

## Programas en C

1.- Realice un programa en lenguaje C al cual se le pase como argumento dos números que representarán el número de filas y el número de columnas de una matriz y que rellena esta de la siguiente forma:

- La diagonal secundaria con 1
- La diagonal principal con 0
- El resto de la matriz deberá inicializarla con valores aleatorios entre 5 y 10

2.-Realizar una función en C a la que llamaremos Fibonacci, que pasándole un vector y el número de elementos del vector:

- Me rellene el vector con la sucesión de Fibonacci
- Me devuelva el número de elementos del vector cuyo contenido sea par en dicha sucesión

La sucesión de Fibonacci es : 1 1 2 3 5 8 13 21 34, ..... , nótese que los dos primeros elementos de la sucesión son fijos, es decir son 1 los dos, y a partir de estos se calcularán los siguientes.

3.- ¿Qué imprime este fragmento de código?

```
int aux=3;

if (aux%2)

printf("Creo q apruebo");

else

printf("Bueno siempre me queda otra convocatoria");
```

4. Qué imprime este fragmento de código?

```
int aux=3;

if (aux%2)

    printf("Creo q apruebo");

printf("Bueno siempre me queda otra convocatoria");
```

5. Realice un programa en lenguaje C al cual se le pase como argumento un número que representarán el orden de una matriz y que contenga una función inicializa para rellenar la matriz de la siguiente forma:

- La diagonal principal con 0
- El resto de la matriz deberá inicializarla con valores aleatorios entre 5 y 10

La función principal, comprobará que los argumentos son correctos y si no, dará un mensaje de error. Si los argumentos son correctos, inicializará una matriz del tamaño indicado por el argumento llamando a la función e imprimirá la matriz inicializada por pantalla. La función `inicializa` rellenará la matriz que recibe como argumento con los valores correspondientes.

6. Realice un programa en lenguaje C que lea una cadena de como máximo 100 caracteres y llame a una función `quita_espacios` que elimine los espacios en blanco de la cadena y la guarde en otra cadena del mismo tamaño.

A la función se le pasará como parámetro la cadena y la cadena destino y no devuelve nada. La función principal llamará a la función e imprimirá la cadena resultante de eliminar los espacios.

Ejemplo:

Cadena original: "Qué bonito es programar en C"

Salida: "QuébonitoesprogramarenC"

7. Realizar una **función** en C a la que llamaremos `cubos`, que pasándole un vector de tamaño 1000:

- Me rellene el vector con los cubos de los 1000 primeros números impares; es decir,  $1^3, 3^3, 5^3, 7^3, 9^3 \dots$
- Me devuelva el número de elementos del vector cuyo valor sea par.

### Programas en ensamblador del MIPS

1- Se pide realizar un programa que calcule la suma de los elementos de un vector inicialmente declarado en el segmento de datos mediante una subrutina. Dicha subrutina recibirá un puntero a dicho vector y su tamaño y devolverá la suma de todos los elementos del vector. **NOTA:** Recordar que las subrutinas reciben los argumentos en los registros `$a0 . . . $a3` y devuelven los resultados en `$v0` o `$v1`.

Un ejemplo de vector con 10 valores sería el siguiente:

```
vector10: .word 4, 1, 20, 30, 12, 17, 27, 29, 87, 98
```

Debería devolver 325 que es la suma de todos los números.

2. Realizar un programa que calcule la media de dos vectores. Dicho programa debe contener dos subrutinas:

- La primera subrutina recibirá los dos punteros a los vectores en dos registros y los tamaños de cada una de ellas en otros dos registros. Esta

subrutina llamará a la segunda subrutina dos veces (una por cada vector) que devolverá la suma de los elementos del vector que reciba. Finalmente la primera subrutina devuelve la media total (en número entero) de todos los valores. Por ejemplo: Si el primer vector es { 3,3,3,4 } y el segundo vector es { 4,4 } la suma total del primero es 13, la suma total del segundo es 8,  $13 + 8 = 21$  y  $21$  entre  $6$  (número total de elementos del vector 1 y 2) es  $3$ .

- La segunda subrutina recibirá como argumento una dirección de memoria a un vector y su tamaño. Se encargará de sumar todos los elementos del vector y devolverlo.

- NOTA: Recordar que las subrutinas reciben los argumentos en los registros \$a0...\$a3 y devuelven los resultados en \$v0 o \$v1.

3. Realizar un programa que solicite al usuario una cadena de texto y tres números enteros. A continuación, se llamará a una subrutina cuya tarea es la de contabilizar el número de caracteres de la cadena y el número de caracteres ASCII que posee dicha cadena en función de los 3 números solicitados al usuario. Es decir, si el usuario inserta:

Cadena: "La cadena de texto es esta"

Número 1: 97

Número 2: 99

Número 3: 101

El 97 es la letra 'a', el 99 la letra 'c' y el 101 la letra 'e' Con lo cual el programa debe devolver:

La cadena texto introducirá por el usuario posee 28 caracteres 4a, 1c, 5e

Pistas: Se pueden reservar 3 espacios de un byte en el segmento de datos para guardar los caracteres como "bytes" y posteriormente leerlos. Se pueden pasar en \$a0, \$a1, \$a2 y \$a3 todos los argumentos, no es necesario usar la pila. Se debe utilizar la pila para devolver al menos dos de los cuatro resultados que proporciona la subrutina.