

APELLIDO Y NOMBRE: _____ CANTIDAD DE HOJAS: _____

PARTE I – Complete el programa (2 puntos)

1 - Se desea realizar una función que reciba como parámetro una matriz de enteros de tamaño FILAS x COLS y que realice las siguientes tareas en el siguiente orden:

. Realizar un AND bit a bit de cada valor con el valor correspondiente a la columna anterior de la misma fila (la primera columna no se cambia)

. Realizar un OR bit a bit de cada valor con el valor correspondiente a la fila anterior de la misma columna (la primera fila no se cambia)

. Desplazar a izquierda los bits de cada una de las posiciones de la matriz, una cantidad de lugares equivalente a el valor guardado en la fila 0, columna 0.

Para realizar este ejercicio debe modificar la función (incompleta) que se indica a continuación de manera tal de que la misma cumpla con el enunciado. Complete las secciones indicadas con "____" con el código correspondiente. Complete las secciones indicadas con "__(*)__" con una o mas líneas de código (todas las que considere necesarias.) El resto de las líneas de código del programa deben quedar sin modificar, y en el orden mostrado:

```
#define FILAS 12
#define COLS 9
void procesar (int matriz[FILAS][COLS])
{
    int fila, col;
    for (_____)
    {
        for (fila=0; fila<FILAS; fila++)
        {
            __(*)__
        }
    }
    for (_____)
    {
        for (col=0; col<COLS; col++)
        {
            __(*)__
        }
    }

    for (_____)
    {
        for (_____)
        {
            __(*)__
        }
    }
}
```

PARTE II – Opción múltiple (2 puntos, solamente si están todas correctas)

2 - Marque la opción correcta que se corresponda con lo que el código imprimiría por pantalla:

<pre>int a=9, b=13, c=-3, d=8; int *bb, **cc; bb=&a; cc= &bb; (*bb)++; a= 5; **cc = **cc + 4; cc = &d; printf("%d", *bb);</pre>	<pre>char x (char a) { if (a<='w') printf("%c", x(a+1)); return 'w'; } void main() { printf("%c",x('u'));</pre>
<p>(A) 12 (B) 13 (C) 9 (D) 1 (E) 14 (F) 10</p>	<p>(A) uvw (B) wvu (C) uvww (D) wvuw (E) wwwww (F) uwwwww (G) www</p>
<pre>char *b = "3456789"; char * p = b; p += 5; printf("%s%s", p,b);</pre>	<pre>unsigned char a; a= ~((~0) << (1<<2)); printf("%d",a);</pre>
<p>(A) 5456789 (B) 893456789 (C) 54567893456789 (D) 34567943456794 (E) 3456794,3456789 (F) 34567943456789</p>	<p>(A) 240 (B) 1 (C) 9 (D) 15 (E) 14 (F) -240 (G) 4</p>

PARTE III – Programación (6 puntos)

Archivo	Contenido
Actores.dat	<u>Registros:</u> Apellido del Actor (char [50]) Nombre del actor (char [50]) Código de actor (int)
Peliculas.dat	<u>Registros:</u> Código de película (int) Nombre de la película (char [80]) Año de estreno (int)
Peliculas_Actores.txt (indica para cada película, que actores trabajaron en la misma)	<u>Archivo CSV con:</u> Código de película (numérico) Código de actor (numérico)

- 1- Realizar una función que reciba como parámetro un número entero que representa un código de actor y que imprima por pantalla un listado (incluyendo nombre y año de estreno) de todas las películas donde el único actor que actuó fue el recibido como parámetro.
- 2- Realizar una función que reciba como parámetros el APELLIDO de un actor y un AÑO; y que imprima por pantalla todos los nombres de las películas donde haya actuado algún actor de ese apellido (pueden ser varios actores distintos) y que haya sido estrenada con posterioridad al AÑO recibido por parametro.

Condiciones para todos los ejercicios:

- Declarar todos los "struct" y "typedef" que utilice para resolver el ejercicio.
- No desperdiciar memoria. Liberar todos los espacios que se hayan reservado y que no se utilicen.
- En los ejercicios de programación se pueden programar funciones adicionales a las pedidas.