

Εργαστήριο Προγραμματισμού Υπολογιστών I (C)

Εργαστήριο 1

Σκοπός:

- Γνωριμία με τη γλώσσα προγραμματισμού C, το περιβάλλον της και τον compiler της C που χρησιμοποιείται στο εργαστήριο.
- Βασική δομή ενός προγράμματος.
- Δημιουργία, μεταγλώττιση και εκτέλεση απλών προγραμμάτων.
- Μεταβλητές και εκχώρηση τιμών σε αυτές.

Μπείτε στο λογαριασμό σας και φτιάξτε ένα φάκελο μέσα στον οποίο θα γράψετε όλα τα προγράμματα για το εργαστήριο. Μπορείτε να τον ονομάσετε ProgI ή κάποιο άλλο αντιπροσωπευτικό όνομα. Μέσα σε αυτό το φάκελο φτιάξτε έναν γι'αυτό το πρώτο εργαστήριο. Μπορείτε να τον ονομάσετε lab1 ή κάποιο άλλο αντιπροσωπευτικό όνομα.

Ανοίξτε τη Dev C++ (ή Microsoft Visual C++) που είναι εγκατεστημένη στους υπολογιστές του εργαστηρίου, ή όποιο άλλο προγραμματιστικό περιβάλλον είναι εγκατεστημένο για τη C (περαιτέρω πληροφορίες θα δοθούν στο εκάστοτε εργαστήριο), δημιουργήστε ένα νέο project (empty workspace) και στη συνέχεια δημιουργήστε ένα νέο αρχείο για να γράψετε το πρόγραμμά σας (new → C++ source file).

Θα δώσετε ένα όνομα στο αρχείο που θα δημιουργήσετε για την 1^η εργαστηριακή άσκηση το οποίο θα έχει την επέκταση .cpp (π.χ. askisi1.cpp).

Header files και η ντιρεκτίβα του προεπεξεργαστή (preprocessor)

```
#include
```

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

Δομή ενός προγράμματος σε C

- Εντολές προεπεξεργαστή
 - Δηλώσεις συναρτήσεων
 - Δηλώσεις μεταβλητών
 - Κυρίως πρόγραμμα**
 - Ορισμοί συναρτήσεων

Κυρίως Πρόγραμμα

```
int main ( )
```

```
{
```

- δηλώσεις μεταβλητών

- Είσοδος δεδομένων
- Επεξεργασία δεδομένων
- Έξοδος αποτελεσμάτων

- Συναρτήσεις εισόδου/εξόδου
π.χ. `printf("αλφαριθμητικό");`

Χρήση σχολίων και αναγνωσιμότητα προγράμματος

```
/* Εισαγωγή σχολίων στο πρόγραμμα σας είναι απαραίτητη για τον Καλό */  
/* Προγραμματισμό! */
```

Παράδειγμα 1

Χρήση της συνάρτησης `printf()`.

Πληκτρολογήστε στο **Devcpp** το πιο κάτω πρόγραμμα και ονομάστε το **p1.c**.

```
#include <stdio.h>  
int main(){  
    printf("Hello world!!");  
    return 0;  
}
```

- Κάντε `compile`, και εφ' όσον δεν υπάρχουν συντακτικά λάθη, τρέξτε το.
- Τροποποιήστε το το `p1.c` έτσι ώστε να τυπώνει στην οθόνη το όνομά σας.
- Προσθέστε τον χαρακτήρα `'\n'` πριν κλείσουν τα διπλά εισαγωγικά και παρατηρήστε τι αλλάζει στην εκτέλεση του προγράμματος.

Δραστηριότητα 1

Γράψτε ένα πρόγραμμα το οποίο κάνει το εξής:

- Εκτυπώνει στην οθόνη το Όνομα και το Επώνυμό σας, τον ΑΜ σας και το Τμήμα στο οποίο ανήκετε, σε διαφορετικές γραμμές το καθένα με την χρήση του `'\n'`.

ΔΗΛΩΣΗ ΜΕΤΑΒΛΗΤΩΝ & ΕΚΧΩΡΗΣΗ ΤΙΜΩΝ

Κάθε μεταβλητή ή σταθερά έχει ένα τύπο

τύπος όνομα-μεταβλητής;

Τύπος	Δεσμευμένη λέξη
χαρακτήρας	<code>char</code>
προσημασμένος ακέραιος	<code>int</code>
αριθμός κινητής υποδιαστολής	<code>float</code>
αριθμός κινητής υποδιαστολής διπλής ακρίβειας	<code>double</code>
απουσία τιμής	<code>void</code>

- Ο τύπος καθορίζει το μέγεθος του «κουτιού» που θα δεσμευτεί στη μνήμη

- char: 1 byte
- int: 4 bytes
- float: 4 bytes
- double: 8 bytes
- Προσδιοριστές
 - short int: 2 bytes
 - long int: 8 bytes
 - long double: .. bytes
- Προσδιοριστές: *signed, unsigned*
- unsigned int, unsigned char
 - 32 bits, 8 bits
 - $0 \dots 2^{32}-1$, $0 \dots 2^8-1$
- signed int, signed char
 - 31 bits, 7 bits
 - $-2^{31} \dots 2^{31}-1$, $-2^7 \dots 2^7-1$
- Πολλαπλή δήλωση μεταβλητών
Π.χ. float x, y, z;
- Ονοματολογία μεταβλητών όμοια με συναρτήσεις (0-9 όχι στην αρχή, γράμματα αλφαβήτου και κάτω παύλα)
- Η C είναι case-sensitive
Π.χ. int count, COUNT;
- Εκχώρηση τιμών (αποτελεί εντολή): όνομα-μεταβλητής = τιμή;
Παράδειγματα: counter = 100;

fl = 100.1;

fl = 100.0;
- Κάτι ακόμα για τη συνάρτηση εξόδου printf()


```
printf("This prints the number %d", 99);

printf("This displays %d, too", 99);
```

Παράδειγμα 2

```
#include <stdio.h>

int main(void)
```

```

{
    char ch;
    int i;
    float f;
    double d;
    ch='A';
    i=10;
    f=100.1;
    d=150.2;

    printf("ch is %c, ", ch);
    printf("i is %d, ", i);
    printf("f is %f, ", f);
    printf("d is %f.", d);
    return 0;
}

```

Παράδειγμα 2

Πράξεις με ακέραιους αριθμούς.

Πληκτρολογήστε στο **Devcpp** το πιο κάτω πρόγραμμα και ονομάστε το **p2.c**.

```

/*finds the sum of 2 numbers */
#include <stdio.h>
int main(){
    int a,b,sum;

    a=10;
    b=15;
    sum=a+b;
    printf("sum= %d\n", sum);
    return 0;
}

```

- Προσθέστε κατάλληλες εντολές στο p2.c έτσι ώστε να υπολογίζει τη διαφορά και το γινόμενο 2 αριθμών. Hint: ορίστε 2 ακόμη μεταβλητές π.χ. **diff** για τη διαφορά (difference) και **prod** για το γινόμενο (product) και γράψτε τις αντίστοιχες εντολές με την γραμμή: `sum=a+b;`
example: `diff=a-b;`

Προβλήματα

1. Γράψτε ένα πρόγραμμα με χρήση μεταβλητών, το οποίο να εμφανίζει τους παρακάτω αριθμούς στην μορφή που δίνονται:

1	1
12	12
123	123
1234	1234
12345	12345

2. Γράψτε ένα πρόγραμμα το οποίο να υπολογίζει και να εκτυπώνει το συνημίτονο, το ημίτονο και την εφαπτομένη των 60° . Για εκτελεστεί σωστά το πρόγραμμα αυτό απαιτείται να συμπεριλάβετε τη βιβλιοθήκη `math.h` και να μετατρέψετε τις μοίρες της γωνίας σε ακτίνια, χρησιμοποιώντας τον τύπο $\text{ακτίνια} = \text{μοίρες} * \pi / 180$