

Προγραμματισμός 1

Μάθημα 1^ο.

Βασικά τμήματα ενός προγράμματος C

1. Τμήμα ορισμού απαραίτητων βιβλιοθηκών πχ #include <stdio.h> (standard input output) Αυτά είναι τα λεγόμενα αρχεία επικεφαλίδας- header files
2. τμήμα δήλωσης σταθερών , καθολικών μεταβλητών , τύπων οριζόμενων από το χρήστη και ορισμού συναρτήσεων
3. τμήμα κύριας συνάρτησης. Θα το δούμε σαν main() ή int main()

Άσκηση1

Πρόγραμμα που τυπώνει το μήνυμα στην οθόνη : «Σήμερα είναι Τετάρτη»

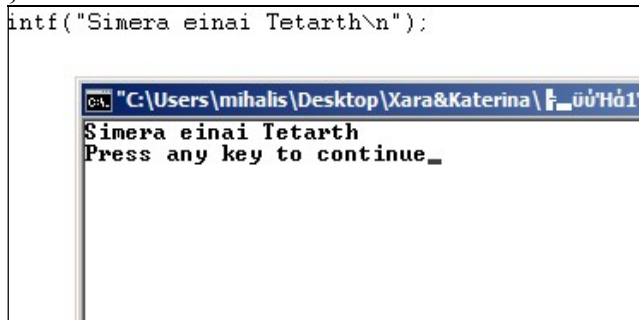
```
#include<stdio.h> //dilos i aparaithtis bibliothikis
```

```
void main(){
```

```
printf("Simera einai Tetarth\n");
```

```
}
```

```
printf("Simera einai Tetarth\n");
```



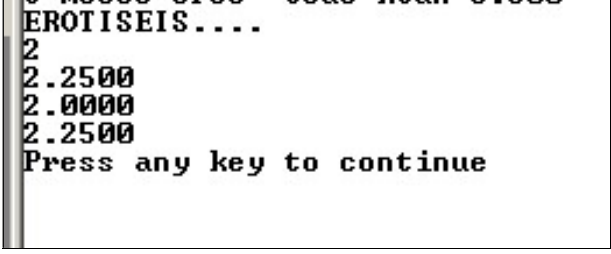
Αν σήμερα ήταν άλλη μέρα τι θα εμφανιζόταν στο πρόγραμμα;

Ερωτήσεις.

Στο παραπάνω παράδειγμα με `arithmos1=9` και `arithmos2=4` εκτελώντας τις παρακάτω εντολές :

```
printf("EROTISEIS....\n");  
printf("%d\n",arithmos1/arithmos2);  
printf("%.4f\n",arithmos1/(arithmos2*1.0));  
printf("%.4f\n",arithmos1/arithmos2*1.0);  
printf("%.4f\n",(float)arithmos1/arithmos2);
```

Γιατί πήρα τα παρακάτω αποτελέσματα;



```
EROTISEIS....  
2  
2.2500  
2.0000  
2.2500  
Press any key to continue
```

Άσκηση2

Πρόγραμμα που δέχεται δύο ακέραιους αριθμούς από το πληκτρολόγιο και εμφανίζει :

1. το άθροισμά τους
2. το πηλίκο τους
3. το ακέραιο πηλίκο τους
4. το υπόλοιπο της διαίρεσής τους
5. το μέσο όρο των δύο αριθμών

```
#include<stdio.h>
void main() {
int arithmos1,arithmos2; //dilosi topikon metavliton
float piliko;
printf("Pliktrologise ton proto arithmo..");
scanf("%d",&arithmos1);
printf("Pliktrologise ton deytero arithmo..");
scanf("%d",&arithmos2);
//ENALLAKTIKA scanf("%d %d",&arithmos1,&arithmos2);

printf("To athrisma tous htan %d \n",arithmos1+arithmos2);

piliko=arithmos1/(float)arithmos2;

printf("To piliko tous htan %4.3f\n",piliko);

printf("To AKERAIIO piliko tous htan %d\n",arithmos1/arithmos2);

printf("To ypoloipo ths diairesis tous htan %d\n",arithmos1%arithmos2);

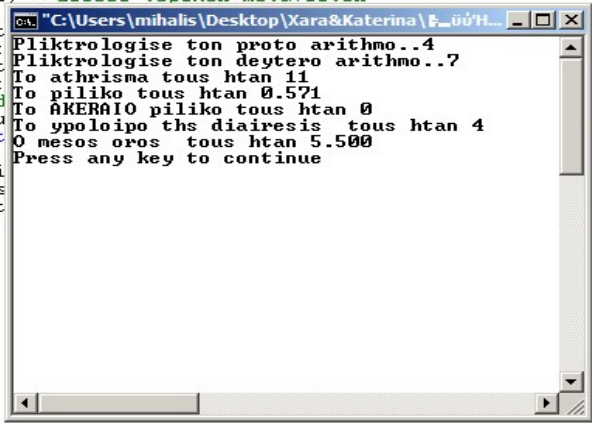
printf("O mesos oros tous htan %4.3f\n",(arithmos1+arithmos2)/2.0);
// ENALLAKTIKA ((float)(arithmos1)+arithmos2)/2
}
```

Τα πιο συχνά...

- %d για ακέραιο
- %c για χαρακτήρες
- %f για κινητής υποδιαστολής (float)
- %s για αλφαριθμητικά
- %lf για κινητής υποδιαστολής (double)

Γράψε στην οθόνη σύνολο 4 στοιχεία το ελάχιστο (ψηφία + υποδιαστολή), εκ των οποίων τα 3 θα ανήκουν στο κλασματικό-δεκαδικό μέρος. Πχ 3.278.

παράδειγμα με τους αριθμούς 4 και 7...



```

dio.h>
s1,arithmos2; //dilosi topikon metavliton
P:
ktrologise t
&arithmos1);
ktrologise t
&arithmos2);
KA scanf("%d
athrisma tou
hmos1/(float
piliko tous
AKERAIIO pili
ypoloipo ths
esos oros t
KAprintf("O

```

Προσοχή! :Στο παραπάνω παράδειγμα έκανα πράξεις με δύο ακεραίους. Η διαίρεση μεταξύ τους θεωρείται ακέραιο αποτέλεσμα και όχι αποτέλεσμα κινητής υποδιαστολής. Για να πάρω σωστό αποτέλεσμα πρέπει ο ένας από τους δύο αριθμούς που λαμβάνουν μέρος στην πράξη της διαίρεσης να γίνει με κάποιο τρόπο float. Αυτό γίνεται με :

1. πολλαπλασιάζω με 1.0 κάποιον από τους δύο ακεραίους
2. Βάζοντας τη λέξη float μπροστά από τον int που θέλω να μετατρέψω (casting).

Τελεστές μοναδιαίας αύξησης -μείωσης.

Άσκηση3

Τι θα εμφανιστεί στην οθόνη μετά την εκτέλεση του παρακάτω προγράμματος και γιατί;

```
#include<stdio.h>
void main()
{
    int x=2, y=4;
    float z;
    printf("O x einai: %d\n",x);
    printf("O y einai: %d\n",y);
    printf("O x tora einai: %d\n",x++);
    printf("O x tora einai: %d\n",++x);

    y=x++;
    printf("\nO y tha einai: %d\n",y);
    printf("O x einai: %d\n",x);

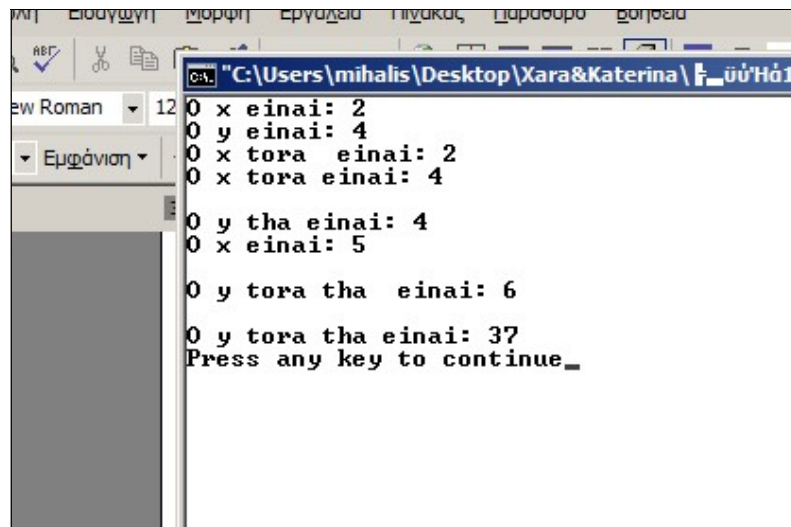
    y=++x;
    printf("\nO y tora tha einai: %d\n",y);

    y=y*++x-y++;
    printf("\nO y tora tha einai: %d\n",y);
}
```

Πρώτα χρησιμοποιεί τον x και μετά αυξάνει την τιμή του.

Πρώτα αυξάνει τον x και μετά χρησιμοποιεί την νέα αυξημένη τιμή του.

Αν έβαζα $y=y*++x-++y$ τι θα έβγαζε και γιατί;



```
C:\Users\mihalis\Desktop\Xara&Katerina\..._ού'Ηό1
O x einai: 2
O y einai: 4
O x tora einai: 2
O x tora einai: 4
O y tha einai: 4
O x einai: 5
O y tora tha einai: 6
O y tora tha einai: 37
Press any key to continue_
```

Λογικές πράξεις

Πίνακας αληθείας λογικού AND (&&)

A	B	A&&B
FALSE (0)	FALSE (0)	FALSE (0)
FALSE (0)	TRUE (1)	FALSE (0)
TRUE (1)	FALSE (0)	FALSE (0)
TRUE (1)	TRUE (1)	TRUE (1)

Πίνακας αληθείας λογικού OR (||)

A	B	A B
FALSE (0)	FALSE (0)	FALSE (0)
FALSE (0)	TRUE (1)	TRUE (1)
TRUE (1)	FALSE (0)	TRUE (1)
TRUE (1)	TRUE (1)	TRUE (1)

Το αποτέλεσμα ενός ελέγχου εκτιμάται σαν true (1) /false (0). Πχ if (a>b) then Αν το a έχει την τιμή 5 και το b την τιμή 6 τότε αυτό που υπάρχει στην παρένθεση γίνεται false δηλαδή 0. Μπορεί να έχουμε και σύνθετες εντολές στη συνθήκη.

Τί αποτέλεσμα δίνουν οι παρακάτω λογικές πράξεις;

Έστω a=5, b=7, c=true

1. (a = b) or c
2. (a <> b) and not(c)
3. not((a >b) and c)

Άσκηση 4 Τι αποτέλεσμα έχουν οι παρακάτω προτάσεις και γιατί;

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

```
void main(){
```

```
int a=1,b=0,c;
```

```
c=b&&a;
```

```
printf("c=%d\n",c);
```

```
c=a&&(++b);
```

```
printf("c=%d\n",c);
```

```
c=a&&(b++);
```

```
printf("c=%d\n",c);
```

```
printf("b=%d\n",b);
```

```
b=a=0;
```

```
c=a&+b;
```

```
printf("c=%d\n",c);
```

```
printf("b=%d\n",b);
```

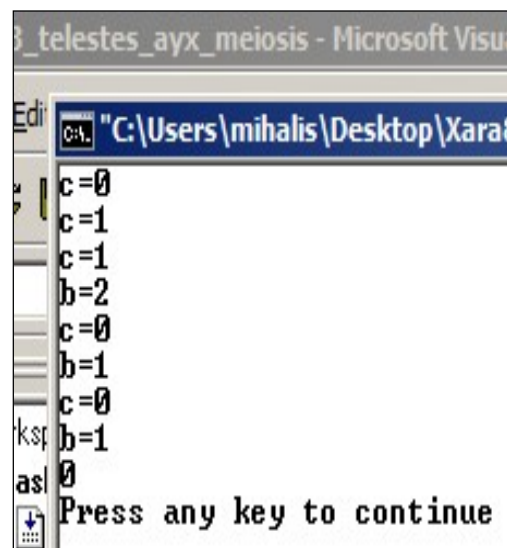
```
c=(!b&&a)&&(++b);
```

```
printf("c=%d\n",c);
```

```
printf("b=%d\n",b);
```

```
printf("%d\n",a==b && !(c||b));
```

```
}
```



```
3 _telestes_ayx_meiosis - Microsoft Visu
Edit
C:\Users\mihalis\Desktop\Xara8
c=0
c=1
c=1
b=2
c=0
b=1
c=0
b=1
0
Press any key to continue
```

Τι θα γινόταν εδώ και γιατί αν έγραφα ;
c=(!b&&a)&&(++b);

Άσκηση 5

Να γίνει πρόγραμμα που να υπολογίζει την τιμή της συνάρτησης

$$y = \frac{\sqrt{e^3} + \sin^{\frac{1}{2}}(x)}{\sqrt{\sqrt{\ln^2(x)}}} . \text{ Η τιμή του } x \text{ θα δίνεται από το χρήστη μετά από}$$

σχετικό μήνυμα που θα εμφανίζεται στην οθόνη και το αποτέλεσμα του y θα εμφανίζεται στην οθόνη.

Βοήθεια:

- $e^3 \Rightarrow \text{exp}(3)$
- $\sqrt{e^3} = \text{sqrt}(\text{exp}(3))$
- $\sin^{\frac{1}{2}}(x) \Rightarrow \text{pow}(\sin(x), 0.5)$
- $\ln^2(x) = \log(x) * \log(x) \Rightarrow \text{pow}(\log(x), 2)$
- $\sqrt{\sqrt{\ln^2(x)}} = \text{sqrt}(\text{sqrt}(\ln^2(x)))$

Προσέχω τις προτεραιότητες πράξεων και βάζω όσες παρενθέσεις χρειάζομαι. Πχ ό,τι χρειάζομαι για τον αριθμητή το βάζω σε παρένθεση για να διαιρεθεί ολόκληρος σωστά με τον παρονομαστή.

Λύση:

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

void main()
{
float x;
double y;
printf("Pliktrologiste timh gia to x\n");
scanf("%d",&x);
y=(sqrt(exp(3))+ pow(sin(x),0.5))/sqrt(sqrt(pow(log(x),2)));
printf("o apotelesma htan %.3lf\n",y);
}
```

