωρισμός μεταξύ κεφαλαίων και μικρών (Case Sensitive)

πχ.

`printf` δεν είναι ίδια με την `Printf`

## Νέο Πρόγραμμα

- Δημιουργήστε ένα νέο project με τίτλο

**Printf1**

**Αποθήκευση στον φάκελο**

**Domimenos\11\_10\_2012**

# Ειδικοί χαρακτήρες

- Πώς λέτε στον υπολογιστή ότι θέλετε να αλλάξει γραμμή κατά την εμφάνιση ενός μηνύματος στην οθόνη;

**με τον χαρακτήρα αλλαγής γραμμής `\n`**

- Στη C υπάρχουν κι άλλοι τέτοιοι ειδικοί χαρακτήρες, για να τους γράψουμε χρησιμοποιούμε το πρόθεμα `\`.

# Άσκηση

- **Γράψτε ένα πρόγραμμα που να τυπώνει τη φράση:**

**ΑΤΕΙ ΚΡΗΤΗΣ  
ΤΜΗΜΑ ΒΠΙΣ**

- **σε δύο χωριστές γραμμές.**

# Ειδικοί χαρακτήρες

- `\n` Νέα γραμμή
- `\t` Οριζόντιο tab
- `\a` Alert. Προκαλεί ήχο στο σύστημα
- `\b` Οπισθοχώρηση (backspace)
- `\r` Επαναφορά κεφαλής (return)
- `\\` Προβάλλει το χαρακτήρα `\`
- `\"` Προβάλλει το χαρακτήρα `"`

# Νέο Πρόγραμμα

- Δημιουργήστε ένα νέο project με τίτλο

**Printf2**

**Αποθήκευση στον φάκελο**

**Domimenos\11\_10\_2012**

# Άσκηση

- Γράψτε ένα πρόγραμμα που εμφανίζει στην οθόνη το παρακάτω:

\*

\*\*\*

\*\*\*\*\*

- Γράψτε ένα πρόγραμμα που να εμφανίζει το παρακάτω:  
 $213432 + 3434$

# Αλφαριθμητικές σταθερές ή συμβολοσειρές

- Όπως είδατε για να τυπώσετε ένα μήνυμα με την printf το εσωκλείετε μέσα σε εισαγωγικά.
- Ένα τέτοιο μήνυμα ονομάζεται **αλφαριθμητική σταθερά ή συμβολοσειρά**.
- Αλφαριθμητική γιατί μπορεί να περιέχει γράμματα και αριθμούς
- Σταθερά γιατί η τιμή του, το περιεχόμενό του δηλαδή δεν αλλάζει

# Μεταβλητές

- Διαχείριση πληροφορίας σε μορφή αριθμητικών δεδομένων, γραμμάτων ή ακολουθίας γραμμάτων
- Τα δεδομένα αποθηκεύονται στην μνήμη και χρειάζονται ένα μέσο για να κληθούν από τα προγράμματα
- Η λύση στο πρόβλημα **αναφοράς** και **διαχείρισης** των δεδομένων από την μνήμη δόθηκε με την έννοια της **μεταβλητής**

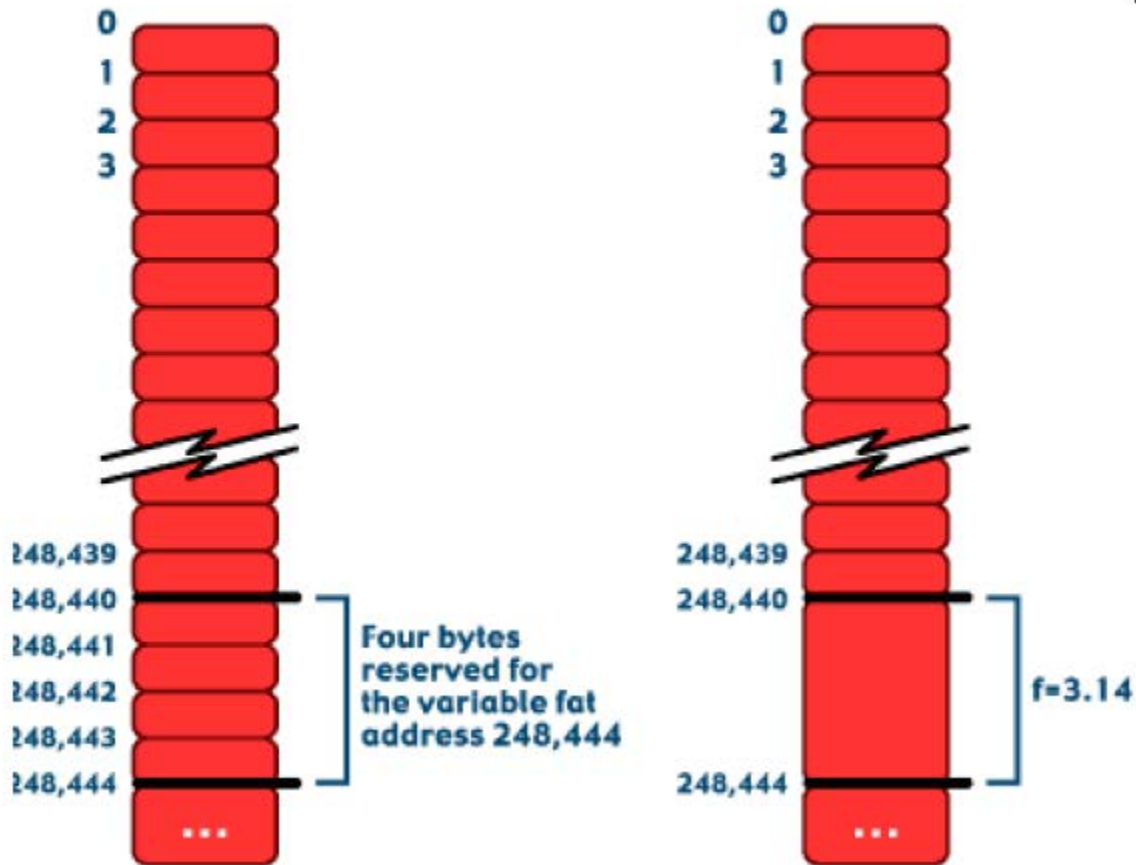
## Τι είναι μεταβλητή;

- Μεταβλητή είναι μια περιοχή στην μνήμη του υπολογιστή στην οποία μπορούμε να αποθηκεύσουμε μια τιμή και να την ανακτήσουμε. Το όνομα μιας μεταβλητής είναι άμεσα συνδεδεμένο με την διεύθυνση με την οποία είναι αποθηκευμένο το δεδομένο.
- Την μνήμη του υπολογιστή μπορούμε να την φανταστούμε σαν μια σειρά άδεια κελιά τα οποία είναι στοιχισμένα σε μια διαδοχική σειρά. Το κάθε κελί ή θέση μνήμης αριθμείτε σειριακά. Αυτοί οι αριθμοί είναι γνωστοί ως διευθύνσεις μνήμης.

# Τι είναι μεταβλητή;

float f;

f=3.14;



# Κύκλος Ζωής Μεταβλητής

## 1. Επιλογή Ονόματος

### Σχηματισμός Ονόματος από

- Γράμματα του Αλφαβήτου
- Ψηφία 0-9
- Χαρακτήρα υπογράμμισης \_

### Ξεκινάμε με

- Γράμμα ή χαρακτήρα υπογράμμισης
- Όχι δεσμευμένη λέξη
- Μέγεθος μέχρι 31 γράμματα
- Ενδεικτικό με την ιδιότητα του τύπου που αντιπροσωπεύει (Περιγραφικά ονόματα)

# Κύκλος Ζωής Μεταβλητής

## 2. Επιλογή Τύπου (Βαθμωτός ή Συναθροιστικός)

### Βαθμωτών Τύπων

- **int** κωδική λέξη για ακεραίους (integers)
- **float** κωδική λέξη για πραγματικούς (real)
- **double** κωδική λέξη για αριθμούς διπλής ακρίβειας
- **char** κωδική λέξη για χαρακτήρα (character)
- Δείκτες (pointers)
- Απαριθμητικό τύπο enum (enumerated)

### Συναθροιστικών Τύπων

- Πίνακες (Array)
- Δομές (Struct)
- Ενώσεις (Union)

# Κύκλος Ζωής Μεταβλητής

## 3. Δήλωση

- Με πρόταση ορισμού η οποία τελειώνει με ;
- Η μορφή δήλωσης είναι:  
κωδική\_λέξη\_τύπου όνομα\_μεταβλητής ;
  - int κωδική λέξη για ακεραίους (integers)
  - float κωδική λέξη για πραγματικούς(real)
  - double κωδική λέξη για αριθμούς διπλής ακρίβειας
  - char για χαρακτήρα (character)
- Δηλώνεται στην αρχή μιας συνάρτησης
- Μετά το εισαγωγικό άγκιστρο {

**Οποσδήποτε πριν απο την χρήση**

# Κύκλος Ζωής Μεταβλητής

## 4. Αρχική Τιμή

- Δίνεται αμέσως μετά την δήλωση ή πριν από την πρώτη χρήση της.
- Ισχύει μέχρι την πρώτη αλλαγή τιμής εντός προγράμματος
- Κρίνεται αναγκαία για την σωστή λειτουργία του προγράμματος
- Εξυπηρετεί στην τοποθέτηση μαζικών τιμών και δοκιμών

# Κύκλος Ζωής Μεταβλητής

## 5. Ανάθεση τιμών, Χρήση

- Πάντοτε εντός της συνάρτησης
- Όσες φορές κρίνεται αναγκαίο
- Χρήση ως προς ανάκτηση τιμής αποθηκευμένης τιμής
- Χρήση ως προς τοποθέτηση νέας τιμής

# Ο τύπος ακεραίου

- Χρησιμοποιείται για να παραστήσει ακεραίους αριθμούς
- Η δήλωση μιας μεταβλητής ακεραίου τύπου έχει τον φορμαλισμό **int όνομα\_μεταβλητής;**  
**Π.χ. int num;**
- Δηλώνεται πάντα στην αρχή μιας συνάρτησης και αμέσως μετά το {
- Το μέγεθος μια μεταβλητής ακεραίου τύπου είναι 16 bits και το σύνολο των δυνατών τιμών είναι  $2^{16} = 65536$  με εύρος τιμών 32768 έως +32768
- Προσδιοριστές
  - long int εξασφαλίζει αποθηκευτικό χώρο 32 bits
  - unsigned int δεσμεύει τιμές χωρίς πρόσημο 0-65535
  - short int τουλάχιστον 16 bits

Περιβάλλον Windows XP οι ακέραιοι έχουν 4 bytes με σύνολο δυνατών τιμών  $2^{32} = 4.294.967.296$

# Τύποι πραγματικών αριθμών

- Αναπαράσταση πραγματικών αριθμών
- Για αριθμούς που διαθέτουν κλασματικό μέρος
- Η δήλωση μιας μεταβλητής πραγματικού τύπου απλής ακρίβειας έχει τον φορμαλισμό  
**float όνομα\_μεταβλητής;**  
**π.χ. float num;**  
*Μέγεθος 32 bits*
- διπλής ακρίβειας έχει τον φορμαλισμό  
**double όνομα\_μεταβλητής;**  
**πχ. double plank;**  
*Μέγεθος 64 bits*
- Προσδιοριστής
  - long πριν από τον τύπο double για δήλωση μεταβλητής κινητής υποδιαστολής εκτεταμένης ακρίβειας  
**πχ. long double plank;**

# Έκφραση Πραγματικών Αριθμών

Αριθμός με δεκαδικά	Επιστημονική σημειογραφία	Εκθετική σημειογραφία
123456	$1.23456 \times 10^2$	1.23456e+02
0.00002	$2.0 \times 10^{-5}$	2.0e-05
50000.0	$2.0 \times 10^4$	5.0e+04

Σε μεταβλητές τύπου float από τα 32 bits τα 8 bits χρησιμοποιούνται για τον Εκθέτη, ένα για το πρόσημο και τα υπόλοιπα 23 για το κλασματικό μέρος  
Εκθέτης  $2^8 = 256$  δυνατές τιμές

## Ο τύπος χαρακτήρα

- Ο τύπος χαρακτήρα παριστάνει απλούς χαρακτήρες του αλφάβητου της γλώσσας
- Η δήλωση μιας μεταβλητής χαρακτήρα έχει τον φορμαλισμό  
**char όνομα\_μεταβλητής;**  
**π.χ. char choice;**
- 1byte για την αποθήκευση της τιμής μίας μεταβλητής χαρακτήρα
- Χαρακτήρες δια μέσο του κώδικα ASCII (American Standard Code for Information Interchange).  
Χαρακτήρας σε αντιστοίχιση με οκταψήφιο δυαδικό αριθμό

# Μηνύματα στην οθόνη

- Πως; → Με την χρήση της εντολής printf();
  - Πού ; → Εντός του σώματος του προγράμματος
  - Απλή μορφή μηνυμάτων :
    - Δέχεται ένα όρισμα το οποίο και αποστέλλει στην έξοδο. Το όρισμα περικλείεται από " μήνυμα "
- πχ. printf( " HELLO WORLD ");**
- Μορφοποιημένα μηνύματα
    - Δέχεται ένα όρισμα το οποίο και αποστέλλει στην έξοδο μορφοποιώντας το ανάλογα με τον οδηγό εντολών (μη εκτυπωμένους χαρακτήρες) που εσωκλείονται στο μήνυμα. Το μήνυμα περικλείεται από " "

```
printf("My first name is : \t Yannis \n Family Name is : \t Aikaterinidis");
```

# Τιμές μεταβλητών και εμφάνιση

- Με την χρήση της εντολής `printf()`;
- Σύνθετη Μορφή  
`printf(" μήνυμα ", μεταβλητή1, μεταβλητή2, ..., μεταβλητή_n);`
- Για κάθε μεταβλητή πρέπει να έχουμε και έναν προσδιοριστή εντός του μηνύματος.
- Το σύνολο μεταβλητών και προσδιοριστών θα πρέπει να είναι ίσο
- Οι προσδιοριστές τοποθετούνται εντός του μηνύματος, με το σύμβολο % και δίπλα έναν χαρακτήρα ανάλογα με τον τύπο της μεταβλητής
- Στους ακεραίους και πραγματικούς μπορούμε να ορίσουμε το πλήθος των ψηφίων που θα εκτυπωθούν τοποθετώντας τον επιθυμητό αριθμό ανάμεσα στο % και τον χαρακτήρα με την μορφή `a.b`  
π.χ `%3.2f`, `%4d`

## Προσδιοριστές μεταβλητών για εμφάνιση και ανάγνωση τιμών

Είδος Τιμής	Τύπος Μεταβλητής	Προσδιοριστής
Χαρακτήρας	char	%c
Σύνολο χαρακτήρων	string	%s
Ακέραιος δεκαδική	int	%d
Ακέραιος Οκταδική	int	%o
Ακέραιος Δεκαεξαδική	int	%h
Ακέραιος μεγάλου μεγέθους	long int	%ld
Ακέραιος μικρού μεγέθους	short int	%sd
Ακέραιος χωρίς πρόσημο	unsigned	%ud
Πραγματικός Απλής ακρίβειας	float	%f
Πραγματικός Διπλής Ακρίβειας	double	%f

# Νέο Πρόγραμμα

- Δημιουργήστε ένα νέο project με τίτλο

## **Variables**

# Άσκηση

- Γράψτε ένα πρόγραμμα όπου αποθηκεύετε σε μεταβλητές
  - Την ημέρα γεννήσεως σας (πχ 12)
  - Το μήνα γεννήσεως σας (πχ 7)
  - Το έτος γέννησης (πχ 1984)
  - Το ύψος σας σε μέτρα (πχ 1.85)
  - Το πρώτο γράμμα του ονόματός σας (πχ Χ)

**Προσοχή: Στη C ένας χαρακτήρας πρέπει να είναι ανάμεσα σε μονά εισαγωγικά για να θεωρηθεί χαρακτήρας, πχ 'X'**

**Ένα αλφαριθμητικό πρέπει να είναι ανάμεσα σε διπλά εισαγωγικά "xx xx"**

## Εμφανίζοντας τις τιμές των μεταβλητών

- Αν γράψετε `printf("hlikia")` τότε θα εμφανιστεί η λέξη `hlikia` και όχι η τιμή της μεταβλητής.
- Η `printf` έχει το δικό της τρόπο για να σας επιτρέψει να τυπώσετε την τιμή μιας μεταβλητής. Στο αλφαριθμητικό **“h hlikia moy einai %d”** οι δύο τελευταίοι χαρακτήρες σημαίνουν ότι σε εκείνη τη θέση θα μπει η τιμή μιας ακεραίας μεταβλητής.
- Μετά το αλφαριθμητικό βάζουμε και τη μεταβλητή την οποία θέλουμε να χρησιμοποιήσουμε.

# Παραδείγματα

- **printf(“Η ηλικια μου είναι %d”,hlikia)**  
θα τυπώσει στη θέση του %d την τιμή της μεταβλητής hlikia
- **printf(“Η ηλικια μου είναι %d και το ypsos μου %d cm”,hlikia, ypsos)**  
στο πρώτο %d θα τυπωθεί η τιμή της μεταβλητής hlikia και στο δεύτερο η τιμή της μεταβλητής ypsos

Για μεταβλητές τύπου

**float** και **double** χρησιμοποιείτε το **%f**

**int** χρησιμοποιείτε το **%d**

**char** χρησιμοποιήστε το **%c**.

# Παράδειγμα

- Ένα πρόγραμμα που εκχωρεί σε μια μεταβλητή το έτος και το τυπώνει:

```
#include<stdio.h>
void main(void)
{
    int etos;
    etos = 2003;
    printf("To etos einai %d\n", etos);
}
```

## Εμφανίζοντας τις τιμές των μεταβλητών

Επεκτείνετε το πρόγραμμα σας ώστε να εμφανίζονται οι τιμές των μεταβλητών στην οθόνη

# Η συνάρτηση scanf

- Ανήκει και αυτή στην κατηγορία εντολών εισόδου εξόδου
- Ορίζεται στην βιβλιοθήκη **stdio.h**
- Χρησιμοποιείται για την είσοδο χαρακτήρων από το πληκτρολόγιο ( από τη ροή stdin)

# Η συνάρτηση scanf

- Η εισαγωγή τιμών στις μεταβλητές μπορεί να γίνει με τη χρήση της συνάρτησης scanf και του αντίστοιχου προσδιοριστή (specifier)

**π.χ. scanf( " %c" , &ch );**

- **Επιστρέφει:**
  - τον αριθμό των στοιχείων που διάβασε επιτυχώς.
  - 0 αν δεν διαβάσει τίποτα
  - μια ειδική τιμή, την EOF (συνήθως ισούται με -1), αν βρεθεί σε κατάσταση end of file
- **Προσδιοριστές:**

• <b>%c</b> χαρακτήρας	<b>%d</b> ακέραιος
• <b>%f</b> κινητής υποδιαστολής	<b>%lf</b> διπλής ακρίβειας
• <b>%x</b> δεκαεξαδικός	<b>%s</b> συμβολοσειρά

# Εισαγωγή τιμών σε μεταβλητές

Στην εντολή `scanf` πριν από κάθε μεταβλητή τοποθετούμε τον τελεστή διεύθυνσης **&** όχι στην `printf`

```
double var;  
scanf ( "%f", &var );
```

```
scanf ( "%f", var );
```

# Άσκηση

Επεκτείνετε το πρόγραμμά σας ώστε τιμές των μεταβλητών να εισάγονται από το πληκτρολόγιο με χρήση της συνάρτησης **scanf()**

...και στη συνέχεια να τυπώνονται