

## **PROBLEMES Programació C**

### **CURS 2022-23**

1. Escriure un programa que demani tres nombres de tipus int. El programa ha de mostrar per pantalla la suma dels tres nombres introduïts.
2. Escriure un programa que mostri per pantalla el màxim de dos nombres de tipus float introduïts per l'usuari.
3. Fer un programa que demani un número de tipus int i imprimeixi per la pantalla el seu factorial.
4. Escriure un programa que, donat el nombre d'operacions N que volem calcular (passada com a argument al programa), mostri la sèrie de totes les sumes possibles des de 1 fins a N (per cada i de 1 a  $i \leq N$ ), en el format indicat. Per exemple, si  $N = 3$ , la sortida del programa hauria de ser:

```
1=1
1+2=3
1+2+3=6
```

5. Feu un programa que: a) Inicialitzi un vector amb una seqüència de 20 valors numèrics. b) Mostri per pantalla el valor màxim, així com la posició que ocupa en el vector. En el cas d'estar repetit el valor màxim es mostrarà el màxim índex dels valors màxims.
6. Fer un programa que realitzi els següents passos:
  - i. Inicialitza una matriu 4x4 amb valors numèrics tipus int.
  - ii. Mostra per pantalla el contingut de la matriu.
  - iii. Calcula i mostra per pantalla la suma de tots els elements
  - iv. Mostri per pantalla el valor mínim, així com la posició que ocupa en el vector. En el cas d'estar repetit el valor mínim -> es mostrarà la seva última posició.
7. Fer un programa que treballi amb matrius NxN (es declara una matriu global d'una mida fixa NxN) i funcions:
  - i. Init () - inicialitza la matriu amb nombres de tipus int.
  - ii. QuantsCops () - donat un enter retorna el nombre de vegades que apareix el nombre introduït en la matriu.
  - iii. Duplicats () - modifica la matriu multiplicant tots els seus elements per 2.
  - iv. GeneraVector () - inicialitza un vector de dimensió N amb els valors de la diagonal de la matriu.
8. Fer un programa que escrigui en la pantalla: 1.) quantes vegades apareix la lletra "a" i 2.) quantes lletres en "majúscula" n'hi han en una frase que s'introdueix per l'entrada.
9. Fer una funció que trobi el 2n element més gran d'un vector (A) d'N elements, retornant la seva posició.
10. Fer una funció que calculi l'histograma, és a dir, el nombre d'aparicions de cada lletra minúscula (a-z) en un string i ho emmagatzemi en un array de 26 posicions (la posició 0 correspon al número d'ocurrències de la lletra a, la 1 a les de la b i així successivament fins la lletra z a la que li correspon la posició 25). El codi ascii de la lletra 'a' és el 92, el de la 'b'

el 93 i així successivament fins la 'z' que té codi 122. La funció rep l'string sobre el que calcularà l'histograma, l'array on l'emmagatzemarà (no cal inicialitzar-lo) i la longitud de l'string. La funció retornarà el número de caràcters total que ha comptabilitzat.

11. Fer una funció que inverteix un array de 100 enters, és a dir, el primer element passa a ser l'últim, el segon el penúltim, etc.
12. Escriu un programa que, per a un nombre N superior a 2 (passat com argument al programa), mostri a la pantalla els primers N nombres de la seqüència de Fibonacci. Recordeu que la sèrie de Fibonacci es defineix de la següent manera:

$$F(0)=0$$

$$F(1)=1$$

$$F(2)=1$$

$$F(n)=F(n-2)+F(n-1)$$

El format de sortida hauria de ser el següent (suposant, per exemple,  $N=7$ ):

Seqüència de Fibonacci per a  $N=7$ :

$$F(0)=0$$

$$F(1)=1$$

$$F(2)=1$$

$$F(3)=2$$

$$F(4)=3$$

$$F(5)=5$$

$$F(6)=8$$

$$F(7)=13$$

13. Desenvolpeu una rutina en llenguatge C que ordeni una llista de números enters de forma ascendent fent servir el següent algorisme:
  1. Dividim la llista en una sub-llista ordenada (inicialment buida) i una desordenada (inicialment tota la llista). No cal crear dues llistes, la divisió es fa in-situ (n'hi ha prou en saber en quina posició comença la llista desordenada, inicialment en la posició 0).
  2. Mentre hi hagin elements a la llista desordenada fer
    - a. Buscar l'element més petit de la llista desordenada
    - b. Intercanviar-lo amb el primer element de la llista desordenada
    - c. Incrementar l'índex que indica on comença la llista desordenada

La capçalera de la rutina que heu de programar és:

```
// llista[]: array d'enters inicialment desordenats
// len: longitud (nombre d'elements) de la llista
void ordena(int llista[], int len)
```

**14.** Quina de les afirmacions següents és una declaració correcta d'un punter en C?

- A) ptr c;
- B) int p;
- C) int \*p;
- D) prt &c;

**15.** Hem declarat la variable int p; quina de les instruccions següents retorna l'adreça de p?

- A) &p
- B) \*pàg
- C) \*\* pàg
- D) Cap de les anteriors

**16.** Quin és el resultat de l'execució del codi següent:

```
int Data=0;
int *ptrData;
ptrData=&Data;
printf ("%p, %d\n", &Data, Data);
printf ("%p, %d\n", ptrData, *ptrData);
```

**17.** Quin serà el valor de (\*q) després d'executar aquest codi?

```
int x = 10;
int *p = &x;
int *q = p;
```

- A) Cap, el codi és incorrecte
- B) 10
- C) L'adreça de p
- D) Cap de les anteriors

**18.** Quina de les següents instruccions ens permet assignar memòria per a 10 elements de tipus float?

- A) (float)malloc (10)
- B) (float \*)malloc (10)
- C) (float \*)malloc (10\*sizeof(float))
- D) (float \*)malloc (10\*sizeof(float\*))

,

- 19.** Doneu el valor resultant de cada assignació. Suposem que les línies s'executen seqüencialment:

```
int main()
{
    int blocks[3] = {1,2,3};
    int *ptr = &blocks[0];
    int temp;

    temp = blocks[0];
    temp = *(blocks + 2);
    temp = *(ptr + 1);
    temp = *ptr;

    ptr = blocks + 1;
    temp = *ptr;
    temp = *(ptr + 1);

    ptr = blocks;
    return 0;
}
```

- 20.** Quines línies d'aquest fragment de codi contenen un error?

```
int *ip;
ip = (int *)malloc (sizeof(int)*10);
ip[6] = 42;
ip[10] = 99;
*(ip+4) = 4;
ip+4 = 4;
```