

**Programació 1 – Grau en Enginyeria Informàtica**  
Escola Politècnica Superior  
Recuperació - 31 de Gener de 2019

---

**DNI:**

**Cognoms, Nom:**

---

**1. (3 punts)** Donat el següent programa en ANSI C++:

```
#include <stdio.h>

#define N 5

void A(char m[N][N + 1], int n);
int B(int i, int &plus);

int main( ){
    char m[N][N + 1] = {"abcde", "fghij", "klmno", "pqrst", "uvwxy"};
    int i = 0, plus = 1;

    while (i < N * N) {
        A(m, i);
        i = B(i, plus);
    }
}

void A(char m[N][N + 1], int n) {
    int i, j;

    i = n / N;
    j = n % N;
    printf("Element[%i][%i]=%c\n", i, j, m[i][j]);
}

int B(int i, int &plus) {
    i = i + plus;
    plus++;
    return i;
}
```

Indiqueu què **mostra exactament per pantalla** la seva execució. Explica breument què fa el programa (no vol dir explicar què fa cada línia de codi).

**2. (3 punts)** Implementeu en el llenguatge de programació ANSI C/C++ una funció que anomenareu `filsen_igual_colpar` que determini si els valors de les files senars sumen igual que els valors de les columnes parells. Només cal implementar aquesta funció i si en necessiteu alguna altra al aplicar disseny descendent, **no s'han d'implementar les funcions d'entrada de dades a la matriu ni la funció main**.

La funció tindrà el prototipus següent:

```
bool filsen_igual_colpar(int m[N][N]);
```

on  $N$  és una constant definida entre 4 i 1000.

La funció `filsen_igual_colpar` rebrà com a entrada la matriu  $m$  de  $N \times N$  valors enters i retornarà `true`, si la suma dels valors de les files senars és iguals a la suma dels valors de les columnes parell, i `false` en cas contrari. Per aquest exercici considerarem que la primera dimensió de la matriu són les files i la segona les columnes. **Feu un disseny que no depengui de la mida de la matriu quadrada**, és a dir, que no depengui de si  $N$  val 4 o 1000.

Per exemple, si la matriu d'entrada  $m$ , per una  $N$  de 4, és la següent:

	0	1	2	3
0	19	-8	2	0
1	24	2	-3	-7
2	-17	8	3	10
3	5	8	-64	4

la funció `filsen_igual_colpar` retornarà `true` ja que els valors de la fila 1 (24, 2, -3, -7) i fila 3 (5, 8, -64, 4), que són les files senars, sumen igual que els valors de la columna 0 (19, 24, -17, 5) i columna 2 (2, -3, 3, -64), que són les columnes parells. Les files senars sumen  $24 + 2 + (-3) + (-7) + 5 + 8 + (-64) + 4 = -31$  i les columnes parells sumen  $19 + 24 + (-17) + 5 + 2 + (-3) + 3 + -64 = -31$ .

**3. (4 punts)** Donada una llista de paraules emmagatzemada en una matriu de caràcters, es demana que dissenyeu i implementeu en el llenguatge ANSI C++ i utilitzant la tècnica de disseny descendent, una acció que mostri les paraules de la llista que tenen el seu primer caràcter repetit en la resta de la paraula, quantes vegades es repeteix el caràcter i quin és aquest primer caràcter. Les paraules que no tinguin cap repetició del primer caràcter no s'han de mostrar. Tingueu en compte que:

- La llista pot contenir paraules buides (com a l'exemple que es mostra a continuació).
- Si la matriu no està plena de paraules, indicarem el final de la llista amb la paraula "zzz" (com a l'exemple).
- El primer caràcter no es compta com a repetició, mireu l'exemple per més claredat.
- **No** heu de fer la funció `main` ni cap altra funció per l'entrada de les paraules a la llista, només l'acció que se us demana i si necessiteu alguna altra funció/acció al aplicar el disseny descendent.
- Feu el problema en general per qualsevol llista de paraules, no només per resoldre el cas de l'exemple.

L'acció tindrà el prototipus següent:

```
#define T 100 // Màxim de paraules
#define M 20  // Màxim de caràcters per paraula
void mostrar_repeticions_1e_caracter(char lp[T][M + 1]);
```

Per exemple, si inicialitzem la llista de la següent forma:

```
char lp[T][M + 1] = {"gat", "acabalar", "tortuga", "gos", "toro", "cocodril", "", "xai", "zzz"};
```

i seguidament la passem com a paràmetre a la funció `mostrar_repeticions_1e_caracter`, la sortida ha de ser:

```
acabalar té 3 repeticions de la primera lletra a.
tortuga té 1 repeticions de la primera lletra t.
cocodril té 1 repeticions de la primera lletra c.
```

Per resoldre el problema podeu utilitzar les funcions de la llibreria estàndard `string.h` que cregueu convenientes.