

Programación Orientada a Objetos 1 Guía de Ejercicios II Pregrado

1. Elabore un programa que solicite al usuario el ingreso de un número F y un número C. Donde F representa el número de filas y C el número de columnas de la matriz dinámica a crear. Validar que F sea mayor a 5 y C sea un número entero par. Se llenará con valores numéricos aleatorios, luego de lo cual se generan dos nuevas matrices que resultan de dividir verticalmente la matriz inicial. Imprimar las dos nuevas matrices.

Listing 1: Ejemplo 1

```
Ingrese cantidad de filas:4
Ingrese cantidad de filas:5
Ingrese cantidad de filas:6
Ingrese cantidad de columnas:6
Matriz original:
                                                  32
                                                             61
       81
                  54
                             68
                                       54
       32
                  71
                             51
                                       67
                                                  44
                                                             60
       43
                  34
                                                             90
                             56
                                       84
                                                  55
                  24
                                       27
                                                             92
       41
                             91
                                                   9
       25
                  50
                             69
                                       26
                                                  90
                                                             43
       81
                   6
                              2
                                       21
                                                   3
                                                             54
Matriz 1:
       81
                  54
                             68
       32
                  71
                             51
       43
                  34
                             56
       41
                  24
                             91
       25
                  50
                             69
       81
                   6
                              2
Matriz 2:
       54
                  32
                             61
       67
                  44
                             60
       84
                  55
                             90
       27
                   9
                             92
                  90
       26
                            43
       21
                   3
                            54
Process finished with exit code 0
```

```
Ingrese cantidad de columnas:3
Ingrese cantidad de columnas:7
7
Ingrese cantidad de columnas:8
Matriz original:
    51
             59
                     13
                                                              26
                             95
                                     47
                                              42
                                                      14
    51
                     94
                             56
             98
                                     37
                                              96
                                                      51
                                                              64
    28
             19
                     97
                             51
                                              49
                                                      82
                                                               3
                                     33
    37
                                                      75
            81
                     82
                             10
                                     39
                                              56
                                                              36
     7
            54
                     80
                             17
                                     72
                                              19
                                                      25
                                                               9
    73
             22
                     47
                             74
                                     72
                                              96
                                                       6
                                                              47
    31
                             45
                                              49
                                                      61
                                                               6
             12
                     18
                                     81
Matriz 1:
    51
             59
                     13
                             95
                     94
    51
             98
                             56
    28
             19
                     97
                             51
    37
            81
                     82
                             10
     7
            54
                     80
                             17
    73
             22
                     47
                             74
    31
             12
                     18
                             45
Matriz 2:
    47
            42
                     14
                             26
    37
            96
                     51
                             64
    33
            49
                     82
                              3
    39
            56
                     75
                             36
    72
                              9
             19
                     25
    72
             96
                      6
                             47
    81
             49
                     61
                              6
Process finished with exit code 0
```

Listing 3: Ejemplo 3

```
Ingrese cantidad de filas:8
Ingrese cantidad de columnas:9
Ingrese cantidad de columnas:4
Matriz original:
      94
                73
                           75
                                     63
      28
                 16
                           79
                                     62
                 0
      35
                           75
                                     57
                           79
                                     29
      81
                41
      61
                34
                           3
                                     68
      43
                92
                           47
                                     48
      16
                72
                           42
                                     18
       9
                36
                           26
                                     40
```

```
Matriz 1:
       94
                   73
       28
                   16
                    0
       35
       81
                   41
       61
                   34
                   92
       43
       16
                   72
        9
                   36
Matriz 2:
       75
                   63
       79
                   62
       75
                   57
       79
                   29
        3
                   68
       47
                   48
       42
                   18
       26
                   40
Process finished with exit code 0
```

2. Alessia es una niña muy afortunada y se ganó un nuevo juguete "Don't Break the Ice", que es un juego de mesa y se juega con un conjunto de "bloques de hielo" de plástico (generalmente 6x6), un mazo en miniatura y un soporte. Un bloque de hielo es más grande que el resto (Normalmente 2x2), y ya sea un hombre, un oso polar o un Pingüino, se encuentra en este bloque. Los jugadores se turnan para quitar bloques tocando con el mazo. El juego termina cuando un jugador que "rompe el hielo", lo que provoca que el hombre, el oso o Pingüino se caiga. El jugador que haya eliminado la mayoría de los bloques sin "romper el hielo" es el ganador.



Figure 1: Juego "Don't Break the Ice"

Alessia, decide experimentar con su nuevo juguete y extrañamente cuando golpea el bloque grande los demás bloques se caen, pero, cuando golpea un bloque pequeño se cae solo el bloque golpeado. Desarrolle un algoritmo que permita modelar las ocurrencias de Alessia. Para ello, considere utilizar una matriz dinámica de 6x6, además, cada bloque pequeño estará representado por cualquier número mayor a 2 o menor a 10, el bloque grande estará representado por 1s, los bloques caídos serán representados con 0.

Input

• La primera línea contiene los elementos de la matriz 6x6 que son números enteros.

3	3	6	7	8	8
5	7	8	1	1	3
6	9	5	1	1	8
7	6	9	3	8	8
3	3	6	7	7	8
3	3	6	7	8	3

• La segunda línea contiene dos enteros positivos F y C $(0 \le F \le 6; 0 \le C \le 6)$, lo que corresponde a la ubicación del bloque a golpear.

Ouput

• Se espera Imprimir una nueva matriz con los bloques caídos.

Algunos ejemplos de este programa serían:

Listing 4: Ejemplo 1

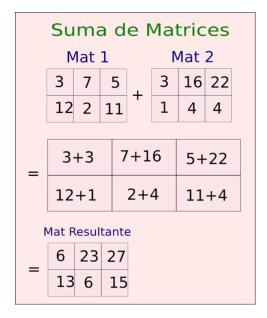
```
Ingrese ubicacion del bloque: 0 2
Output:
3 3 0 7 8 8
5 7 8 1 1 3
6 9 5 1 1 8
7 6 9 3 8 8
3 3 6 7 7 8
3 3 6 7 8 3
```

Listing 5: Ejemplo 2

```
Ingrese ubicacion del bloque: 1 4
Output:
0 0 0 0 0 0
0 0 0 0 0
0 0 0 0 0
0 0 0 0 0
0 0 0 0 0
0 0 0 0 0
0 0 0 0 0
```

Listing 6: Ejemplo 3

```
Ingrese ubicacion del bloque: 2 3
Output:
0 0 0 0 0 0
0 0 0 0 0
0 0 0 0 0
0 0 0 0 0
0 0 0 0 0
0 0 0 0 0
0 0 0 0 0
```



3. Elabore un algoritmo que permita sumar dos matrices y almacenar el resultado en una nueva matriz, tal como se muestra en la siguiente imagen:

Las funciones a implementar son las que se indica en la función main:

```
#include <iostream>
using namespace std;
int main()
{
    int n, m;
    cin>>n>>m;//dimension de la matriz
    int **ma1 = llenar(n, m);//llenar desde teclado
    int **ma2 = llenar(n, m);//llenar desde teclado
    int **maTsum = suma(ma1, ma2, n, m);
    cout <<"---\n";
    mostrar(maTsum, n, m);
    return 0;
}</pre>
```

4. Dado un programa que genera una matriz cuadrada de forma dinamica de valores entre 0 y 9 como se muestra en el código, corregir y completar el codigo faltante en la función **main** y desarrollar una función **es_triangulo_superior** (declaración y definición) que determine si la matriz es triangular superior. Se dice que una matriz es triangular superior si todos los valores por debajo de la diagonal principal son cero, Ejemplo:

```
6
           5
           7
0
                8
                3
0
     0
           9
0
           0
                2
                1
                2
0
           2
                2
0
     0
           2
                2
0
           0
```

Listing 7: Codigo

```
#include <iostream>
#include "matrices.h"

int main() {
    range n = 0;
    cout >> "tamano:"; cin >> n;

    number ** matriz = generar_matriz(nfil, ncol);
    actualizar_matriz(matriz, nfil, ncol);
    imprimir_matriz(matriz, nfil, ncol);

    if (es_triangulo_superior(matriz, nfil));
}
```

Algunos ejemplos de diálogo de este programa serían:

Listing 8: Ejemplo 1

Listing 9: Ejemplo 1

```
tamano: 4
1 3 4 6
0 4 5 4
0 0 7 6
0 4 0 2
No es triangular Superior.
```

 $5.\,$ Dado una matriz bidimensional de tamaño 5 filas y 7 columnas generada de forma dinámica.

Realizar las siguientes operaciones:

- 1. Llenar la matriz con números aleatorios entre -100 y 100.
- 2. Mostrar la matriz generada.
- 3. Mostrar la matriz con todos los números negativos existente. Mostrar "-" en caso no cumpla la condición.
- 4. Mostrar la matriz con todos los números positivos existentes. Mostrar "-" en caso no cumpla la condición.

Tomar en cuenta lo siguiente:

- Realizar la asignación en la memoria de forma dinámica.
- Compilación: $g + + c \ main.cpp \ funciones.h \ implementacion.cpp$

Listing 10: Ejemplo 1

			<u>J</u>				
Matriz ge	nerada:						
-9	-48	-96	63	97	17	35	
21	-99	26	-73	21	-24	16	
-67	24	-71	42	91	-28	81	
36	-64	41	-6	-50	9	-92	
44	-91	99	-65	-87	-97	50	
Mostrar m	atriz con	numeros	negativos:				
-9	-48	-96	-	_	_	_	
_	-99	_	-73	_	-24	_	
-67	_	-71	_	_	-28	_	
_	-64	_	-6	-50	_	-92	
_	-91	_	-65	-87	-97	-	
Mostrar m	atriz con	numeros	positivos:				
_	_	_	63	97	17	35	
21	_	26	-	21	_	16	
_	24	-	42	91	_	81	
36	_	41	-	-	9	-	
44	_	99	_	-	-	50	

Listing 11: Ejemplo 2

					L -			
Ma	triz genera	da:						
	7	29	61	-34	72	-21	-73	
	1	21	50	-65	59	-79	-36	
	-31	-80	39	-75	67	76	85	
	-51	94	-8	-12	94	2	-4	
	-90	63	-61	-31	-55	-100	36	
Mo	strar matri	z con	numeros	negativos:				
	_	_	_	-34	_	-21	-73	
	_	_	_	-65	_	-79	-36	
	-31	-80	_	-75	_	_	_	
	-51	_	-8	-12	_	_	-4	
	-90	_	-61	-31	-55	-100	_	
Mo	strar matri	z con	numeros	positivos:				
	7	29	61	-	72	-	_	
	1	21	50	-	59	-	-	
	_	_	39	_	67	76	85	
	-	94	-	-	94	2	-	
	_	63	-	_	-	-	36	
Мо	-51 -90	- z con 29 21 - 94	-61 numeros 61 50	-65 -75 -12 -31	72 59 67	-79 - - -100 - - 76	-36 -4 - - - 85 -	

Listing 12: Ejemplo 3

		=	2. 2. 2. 2. 2. 2. 2. 2. 2. 2. 2. 2. 2. 2	I			
Matriz ge	enerada:						
33	11	87	44	-39	26	-90	
26	-27	-86	20	-77	64	65	
92	66	-14	83	50	30	74	
-90	-34	85	-50	70	-2	-47	
61	-76	30	-54	36	-31	91	
Mostrar m	natriz con	numeros	negativos:				
_	-	-	-	-39	_	-90	

_	-27	-86	-	-77	_	-	
_	-	-14	_	_	-	_	
-90	-34	_	-50	_	-2	-47	
_	-76	_	-54	_	-31	_	
Mostrar r	matriz con	numeros	positivos:				
33	11	87	44	_	26	-	
26	-	_	20	_	64	65	
92	66	_	83	50	30	74	
_	-	85	-	70	-	-	
61	_	30	_	36	_	91	

Listing 13: Ejemplo 4

			Listing 19. Lje	mpio i			
Matriz g	generada:						
71	17	73	81	-45	57	-80	
50	-37	77	41	16	-78	-76	
-15	5 45	-47	38	-35	31	47	
-27	7 –99	1	-6	-44	-54	33	
89	87	-2	13	4	-77	-6	
Mostrar	matriz co	n numeros	negativos				
-		-	_	-45	_	-80	
-	- 37	_	_	_	-78	-76	
-15	5 -	-47	_	-35	_	-	
-27	7 –99	_	-6	-44	-54	-	
-		-2	_	_	-77	-6	
Mostrar	matriz co	n numeros	positivos				
71	17	73	81	_	57	-	
50) –	77	41	16	_	-	
-	- 45	-	38	_	31	47	
-		1	_	_	_	33	
89	87	-	13	4	-	-	

6. Usted tiene información de 3 meses de la cantidad de ventas de 3 productos de la canasta basica de alimentos:

producto	enero	febrero	marzo
leche	10	20	30
huevos	30	20	10
pan	5	10	15

- Usando arreglos con memoria dinamica haga un programa para leer los datos.
- Lea precios por cada mes.
- Imprima los promedios por cada producto.
- Libere los recursos usados por el array.

Listing 14: Ejemplo 1

```
Ingrese precio (leche): 10
Ingrese precio (leche): 20
```

```
Ingrese precio (leche): 30
Ingrese precio (huevo): 30
Ingrese precio (huevo): 20
Ingrese precio (huevo): 10
Ingrese precio (pan): 5
Ingrese precio (pan): 10
Ingrese precio (pan): 15

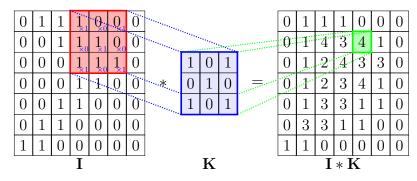
Promedio leche:20
Promedio huevo:20
Promedio pan: 10
```

7. La mayoria de los filtros que usas en Facebook o Instagram usan matrices de convolución. Convolución es el tratamiento de una matriz por otra que se llama "kernel" K. El filtro matriz de convolución usa una primera matriz que es la imagen que será tratada I. La imagen es una colección bidimensional de píxeles en coordenada rectágular. El kernel usado depende del efecto deseado y los de tamaño de 3x3, son las más usadas y son suficiente para los efectos deseados.

Escribe un programa que permita al usuario ingresar las dimensiones de la matriz \mathbf{I} . Luego el programa deberá crear de manera dinámica la matriz con elementos aleatorios entre cero y uno. Después de esto deberás aplicar el filtro de convolución \mathbf{K} tal como se muestra en la imagen y generar una nueva matriz $\mathbf{I}\mathbf{K}$. Los elementos de la nueva matriz $\mathbf{I}\mathbf{K}$ serán calculados usando el filtro de convolución. Por ejemplo el elemento $\mathbf{I}\mathbf{K}[1][4]$ será calculado de la siguiente forma:

$$\mathbf{IK}[1][4] = \mathbf{I}[0][3] \cdot \mathbf{K}[0][0] + \mathbf{I}[0][4] \cdot \mathbf{K}[0][1] + \mathbf{I}[0][5] \cdot \mathbf{K}[\mathbf{0}][2] + \\
\mathbf{I}[1][3] \cdot \mathbf{K}[1][0] + \mathbf{I}[1][4] \cdot \mathbf{K}[1][1] + \mathbf{I}[1][5] \cdot \mathbf{K}[\mathbf{1}][2] + \\
\mathbf{I}[2][3] \cdot \mathbf{K}[2][0] + \mathbf{I}[2][4] \cdot \mathbf{K}[2][1] + \mathbf{I}[2][5] \cdot \mathbf{K}[\mathbf{2}][2]$$
(1)

Hint: Ten en cuenta que los elementos del borde de la matriz no son tomados en cuenta.



Listing 15: Ejemplo 1

```
Ingrese la cantidad de filas:5
Ingrese la cantidad de columnas:7
I:
1
          0
                     1
                                1
                                           0
                                                     1
                                                                0
1
          1
                     1
                                1
                                           1
                                                     0
                                                                1
0
          1
                     0
                                1
                                           1
                                                     1
                                                                0
          1
                     0
                                1
                                           1
                                                     1
                                                                1
1
1
          1
                     1
                                1
                                           0
                                                     1
                                                                1
```

K:						
1	0	1				
0	1	0				
1	0	1				
IK:						
1	0	1	1	0	1	0
1	3	4	3	5	1	1
0	4	4	4	4	5	0
1	3	4	3	5	3	1
1	1	1	1	0	1	1

Listing 16: Ejemplo 2

```
Ingrese la cantidad de filas:4
         la cantidad de columnas:3
Ingrese
I:
1
          0
                    0
          0
                    1
1
1
                    1
          1
1
          1
K:
1
          0
                    1
                    0
0
          1
1
IK:
          0
                    0
1
1
          3
                    1
1
          4
                    1
                    0
1
          1
```

Listing 17: Ejemplo 3

```
Ingrese la cantidad de filas:2
Ingrese la cantidad de columnas:2
I:
1
         0
0
         0
K:
                   1
                   0
0
         1
         0
                   1
1
IK:
1
         0
         0
0
```

- 8. Crear un programa que utilice ASIGNACION DINÁMICA DE MEMORIA y que permita crear una matriz de N x M:
 - Los valores de N (filas) y M (columnas) deben ser ingresados desde teclado
 - Rellenar la matriz con números aleatorios, el rango a generar es de 1 a 10

- Mostrar la matriz en pantalla
- Indicar es la suma de las COLUMNAS que tiene el mínimo valor, si hay varias columnas que tienen el mínimo valor, debe indicarlas

Listing 18: Ejemplo 1

N=3					
M=4					
1	3	3	7		
5	4	1	0		
1	0	9	2		
Respu	esta				
Minim	o Valor:	: 7			
COLUM	NAS: 0,	1			