

# Respostes Prova C

| NOM                   | NIU     |
|-----------------------|---------|
| Gorka Sagristà Novell | 1745950 |
| Biel Puga Jodas       | 1745043 |
| Arnau López Lozano    | 1744876 |
| Iker López Marcos     | 1746098 |

## Llegenda:

- `<text>` → Codi escrit per a resoldre la tasca.
- `<text>` → Comentari, explicació del que fa el codi.
- `<text>` → Secció de la prova.
- `<text>` → `#include ...`
- `<text>` → Indica a on segueix el codi de resposta.
- `<text>` → Resposta directa a la pregunta.
- `<text>` → Enllaç a secció del document.

## Índex

### 1. Declaració de variables i "printf"

- a. 1
- b. 2
- c. 3

### 2. Funcions i paràmetres

- a. 1
- b. 2

### 3. Arrays

- a. 1
- b. 2 + 3
- c. 4

## Declaració de variables i " printf" | [Torna a dalt](#)

---

1)

```
#include <stdio.h>
int main(){
printf("Hello World!\n");
}
```

---

2)

```
#include <stdio.h>
int main(){
int x = 5;
printf("Hello World! %d\n", x);
}
```

---

3)

```
#include <stdio.h>
int main(){
    int x = 5;
    float y = 2.5;
    char c = 'z';
    printf( "%.1f ", y);
    printf("Hello World! %d\n", x);
    printf( "%c\n", c);
}
```

---

1)

```
#include <stdio.h>
#include <stdlib.h>
int Max( int a, int b ){
    if ( a < b ) return b;
    return a;
}
int main(){
    int x, y, z;
    // Generarem una seqüència sempre igual de nombres pseudo-aleatoris.
    srand(1234);
    x = rand()%10000;
    y = rand()%10000;
    z = Max( x, y );
    // printf(??);
}
```

---

2)

```
#include <stdio.h>
#include <stdlib.h>
// Funció que retornarà el màxim entre dos números "a" i "b". Són dues
// variables locals que després s'utilitzaran amb un altre nom per a
// comparar-les.
int Max( int a, int b ){
    if ( a < b ) return b; // Si "a" és menor que "b", retorna "b", ja
// que és major que "a", i viceversa en el cas contrari.
    return a;
}

int SumDiv(int num) {
    int sum = 0;
    for (int i = 1; i <= num; i++) { // Fem una iteració des d'1 fins que
// arribi a "num".
        if (num % i == 0) { // Si "i" és un divisor de "num": ...
            sum += i; // ... el sumem a sum.
        }
    }
    return sum; // Retornem la suma total dels divisors.
}
```

(segueix a baix)

Així és com quedaria el main:

```
int main(){
    int x, y, z; // Utilitzarem "x" i "y" per a emmagatzemar els
nombres generats i "z" per a guardar el màxim entre els dos.
    int sumdX, sumdY; // Declarem aquestes dues variables per a poder
utilitzar-les.
    srand(1234); // Generarem una seqüència sempre igual de nombres
pseudo-aleatoris.
    x = rand()%10000; // Nombre aleatori per a "x" entre 0 i 9999
    y = rand()%10000; // Nombre aleatori per a "y" entre 0 i 9999
    z = Max( x, y ); // Nombre més gran entre "x" i "y". Aquí, "x" es
converteix en "a" i "y" en "b" per al propòsit de la funció.

    sumdX = SumDiv(x); // Emmagatzemarem la suma dels divisors de "x"
calculada.
    sumdY = SumDiv(y); // Emmagatzemarem la suma dels divisors de "y"
calculada.

    printf("La suma dels divisors de %d és %d i la suma dels divisors
de %d és %d.\n", x, sumdX, y, sumdY);
    return 0; // Fi del programa sense errors.
} // Tanquem el main.
```

---

1)

```
#include <stdio.h>
#include <stdlib.h>
#define N 100 // N té valor 100
void InitArray( float vect[N] ){ // Al ser un "void", no retorna res,
    simplement modifica l'array inicial.
    int i; // Declara que i és una variable de tipus numèrica i entera,
    i el vector tindrà un tamany N.
    for( i = 0; i < N; i++ ) // Recorre tots els valors de l'array.
        vect[i] = (rand()%100)/100.0; // Substitueix cada valor de
    l'array inicial per un valor aleatori entre 0 i 1, sense incloure l'1.
}

float MaxArray(float v[N]) { // Aquesta funció calcula i retorna el valor
    màxim de l'array.
    float max = v[0];
    for (int i = 1; i < N; i++) {
        if (v[i] > max) { // Si el següent nombre és més gran que
        l'anterior, es torna el "max", i així ens assegurem que sempre el trobarem.
            max = v[i];
        }
    }
    return max; // Retornem el valor màxim existent a l'array.
}

int main() {
    float v1[N], v2[N]; // V1 i V2 s'utilitzaran per a emmagatzemar els
    valors aleatoris creats per la funció "InitArray".
    srand(1234);
    InitArray(v1);
    InitArray(v2);
    // Calculem el valor màxim dels arrays v1 i v2.
    float max_v1 = MaxArray(v1);
    float max_v2 = MaxArray(v2);

    // Mostrar els resultats.
    printf("El valor màxim de v1 és: %.2f\n", max_v1); // El ".2" indica la
    quantitat de decimals que volem que es mostrin de cada nombre.
    printf("El valor màxim de v2 és: %.2f\n", max_v2);
    return 0;
}
```

---

2 + 3)

```
#include <stdio.h>
#include <stdlib.h>
#define N 100 // N té valor 100
void InitArray( float vect[N] ){ // Al ser un "void", no retorna res,
    simplement modifica l'array inicial.
        int i; // Declara que i és una variable de tipus numèrica i entera,
    i el vector tindrà un tamany N.
        for( i = 0; i < N; i++ ) // Recorre tots els valors de l'array.
            vect[i] = (rand()%100)/100.0; // Substitueix cada valor de
    l'array inicial per un valor aleatori entre 0 i 1, sense incloure l'1.
}
float MaxArray(float v[N]) { // Aquesta funció calcula i retorna el valor
    màxim de l'array.
        float max = v[0];
        for (int i = 1; i < N; i++) {
            if (v[i] > max) { // Si el següent nombre és més gran que
    l'anterior, es torna el "max", i així ens assegurem que sempre el trobarem.
                max = v[i];
            }
        }
        return max; // Retornem el valor màxim existent a l'array.
}
float SumArray(float v[N]) {
        float sum = 0.0;
        for (int i = 0; i <= N; i++) { // Recorre tots els elements.
            sum += v[i];
        }
        return sum; // Retornem la suma total dels elements de l'array.
}
int main() {
        float v1[N], v2[N]; // V1 i V2 s'utilitzaran per a emmagatzemar els
    valors aleatoris creats per la funció "InitArray".
        srand(1234);
        InitArray(v1);
        InitArray(v2);
        // Calculem el valor màxim dels arrays v1 i v2.
        float max_v1 = MaxArray(v1);
        float max_v2 = MaxArray(v2);
        // Calculem la suma dels elements de l'array.
        float sum_v1 = SumArray(v1);
        float sum_v2 = SumArray(v2);
        // Mostrar els resultats.
        printf("El valor màxim de v1 és: %.2f\n", max_v1); // El ".2" indica la
    quantitat de decimals que volem que es mostrin de cada nombre.
        printf("El valor màxim de v2 és: %.2f\n", max_v2);
        printf("El sumatori dels valors de v1 és: %.2f\n", sum_v1);
        printf("El sumatori dels valors de v2 és: %.2f\n", sum_v2);
        return 0;
}
```

---

4)

```
#include <stdio.h>
#include <stdlib.h>
#define N 100 // N té valor 100
void InitArray( float vect[N] ){ // Al ser un "void", no retorna res,
    simplement modifica l'array inicial.
    int i; // Declara que i és una variable de tipus numèrica i entera,
    i el vector tindrà un tamany N.
    for( i = 0; i < N; i++ ) // Recorre tots els valors de l'array.
        vect[i] = (rand()%100)/100.0; // Substitueix cada valor de
    l'array inicial per un valor aleatori entre 0 i 1, sense incloure l'1.
}

float MaxArray(float v[N]) { // Aquesta funció calcula i retorna el valor
    màxim de l'array.
    float max = v[0];
    for (int i = 1; i < N; i++) {
        if (v[i] > max) { // Si el següent nombre és més gran que
    l'anterior, es torna el "max", i així ens assegurem que sempre el trobarem.
            max = v[i];
        }
    }
    return max; // Retornem el valor màxim existent a l'array.
}

float SumArray(float v[N]) {
    float sum = 0.0;
    for (int i = 0; i <= N; i++) { // Recorre tots els elements.
        sum += v[i];
    }
    return sum; // Retornem la suma total dels elements de l'array.
}

void SumaV(float v1[N], float v2[N], float v3[N]) {
    for (int i = 0; i < N; i++) {
        v3[i] = v1[i] + v2[i];
    }
}
```

(segueix a baix)

```

int main() {

    float v1[N], v2[N], v3[N];

    srand(1234);
    InitArray(v1);
    InitArray(v2);

    // Calculem el valor màxim dels arrays v1 i v2.
    float max_v1 = MaxArray(v1);
    float max_v2 = MaxArray(v2);

    // Calculem la suma dels elements de l'array.
    float sum_v1 = SumArray(v1);
    float sum_v2 = SumArray(v2);

    SumaV(v1,v2,v3);

    printf("Primers 10 valors:\n");
    printf("v1: ");
    for (int i = 0; i < 10; i++) {
        printf("%.2f ", v1[i]); // Mostra els primers 10 valors d'v1
    }
    printf("\nv2: ");
    for (int i = 0; i < 10; i++) {
        printf("%.2f ", v2[i]); // Mostra els primers 10 valors d'v2
    }
    printf("\nv3: ");
    for (int i = 0; i < 10; i++) {
        printf("%.2f ", v3[i]); // Mostra els primers 10 valors d'v3
    } printf("\n"); // Salt de línia al final

    // Mostrar els resultats.
    printf("El valor màxim de v1 és: %.2f\n", max_v1); // El ".2" indica la
quantitat de decimals que volem que es mostrin de cada nombre.
    printf("El valor màxim de v2 és: %.2f\n", max_v2);
    printf("El sumatori dels valors de v1 és: %.2f\n", sum_v1);
    printf("El sumatori dels valors de v2 és: %.2f\n", sum_v2);
    return 0;
}

```