

Προγραμματισμός σε C

Σημειώσεις

02 Μεταβλητές, σταθερές, τελεστές

Περίληψη

02 Μεταβλητές, σταθερές, τελεστές

1	Identifiers (αναγνωριστικά)	2
2	Δήλωση μεταβλητών	2
3	Λέξεις κλειδιά και σχόλια	3
4	Σταθερές	4
5	Είσοδος-έξοδος	4
6	Αριθμητικοί Τελεστές - τελεστής αντικατάστασης	6
7	Άσκηση - Γινόμενο & πηλίκο	7
8	Άσκηση - Ευρώ σε δολάρια	8
9	Άσκηση - Περίοδος εκκρεμούς	8
10	Άσκηση- Bits και Bytes	8
11	Άσκηση - DVD Club	8

Notes

1 Identifiers (αναγνωριστικά)

Η γλώσσα προγραμματισμού C χρησιμοποιεί τα αναγνωριστικά ως ονόματα για μεταβλητές, συναρτήσεις κτλ. Ένα αναγνωριστικό πρέπει να ξεκινάει με **γράμμα** ή το **σύμβολο υπογράμμισης** `_` για να είναι έγκυρο. Επίσης, η γλώσσα C είναι μία γλώσσα `case sensitive` που σημαίνει ότι το όνομα `foititis` είναι διαφορετικό από το όνομα `Foititis`. Τέλος, ένα όνομα δεν μπορεί να αποτελείται από κενά, όταν θέλουμε το όνομα να έχει 2 ή περισσότερες λέξεις υπάρχουν οι εξής λύσεις:

`foititis_TEI` ή `foititisTEI`

Notes

2 Δήλωση μεταβλητών

Μία μεταβλητή θα πρέπει να δηλωθεί πριν χρησιμοποιηθεί. Υπάρχουν 4 βασικοί τύποι δεδομένων που μπορούμε να χρησιμοποιήσουμε και φαίνονται στον παρακάτω πίνακα:

Τύπος	Ελάχιστο πλάτος σε bit	Εύρος
char	8	0 έως 255
int	16	-32768 έως 32767
float	32	3.4E -38 έως 3.4E + 38
double	64	1.7E - 308 έως 1.7E + 308

Εκτός από τους βασικούς τύπους δεδομένων υπάρχουν και οι τροποποιητές αυτών, ας τους δούμε συνολικά:

Τύπος	Ελάχιστο πλάτος σε bit	Εύρος
char	8	-128 έως 127
unsigned char	8	0 έως 255
signed char	8	-128 έως 127
int	16	-32768 έως 32767
unsigned int	16	0 έως 65535
signed int	16	-32768 έως 32767
short int	16	-32768 έως 32767
unsigned short int	16	0 έως 65535
signed short int	16	-32768 έως 32767
long int	32	-2147483648 έως 2147483649
signed long int	32	-2147483648 έως 2147483649
unsigned long int	32	0 έως 4294967296
float	32	$3.4^E - 38$ έως $3.4E + 38$
double	64	$1.7^E - 308$ έως $1.7E + 308$
long double	64	$3.4^E - 4932$ έως $1.1E + 4932$

Υπάρχουν **2 είδη μεταβλητών**: οι **τοπικές** μεταβλητές και οι **καθολικές** μεταβλητές. Η δήλωση των τοπικών μεταβλητών πρέπει να γίνεται **στην αρχή κάθε συνάρτησης** που θα χρησιμοποιηθούν και έχουν εμβέλεια μόνο μέσα σε αυτή. Αντίθετα, οι καθολικές μεταβλητές δηλώνονται πριν την βασική συνάρτηση `main()` και έχουν εμβέλεια όλο το πρόγραμμα (χρησιμοποιούνται σε ειδικές περιπτώσεις).

Η δήλωση μιας μεταβλητής γίνεται ως εξής:

τύποςΔεδομένων όνομαΜεταβλητής;

ή

τύποςΔεδομένων όνομαΜεταβλητής = αρχικήΤιμή;

εάν θέλουμε να της αποδώσουμε και κάποια αρχική τιμή. Έτσι, εάν θέλαμε να δηλώσουμε μία μεταβλητή με ακέραια τιμή 10, η δήλωση θα ήταν κάπως έτσι:

int a = 10;

Φυσικά, μπορούμε να ομαδοποιήσουμε πολλές μεταβλητές του ίδιου τύπου κατά την δήλωσή τους:

int a, b, c = 9;

3 Λέξεις κλειδιά και σχόλια

Κατά την δήλωση μιας μεταβλητής, σταθεράς ή συνάρτησης δεν επιτρέπεται να χρησιμοποιήσουμε λέξεις- κλειδιά της γλώσσας δηλαδή λέξεις που έχουν ειδική σημασία για την C. Οι λέξεις αυτές φαίνονται στον ακόλουθο πίνακα:

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

Εάν θέλουμε να βάλουμε σχόλια σε κώδικα υπάρχουν 2 τρόποι:

σχόλιο γραμμής: με χρήση του `//`

σχόλιο παραγράφου: με χρήση του `/* */`

4 Σταθερές

Εάν θέλουμε να δηλώσουμε κάποια σταθερή τιμή στο πρόγραμμά μας η οποία δεν θα αλλάξει κατά την διάρκεια εκτέλεσής του, τότε θα το κάνουμε με χρήση σταθεράς. Οι σταθερές δηλώνονται μετά τις οδηγίες προς τον προεπεξεργαστή (βιβλιοθήκες) και η σύνταξη έχει ως εξής:

#define όνομαΣταθεράς τιμήΣταθεράς

Αν για παράδειγμα θέλαμε να δηλώσουμε μία σταθερά με το όνομα PI και τιμή 3.141925 θα γράφαμε το εξής

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

```
#define PI 3.141925
```

...Είσοδος-έξοδος

Εάν θέλουμε να εκτύπώσουμε στην οθόνη (έξοδος), τότε χρησιμοποιούμε την έτοιμη συνάρτηση printf() που ο κώδικάς της είναι γραμμένος στην βιβλιοθήκη <stdio.h>. Η γενική της μορφή είναι η ακόλουθη:

```
printf("αλφαριθμητικό", μεταβλητές);
```

Το αλφαριθμητικό μπορεί να περιέχει σταθερούς χαρακτήρες που θα εμφανιστούν ως έχουν στην οθόνη, χαρακτήρες διαφυγής (με το \) ή χαρακτήρες φορμαρίσματος στην θέση των οποίων εμφανίζονται τιμές μεταβλητών. Οι κύριοι χαρακτήρες φορμαρίσματος φαίνονται παρακάτω:

%c για μεταβλητή τύπου `char`

%d για μεταβλητή τύπου `integer`

%f για μεταβλητή τύπου `float`

%s για μεταβλητή τύπου πίνακα χαρακτήρων (`string`)

Στους χαρακτήρες φόρμας `%d` και `%f` μπορούμε να εισάγουμε και καθοριστικά ελαχίστου πλάτους πεδίου:

```
int i=10;
```

```
printf("to i exei timi %05d\n", i);
```

που θα εμφανίσει **00010** ή

```
float i=10.123;
```

```
printf("to i exei timi %05.1f\n", i);
```

που θα εμφανίσει **010.1**

Εάν θέλουμε να εισάγουμε τιμή από το πληκτρολόγιο, τότε χρησιμοποιούμε την συνάρτηση `scanf()` η σύνταξη της οποίας φαίνεται παρακάτω:

`scanf("χαρακτηρεςΦόρμας", διεύθυνσηΜεταβλητών);`

η οποία ανήκει επίσης στην βιβλιοθήκη `stdio.h` .

Αν για παράδειγμα θέλαμε να εισάγει ο χρήστης 2 ακεραίους από το πληκτρολόγιο, θα γράφαμε τα εξής:

```
printf("Δωσε τα α και β:");  
scanf("%d%d", &a, &b);
```

Παρατηρήστε ότι οι χαρακτήρες φόρμας πρέπει να συμφωνούν με τον τύπο των `a` και `b` και ότι στις μεταβλητές είναι απαραίτητο να έχουμε μπροστά τον χαρακτήρα της διεύθυνσης μεταβλητής **&**.

6 Αριθμητικοί Τελεστές - τελεστής αντικατάστασης

Στην C υπάρχουν οι παρακάτω **αριθμητικοί τελεστές** με τους οποίους μπορούμε να εκτελέσουμε μαθηματικές πράξεις:

Τελεστής	Πράξη
-	αφαίρεση, αρνητικό πρόσημο
+	Πρόσθεση
*	πολλαπλασιασμός
/	Διαίρεση
%	ακέραιο υπόλοιπο διαίρεσης
--	μείωση κατά ένα
++	αύξηση κατά ένα

Επίσης, υπάρχει και ο **τελεστής αντικατάστασης** που μπορούμε να τον χρησιμοποιήσουμε όταν η μεταβλητή του αριστερού μέλους επαναλαμβάνεται ως πρώτη στην παράσταση του δεξιού μέλους:

$x = x + 2;$ θα μπορούσε να γραφεί ως **$x += 2;$**

$x = x * (y+2)$ θα μπορούσε να γραφεί ως **$x *= y+2;$**

7 Άσκηση - Γινόμενο & πηλίκο

Να γραφεί πρόγραμμα στο οποίο

- θα ζητούνται να εισαχθούν 2 ακέραιοι
- θα υπολογίζει και θα εκτυπώνει το γινόμενο και το πηλίκο τους

Λύση

```
#include <stdio.h>

int main()
{

    int a,b;
    int ginomeno;
    float piliko;

    printf("Δωσε α και β:");
    scanf("%d %d",&a,&b);

    ginomeno = a * b;
    piliko = (float)a / b;

    printf("Γινόμενο: %d\n", ginomeno);
    printf("Πηλίκο: %.2f\n", piliko);

    return 0;
}
```

8 Άσκηση - Ευρώ σε δολλάρια

Να γραφεί πρόγραμμα το οποίο θα διαβάζει ένα ποσό χρημάτων σε ευρώ και θα υπολογίζει και θα εκτυπώνει το ισοδύναμο ποσό σε δολλάρια σύμφωνα με τον τύπο **1 ευρώ = 1,12 δολλάρια**.

9 Άσκηση - Περίοδος εκκρεμούς

Να γραφεί πρόγραμμα το οποίο

- θα ζητάει από το πληκτρολόγιο το μήκος έστω L ενός εκκρεμούς
- θα υπολογίζει και θα εμφανίζει στην οθόνη την περίοδο T του εκκρεμούς

Ισχύει $T = 2\pi \sqrt{L/g}$

$$g = 9,81 \text{ m/sec}^2$$

10 Άσκηση- Bits και Bytes

Να γραφεί πρόγραμμα το οποίο

- θα διαβάσει τη χωρητικότητα ενός αρχείου δεδομένων σε KBytes
- θα τη μετατρέψει σε bytes και σε bits
- θα εμφανίζει τον αριθμό των KBytes, bytes και bits

11 Άσκηση - DVD Club

Να γραφεί πρόγραμμα το οποίο

- θα διαβάσει τον αριθμό των DVD που ενοικίασε ο πελάτης
- θα υπολογίζει και θα εκτυπώνει το κόστος ενοικίασης εάν είναι γνωστό ότι υπάρχουν 2 τρόποι ενοικίασης:
 - 1: σαν μέλος, εγγραφή μέλους 30 ευρώ, τιμή ενοικίασης DVD 1,5 ευρώ
 - 2: χωρίς εγγραφή, τιμή ενοικίασης DVD 2 ευρώ

12

Να γραφεί πρόγραμμα το οποίο τυπώνει το ονοματεπώνυμό, τη διεύθυνση και ταχυδρομικό κώδικα της περιοχής σας σε ξεχωριστές γραμμές.

13

Να γραφεί πρόγραμμα το οποίο διαβάζει από το πληκτρολόγιο τη μέση θερμοκρασία για κάθε μήνα ενός έτους. Στη συνέχεια να τυπώνει στην οθόνη τη μέση θερμοκρασία του έτους, του κάθε εξαμήνου και του κάθε τριμήνου.