

## ΧΕΙΡΙΣΜΟΣ ΜΕΤΑΒΛΗΤΩΝ ΣΤΗ C

### 1. ΔΟΜΗ ΤΟΥ ΠΡΟΓΡΑΜΜΑΤΟΣ ΣΤΗ 'C'

#### 1.1 Εισαγωγή

Η Γλώσσα Προγραμματισμού 'C', όπως και κάθε άλλη δομημένη γλώσσα, απαιτεί τα προγράμματα της να έχουν μια συγκεκριμένη δομή. Η δομή αυτή βοηθά ουσιαστικά το χρήστη στην ανάπτυξη των προγραμμάτων. Μια σωστή δομή προγράμματος προϋποθέτει το σωστό σχεδιασμό της εφαρμογής για την οποία προορίζεται.

Πριν παρουσιάσουμε τη δομή του προγράμματος, αξίζει να αναφέρουμε ότι η γλώσσα 'C' ξεχωρίζει τους πεζούς από τους κεφαλαίους χαρακτήρες.

Οι δηλώσεις '#include' ονομάζονται οδηγίες προεπεξεργαστή της C και εμφανίζονται στις πρώτες γραμμές των περισσότερων προγραμμάτων. Μια οδηγία τέτοιου τύπου προκαλεί την ενεργοποίηση διαφόρων λειτουργιών σε σχέση με τον πηγαίο κώδικα των προγραμμάτων μας πριν την έναρξη της διαδικασίας μεταγλώττισης. Η C διαθέτει μια πληθώρα από τέτοιους είδους οδηγίες. Η οδηγία '#include' χρησιμοποιείται για να διαβάσει το συγκεκριμένο αρχείο (π.χ. 'stdio.h' για να χειρισθούμε τις συναρτήσεις για την είσοδο-έξοδο του υπολογιστή) και να το ενσωματώσει στο πρόγραμμα που γράφουμε σαν να ήταν αναπόσπαστο τμήμα του. Κάθε φορά που συμπεριλαμβάνουμε ένα αρχείο επικεφαλίδας, εμπλουτίζουμε το πρόγραμμα μας με μια γκάμα βιβλιοθηκών/συναρτήσεων (εντολών) που περιέχονται σε αυτό. Έτσι ανάλογα με τις συναρτήσεις που θέλουμε να χρησιμοποιήσουμε ενεργοποιούμε το αντίστοιχο αρχείο μέσω της εντολής '#include'. Τέλος θα πρέπει να αναφέρουμε ότι κάθε χρήστης έχει τη δυνατότητα να δημιουργήσει δικά του αρχεία επικεφαλίδων (βιβλιοθήκες) τα οποία και θα ενσωματώνει με μια εντολή στα προγράμματα του.

#### 1.2 Περιοχή κώδικα κυρίας συνάρτησης (κυρίως πρόγραμμα- main program)

Όλα τα προγράμματα της 'C' είναι στην ουσία ένα σύνολο από συναρτήσεις. Η βασική συνάρτηση που θα περιέχει το κυρίως πρόγραμμα ονομάζεται main(). Εκτός της συνάρτησης 'main', οι υπόλοιπες συναρτήσεις αναπτύσσονται από τον χρήστη και καλούνται είτε μέσω της 'main', είτε μέσω των υπολοίπων συναρτήσεων. Οι διάφορες δηλώσεις και ο κώδικας που θα ενσωματωθούν πρέπει υποχρεωτικά να εμπεριέχονται μεταξύ των συμβόλων "{" και "}" τα οποία και θέτουν το αρχικό και τελικό της όριο μέσα στο πρόγραμμα C. Αυτά τα προγράμματα θα χρησιμοποιούνται ευρέως στα περισσότερα σημεία ενός προγράμματος C μιας και οριοθετούν συγκεκριμένες ενότητες μέσα στο πρόγραμμα. Ένα απλό παράδειγμα προγράμματος στη 'C' είναι το ακόλουθο:

```
#include <stdio.h>

main ()
{
/* This is my first program */
printf("This is my first program\t");
printf("I am a CIED student\n");
}
```

#### 1.3 Περιοχή Δήλωσης Μεταβλητών

Σε αυτό το τμήμα δηλώνουμε μεταβλητές και σταθερές οι οποίες μπορούν να χρησιμοποιηθούν μόνο στη συνάρτηση της main. Οι μεταβλητές δηλώνονται αμέσως μετά την εισαγωγή του συμβόλου '{' στην συνάρτηση

‘main’. Οι μεταβλητές που δηλώνουμε ανήκουν στους τύπους δεδομένων που υποστηρίζει η ‘C’. Κάθε μεταβλητή περιλαμβάνει τα παρακάτω χαρακτηριστικά:

- Τύπο της μεταβλητής
- Όνομα της μεταβλητής
- Τιμή που της εκχωρείται (προαιρετικά)

### **1.3.1 Τύποι Μεταβλητών**

Οι μεταβλητές μπορούν να περιέχουν διάφορους τύπους δεδομένων. Ο παρακάτω πίνακας δίνει τους πιο σημαντικούς τύπους μεταβλητών στη ‘C’, το εύρος τιμών (ελάχιστη τιμή-μέγιστη τιμή που μπορεί να καταχωρηθεί σε μία μεταβλητή) καθώς και τον αριθμό των bytes που καταλαμβάνει ένας τύπος μεταβλητής στην μνήμη του υπολογιστή.

Τύπος Μεταβλητής	Εύρος Τιμών (Ελάχιστη-Μέγιστη Τιμή)	Απαιτούμενα Bytes
char	-128.....127	1 byte
unsigned char	0.....255	1 byte
short int	-32768.....32767	2 bytes
unsigned short int	0.....65535	2 bytes
int	-32768.....32767 -2147483648.....2147483648	2 bytes 4 bytes
unsigned int	0.....65535 0 ... 4294967295	2 bytes 4 bytes
float	$-3.4 \times 10^{38}$ ..... $3.4 \times 10^{38}$	4 bytes
long	-2147483648.....2147483648	4 bytes
double	$-1.7 \times 10^{308}$ ..... $1.7 \times 10^{308}$	8 bytes
long double	$-3.4 \times 10^{-4932}$ ..... $3.4 \times 10^{4932}$	10 bytes

### **1.3.2 Δήλωση μεταβλητών**

Οι πιο βασικοί περιορισμοί στη δήλωση των μεταβλητών είναι οι ακόλουθοι:

- Το μήκος του ονόματος της μεταβλητής πρέπει να είναι τουλάχιστον ένας χαρακτήρας και μέγιστο μήκος πεδίου 31 χαρακτήρες
- Τα ονόματα μεταβλητών πρέπει να αρχίζουν από αλφαβητικό χαρακτήρα
- Οι υπόλοιποι χαρακτήρες μπορεί να είναι συνδυασμός γραμμάτων, αριθμών ή υπογράμμισης ( \_ )
- Το όνομα της μεταβλητής δεν θα πρέπει να περιλαμβάνει ονόματα που είναι δεσμευμένα στην γλώσσα ‘C’ (π.χ. int, print, scanf, pow, sqrt, for);
- Η ‘C’ κάνει διαχωρισμό μεταξύ των μεταβλητών που έχουν δηλωθεί με μικρά γράμματα από αυτές που έχουν δηλωθεί με κεφαλαία γράμματα

Έγκυρος τρόπος μιας μεταβλητής:

```
main ()
{
int Sales31; /* Δήλωση μιας μεταβλητής που παίρνει ακέραιες τιμές*/
int sAles31; /*Δήλωση μιας μεταβλητής που παίρνει ακέραιες τιμές*/
float SAles31; /*Δήλωση μιας μεταβλητής που παίρνει πραγματικές τιμές με το πολύ 6 σημαντικά ψηφία*/
long tax_peter; /*Δήλωση μιας μεταβλητής που παίρνει μεγάλες ακέραιες τιμές*/
double inc1985; /*Δήλωση μιας μεταβλητής που παίρνει πραγματικές τιμές με το πολύ 15 σημαντικά ψηφία */
}
```

Μη έγκυρος τρόπος δήλωσης μιας μεταβλητής:

```
main ()
{
int 31_Sales;
long tax+peter;
double income*1985;
long int,printf;
}
```

### **1.3.3 Εκχώρηση τιμών σε μεταβλητές**

Η εκχώρηση τιμής σε μεταβλητή γίνεται με την πρόταση εκχώρησης, η οποία χρησιμοποιεί για το σκοπό αυτό το σύμβολο ίσον, '=' και έχει την ακόλουθη σύνταξη:

μεταβλητή=παράσταση;

Η μεταβλητή είναι το όνομα οποιασδήποτε μεταβλητής έχετε ορίσει. Η παράσταση μπορεί να είναι μια από τις παρακάτω κατηγορίες:

- Ένας αριθμός
- Μια παράσταση από μεταβλητές και/ή σταθερές
- Ένας συνδυασμός των παραπάνω κατηγοριών

```
main ()
{
int a;
long tax31;
float res_4;
a=4; /* Ekxwrisi mias statheras akeraias timis */
tax31=25.122; /* Ekxwrisi mias statheras pragmatikis timis */
res_4=25.122/0.45; /* Ekxwrisi timis apo arithmitiki pra3i */
a=5*tax31+20; /* Ekxwrisi timis apo parastasi metablitwn */
a=4/2; /* Ekxwrisi timis apo arithmitiki pra3i */
}
```

### **1.4 Σταθερές**

Κατά τη δήλωση των σταθερών, είμαστε υποχρεωμένοι να αποδώσουμε μια και μοναδική τιμή, η οποία και θα παραμείνει αναλλοίωτη στη διάρκεια εκτέλεσης του προγράμματος. Η δήλωση αυτή γίνεται με δύο τρόπους (χρησιμοποιούμε μία από τις δύο μορφές δήλωσης):

```
#define PI 3.14159
main ()
{
const pi=3.14159;
}
```

### **1.5 Η συνάρτηση printf()**

Η 'printf()' είναι η βασική συνάρτηση παρουσίασης ενός μηνύματος ή της τιμής μιας μεταβλητής στην οθόνη του υπολογιστή. Η συνάρτηση 'printf()' περιλαμβάνεται στην βιβλιοθήκη <stdio.h>. Η 'printf()' απαιτεί πάντα 'Συμβολοσειρά Ελέγχου'. Πρόκειται για μια συμβολοσειρά που καθορίζει πώς παρουσιάζεται το κείμενο και οι τιμές των μεταβλητών αν υπάρχουν.

```
#include <stdio.h>
main ()
{
printf("Line 1\n");
printf("Line 2\t");
printf("\n\tLine 3\n");
}
```

Στην οθόνη του υπολογιστή εμφανίζεται το παρακάτω μήνυμα

```
Line 1
Line 2
Line 3
```

### **1.5.1 Ορισμός Χαρακτήρων Μετατροπής**

Η συνάρτηση 'printf()' περιλαμβάνει κάποιους χαρακτήρες μετατροπής. Αυτοί οι ειδικοί χαρακτήρες δηλώνουν στην συνάρτηση 'printf()' ακριβώς πώς να ερμηνεύσει τα δεδομένα που ακολουθούν. Ο Ακόλουθος Πίνακας περιλαμβάνει μια λίστα με χαρακτήρες μετατροπής.

Χαρακτήρας Μετατροπής	Αποτέλεσμα
%d	Δεκαδικός Ακέραιος
%f	Αριθμός κινητής υποδιαστολής
%e	Αριθμοί κινητής υποδιαστολής με εκθετικό συμβολισμό
%u	Μη προσημασμένος ακέραιος
%o	Οκταδικός ακέραιος
%x	Δεκαεξαδικός ακέραιος
%%	Σύμβολο ποσοστού (%)
%c	Χαρακτήρας
%s	Συμβολοσειρά χαρακτήρων

Για την παρουσίαση μιας αριθμητικής σταθεράς ή μεταβλητής, πρέπει να συμπεριληφθεί στην συνάρτηση ο κατάλληλος χαρακτήρας μετατροπής μέσα στη συμβολοσειρά ελέγχου.

```
#include <stdio.h>

main ()
{
int age=32;
float salary=25000.356718;
char first='E';
char middle='C';
char last='E';
printf("%c\t%c\t%c\n",first,middle,last);
printf("%d%\t%f\n",age,salary);
}
```

Το αποτέλεσμα του προγράμματος θα είναι το ακόλουθο:

```
E C E
```

```
32 25000.357422
```

Εξηγήστε το αποτέλεσμα.

Για την περίπτωση που θέλετε να μορφοποιήσετε τον τρόπο παρουσίασης πραγματικών σταθερών ή μεταβλητών χρησιμοποιείται ένας αριθμός τόσο για το ακέραιο όσο και για το δεκαδικό μέρος. Για παράδειγμα αντικαταστήστε τη δεύτερη εντολή printf στο παραπάνω παράδειγμα με την εξής έκφραση:

```
printf("%d%\t%9.3f\n",age,salary);
```

Τι διαφορά παρατηρείτε;

Η γλώσσα 'C' μας δίνει τη δυνατότητα να ενεργοποιήσουμε ειδικές μορφοποιήσεις της εκτύπωσης συνδυάζοντας το σύμβολο '\ ' και τον χαρακτήρα που αντιστοιχεί στην κατάλληλη λειτουργία. Θα πρέπει να σημειώσουμε ότι για την εκτύπωση χαρακτήρων όπως τα μονά ή διπλά εισαγωγικά και το σύμβολο '\ ' τα οποία η C χρησιμοποιεί στις διαδικασίες της εκτύπωσης είναι απαραίτητο να υιοθετηθεί ένα συγκεκριμένος τρόπος σύνταξης προκειμένου να εκτυπωθούν. Ο παρακάτω πίνακας περιέχει την αντιστοιχία της ειδικής λειτουργίας με την ακολουθία χαρακτήρων μέσω της οποίας ενεργοποιείται.

Ακολουθία Χαρακτήρων	Λειτουργία
\a	Παραγωγή ήχου
\b	Backspace
\f	Αλλαγή σελίδας
\n	Αλλαγή γραμμής
\t	Tab
\\	Εκτύπωση \
\"	Εκτύπωση "

## 1.6 Η συνάρτηση scanf()

Η συνάρτηση 'scanf()' είναι ένας τρόπος εισόδου δεδομένων από το πληκτρολόγιο. Όταν ένα πρόγραμμα συναντά τη γραμμή 'scanf()', ο χρήστης μπορεί να πληκτρολογήσει τιμές, οι οποίες εκχωρούνται σε μεταβλητές. Η σύνταξη της συνάρτησης 'scanf()', περιέχει μια συμβολοσειρά ελέγχου και μία ή περισσότερες μεταβλητές και περιλαμβάνεται στην βιβλιοθήκη <stdio.h>. Η συμβολοσειρά ελέγχου περιγράφει στη C ακριβώς τον τύπο που αναμένεται να έχουν οι τιμές που θα πληκτρολογηθούν από το χρήστη.

Η συμβολοσειρά ελέγχου της 'scanf()' χρησιμοποιεί σχεδόν τους ίδιους χαρακτήρες μετατροπής με την 'printf()' με δύο διαφορές. Ποτέ δεν πρέπει να χρησιμοποιούνται οι χαρακτήρες ελέγχου εκτύπωσης όπως ο '\n' κ.λπ. σε μια συμβολοσειρά ελέγχου 'scanf()'. Η συνάρτηση 'scanf()' γνωρίζει τότε τερματίζεται η είσοδος με το πάτημα του πληκτρο 'Enter'. Επίσης η εκχώρηση της τιμής στη μεταβλητή γίνεται μέσω του τελεστή '&', ο οποίος οδηγεί στη διεύθυνση μνήμης που δεσμεύεται από τη C για να αποθηκευθεί η τιμή μιας μεταβλητής.

Επειδή δε μπορείτε να ελέγχετε τον τρόπο πληκτρολόγησης του χρήστη, η ακρίβεια εισαγωγής δεδομένων δεν είναι πάντα εγγυημένη. Έτσι, μπορείτε να ζητήσετε από το χρήστη να γράψει έναν ακέραιο και μετά μια τιμή κινητής υποδιαστολής, αλλά ο χρήστης να γράψει κάτι άλλο. Αν συμβεί αυτό, δε μπορείτε να κάνετε πολλά. Τα αποτελέσματα θα είναι λανθασμένα, αλλά το πρόγραμμα 'C' δεν διαθέτει μέθοδο ελέγχου της ακρίβειας του

χρήστη πριν την εκτέλεση του. Σε επόμενες εργαστηριακές ασκήσεις θα φροντίζεται μέσω κατάλληλης διαδικασίας ελέγχου να αποφεύγονται οι ανεπιθύμητες εισαγωγές δεδομένων.

Παράδειγμα χρήσης εντολής scanf:

```
#include <stdio.h>
main ()
{
float totalSale;
float tax;
scanf("%f",&totalSale);
tax=totalSale*0.07;
printf("The sales tax for %.2f\t is %.2f\n",totalSale,tax);
}
```

Το αποτέλεσμα αν ο χρήστης πληκτρολογήσει την τιμή 10.0 είναι:

```
10.0
The sales tax for 10.00  is 0.70
```