

---

# Llenguatges de Programació

## Curs 2006 – 07 / Bloc #1 - Pràctica #2

---

### Notes importants:

- Per fer els exercicis d'aquesta pràctica necessiteu haver llegit el **tema 3 i el principi del tema 4 (fins les cadenes de caràcters) del manual de C**. Fixeu-vos bé en tots els **exemples** que hi ha explicats en el manual.
- Si no us dóna temps d'acabar tots els exercicis durant la sessió de pràctiques és **molt important** que els **acabeu després** pel vostre compte.
- El darrer exercici s'ha d'entregar abans de la següent sessió de laboratori via el Campus Virtual.

**Exercici #1>** Quan us porteu malament a classe, és probable que el/la professor/ra us faci quedar al final per copiar 500 vegades a la pissarra alguna cosa així com ara: "Estaré atent a classe". El domini de la sentència iterativa `for` us pot ajudar bastant en aquest cas.

a) Feu una petita aplicació en què l'usuari entri un nombre enter positiu (i gran!!!!!!), que representarà el nombre de vegades que heu de copiar una certa frase, i usant el `for`, imprimeu-la les vegades que faci falta.

b) Canvieu la sentència `for` per l'estructura `do/while` i feu que funcioni exactament igual.

c) Feu el mateix però fent servir l'estructura `while`.

**Exercici #2>** Escriviu un algorisme que llegeixi 10 nombres enters per teclat i calculi i escrigui la suma dels nombres parells, quants nombres parells hi ha i la mitja dels nombres senars. Feu tres versions, utilitzant les tres estructures iteratives (*for*, *while*, *do while*).

**Exercici #3>** Escriure un programa que calculi el mínim, el màxim i la mitja d'una sèrie de nombres introduïts per l'usuari. La sèrie acaba quan l'usuari introdueix un nombre negatiu.

**Exercici #4>** Feu un petit programa en el que creeu tres vectors enters (`v1`, `v2` i `v3`) de 10 posicions cadascun i, utilitzant bucles, empleneu-los de la següent manera:

1. A cada posició de `v1` hi col·loqueu el seu índex.
2. A cada posició hi de `v2` hi col·loqueu el doble del seu índex.
3. A cada posició de `v3` hi col·loqueu el doble d'un enter entrat per l'usuari.

**Exercici #5>** Una de les coses a tenir molt en compte quan treballem amb arrays és controlar que accedim sempre a índexs vàlids de l'array. Executeu el següent programa i comproveu què és el que passa. Executeu-lo pas a pas utilitzant el debugger per veure què passa amb el valor de les variables. Fixeu-vos on està el problema i corregiu-lo.

```
#include <stdio.h>

void main()
{
    int i;
    int v[5];

    for (i=0; i<10; i++)
    {
        v[i]=0;
    }
}
```

**Exercici #6>** Feu un programa que mostri per pantalla el següent menú:

1. Introduir nombre
2. Jugar
3. Sortir

En les declaracions de les variables haureu d'afegir una variable entera, N, inicialitzada a -1. Si l'usuari tria l'opció 1 li heu de demanar que introdueixi un nombre de l'0 al 10, i el guardeu a la variable N. Si l'usuari tria jugar s'haurà de generar un nombre aleatori del 0 al 10 i haureu de mostrar un missatge per pantalla si el jugador guanya (és a dir, el nombre coincideix amb el valor N) o bé si perd.

L'usuari ha de poder introduir nombre i jugar tants cops com vulgui fins que decideixi sortir del programa (amb l'opció 3).



**NOTA:** *Aquest programa el podreu aprofitar per pràctiques posteriors.*

**Indicacions:** *Per generar el nombre aleatori feu servir la següent funció:*

```
// Funció per generar un número aleatori entre 0 i max
int NumAleatori(int max)
{
    int num;

    num=((int)ceil(((double)rand()/RAND_MAX)*(max+1))%(max+1));

    return num;
}
```

*Cal que afegiu la següent línia de codi en el vostre programa principal, després de declarar les variables.*

```
srand( (unsigned)time( NULL ) );
```

**Exercici #7>** Feu un petit programa que empleni 2 vectors numèrics de longitud 5 a través de pantalla. Creeu un tercer vector que sigui la suma dels dos anteriors (element a element) i mostreu el resultat.

**Exercici #8**> Les funcions **getchar**, **getche** i **getch**, són funcions que serveixen per a entrar caràcters, per fer-vos una idea de la diferència que hi ha entre elles, executeu el següent programa per veure la diferència de les funcions anteriors:

```
#include "stdio.h"
#include "conio.h"
void main(void)
{
    char c1,c2,c3;
    printf("El següent caràcter l'entraràs usant getchar: ");
    c1=getchar();
    printf("El següent caràcter l'entraràs usant getche: ");
    c2=getche();
    printf("\nEl següent caràcter l'entraràs usant getch: ");
    c3=getch();
    printf("\nEls caracters que has entrat són: \n%c\n%c\n%c\n",c1,c2,c3);
}
```



### **Exercici que heu d'entregar abans de la sessió vinent a través del Campus Virtual**

---

Els programes no tan sols permeten triar una opció entre vàries, sinó que a més, han de ser capaços de donar l'oportunitat a l'usuari de fer-ne d'altres o repetir-les.

Piqueu el codi necessari per tal que el vostre programa us demani dos vectors de nombres reals amb 3 components,  $v$  i  $w$ , i us deixi triar entre les següents opcions:

- a. Sumar-los.
- b. Restar-los
- c. Calcular el mòdul de cadascun dels vectors.
- d. Escriure un vector  $u$  de la següent forma  $u(i) = \max(v(i),w(i))$  per totes les components.
- e. Sortir.

**NOTA**>No s'ha de sortir del programa fins que l'usuari triï l'opció Sortir. Per tant un cop hagi fet una operació ha de tenir l'oportunitat de fer altres operacions amb els mateixos vectors o bé introduir vectors nous i fer operacions amb els nous vectors.