



EDUCACIÓN
SECRETARÍA DE EDUCACIÓN PÚBLICA



TECNOLÓGICO NACIONAL DE MÉXICO

Instituto Tecnológico de Ensenada

Docente: Guillermo Alejandro Chávez Sánchez

Materia: Estructura de Datos

**Título del trabajo: Práctica de Laboratorio: Manejo
de Matrices en C**

Elaborado por: Rosa Santos Meza

No. Control: 24761148

Ensenada Baja California a 6 de octubre de 2025

Durante la realización de esta práctica, los principales desafíos fueron:

1. Manejo de dimensiones en C:

Algunas funciones necesitan saber la cantidad de filas y columnas reales de la matriz. Inicialmente intenté declarar matrices con tamaños fijos en las funciones, pero al usar las transpuestas o sub matrices, el compilador marcaba errores. La solución fue usar un tamaño máximo (`#define MAX 10`) y pasar siempre las dimensiones reales como parámetros.

2. Validación de entradas del usuario:

Para los ejercicios que requieren ingreso manual, necesité validar que el usuario ingresara números enteros. Esto se manejó con `scanf` y comprobando el valor del retorno.

3. Operaciones con matrices cuadradas y no cuadradas:

Algunos ejercicios requieren que la matriz sea cuadrada(diagonales) y otros no (transpuesta). Fue importante adaptar funciones como mostrar Matriz para ser flexibles.

4. Modularidad y organización:

Inicialmente intenté hacer los ejercicios en `main`, pero era confuso y poco legible. Separar cada ejercicio en funciones individuales facilitó la lectura, depuración y reutilización del código.

5. Uso de números aleatorios y reproducibilidad:

Decidí usar(`time(NULL)`) para generar números aleatorios distintos en cada ejecución. Esto me permitió probar los programas con distintos datos y validar los resultados de suma, promedio, máximo, mínimo.

Descripción de la práctica

El objetivo de esta práctica fue reforzar los conocimientos sobre matrices en C, incluyendo su declaración, inicialización, recorridos con ciclos anidados y operaciones básicas. Se realizaron ejercicios que incluyen:

1. Inicialización de matrices.
2. Llenado manual de matrices.
3. Contar números pares e impares.
4. Calcular suma y promedio de los elementos.
5. Encontrar valores máximo y mínimo.
6. Sumar filas y columnas.
7. Suma de matrices.
8. Resta de matrices.
9. Multiplicación de matrices.
10. Matriz transpuesta.
11. Diagonal principal y secundaria.
12. Buscar elementos específicos en la matriz.

Todos los ejercicios fueron implementados de manera modular, utilizando funciones separadas para cada operación, validando entradas de usuario y utilizando números aleatorios en los casos que correspondía.

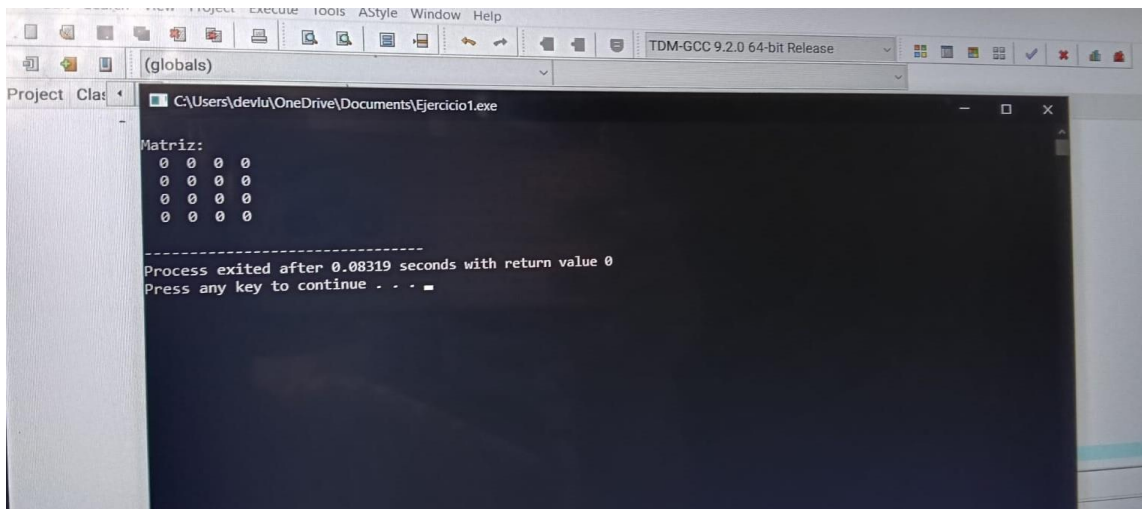
Conclusión:

La práctica ayudó a consolidar los conceptos de matrices en C, manejo de ciclos anidados, funciones, validación de datos y operaciones como

suma, resta, multiplicación y transposición. También fortaleció la organización modular del código, haciendo que cada ejercicio sea independiente y fácil de probar.

Además, se reforzó la importancia de planificar la estructura del programa y de utilizar funciones reutilizables para mantener el código limpio, legible y fácil de mantener.

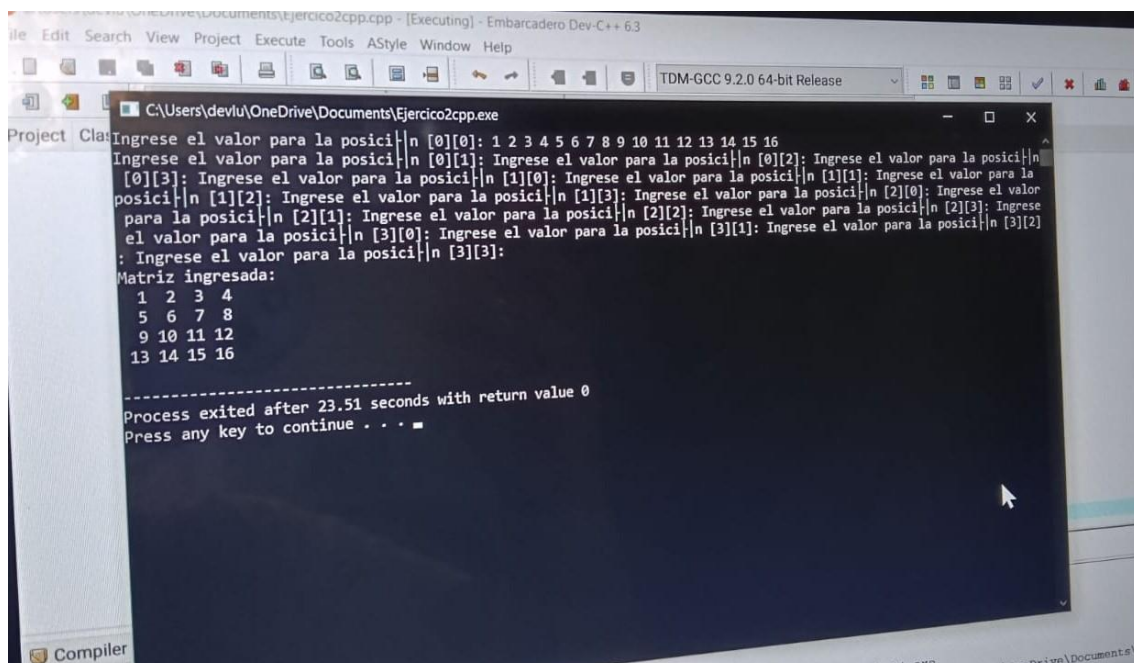
Capturas de cada uno de los 12 códigos.



A screenshot of a C++ program execution in a terminal window. The program displays a 4x4 matrix of zeros. The text in the terminal is as follows:

```
Matriz:
0 0 0 0
0 0 0 0
0 0 0 0
0 0 0 0

-----
Process exited after 0.08319 seconds with return value 0
Press any key to continue . . .
```



A screenshot of a C++ program execution in a terminal window. The program prompts the user to enter values for a 4x4 matrix. The text in the terminal is as follows:

```
Ingrese el valor para la posición [0][0]: 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16
Ingrese el valor para la posición [0][1]: Ingrese el valor para la posición [0][2]: Ingrese el valor para la posición [0][3]:
Ingrese el valor para la posición [1][0]: Ingrese el valor para la posición [1][1]: Ingrese el valor para la posición [1][2]: Ingrese el valor para la posición [1][3]:
Ingrese el valor para la posición [2][0]: Ingrese el valor para la posición [2][1]: Ingrese el valor para la posición [2][2]: Ingrese el valor para la posición [2][3]:
Ingrese el valor para la posición [3][0]: Ingrese el valor para la posición [3][1]: Ingrese el valor para la posición [3][2]: Ingrese el valor para la posición [3][3]:
Matriz ingresada:
1 2 3 4
5 6 7 8
9 10 11 12
13 14 15 16

-----
Process exited after 23.51 seconds with return value 0
Press any key to continue . . .
```

```
File Edit Search View Project Execute Tools AStyle Window Help
(globals) TDM-GCC 9.2.0 64-bit Release
Project Clas Ejemplo3.cpp
1
2 #include <stdio.h>
3
4
5 Matriz generada:
6 0 9 9 6
7 0 0 9 3
8 1 4 3 7
9 5 2 3 9
10
11 N|meros pares: 6
12 N|meros impares: 10
13
14 -----
15 Process exited after 0.07928 seconds with return value 0
16 Press any key to continue . . .
17
18
19
20
21
```

```
File Edit Search View Project Execute Tools AStyle Window Help
(globals) TDM-GCC 9.2.0 64-bit Release
Project Clas Ejercicio4.exe
Matriz:
0 4 0 8
6 3 7 0
3 1 4 5
7 3 9 2
Suma: 62
Promedio: 3.88
-----
Process exited after 0.08478 seconds with return value 0
Press any key to continue . . .
```

```
File Edit Search View Project Execute Tools AStyle Window Help
(globals) TDM-GCC 9.2.0 64-bit Release
Project Clas Ejercicio5.exe
Matriz:
16 60 69 64
59 95 3 36
45 5 47 39
81 85 41 79
M|ximo: 95 en [1][1]
M|nimo: 3 en [1][2]
-----
Process exited after 0.08315 seconds with return value 0
Press any key to continue . . .
```

```
C:\Users\devlu\OneDrive\Documents\Ejercicio6.exe

Matriz y suma por filas:
7 0 5 8 | Suma fila 0: 20
3 7 8 1 | Suma fila 1: 19
0 9 3 3 | Suma fila 2: 15
5 3 1 0 | Suma fila 3: 9

Suma columnas: 15 19 17 12

-----
Process exited after 0.08588 seconds with return value 0
Press any key to continue . . .
```

```
File Edit Search View Project Execute Tools AStyle Window Help
TDM-GCC 9.2.0 64-bit Release
(globals)
Project Clat Ejercicio7.cpp
12 for (int j = 0; j < COLS; j++)
13     tabla[i][j] = rand() % 10;
14 }
15 //
16 //
17 //
18 //
19 //
20 //
21 //
22 //
23 //
24 //
25 //
26 //
27 //
28 //
29 //
30 //
31 //
32 //
33 //
34 //
35 //

C:\Users\devlu\OneDrive\Documents\Ejercicio7.exe

Matriz A:
8 7 5 6
3 7 9 6
4 2 7 2
4 8 8 2

Matriz B:
6 3 4 2
7 1 2 9
0 5 0 7
2 3 7 7

Matriz C:
14 10 9 8
10 8 11 15
4 7 7 9
6 11 15 9

-----
Process exited after 0.1041 seconds with return value 0
Press any key to continue . . .
```

```
C:\Users\devlu\OneDrive\Documents\Ejercicio8cpp.exe

Matriz A:
0 4 4 2
2 3 4 4
7 6 4 1
2 0 1 5

Matriz B:
5 3 9 9
0 2 3 9
5 8 7 0
3 4 1 1

Matriz C:
-5 1 -5 -7
2 1 1 -5
2 -2 -3 1
-1 -4 0 4

-----
Process exited after 0.08886 seconds with return value 0
Press any key to continue . . .
```



```
21 printf("%3d", matriz[i][j]);
22 printf("\n");
23
24
25
26 Matriz A:
27 3 5 5
28 1 9 9
29
30 Matriz B:
31 7 3
32 8 2
33 5 4
34
35 Matriz C:
36 86 39
37 124 57
38
39 -----
40 Process exited after 0.08802 seconds with return value 0
41 Press any key to continue . . .
```

```
Matriz A:
6 9 8 2
4 6 6 5
5 2 9 8

Matriz T:
6 4 5
9 6 2
8 6 9
2 5 8

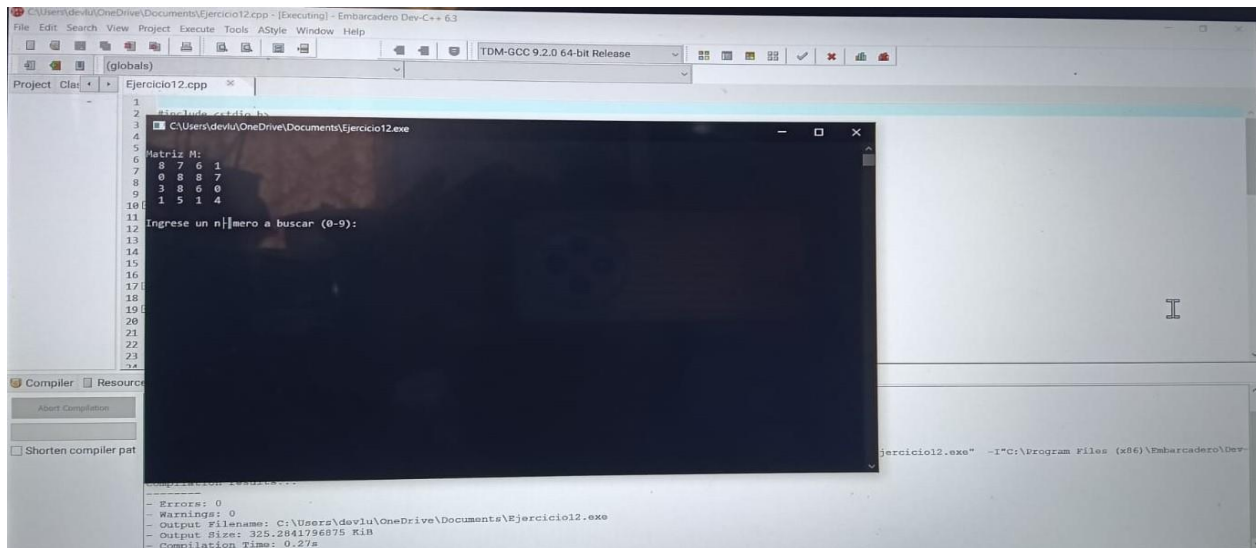
-----
Process exited after 0.08614 seconds with return value 0
Press any key to continue . . .
```

```
Matriz M:
0 6 2 1
0 9 5 3
3 5 4 5
5 9 6 6

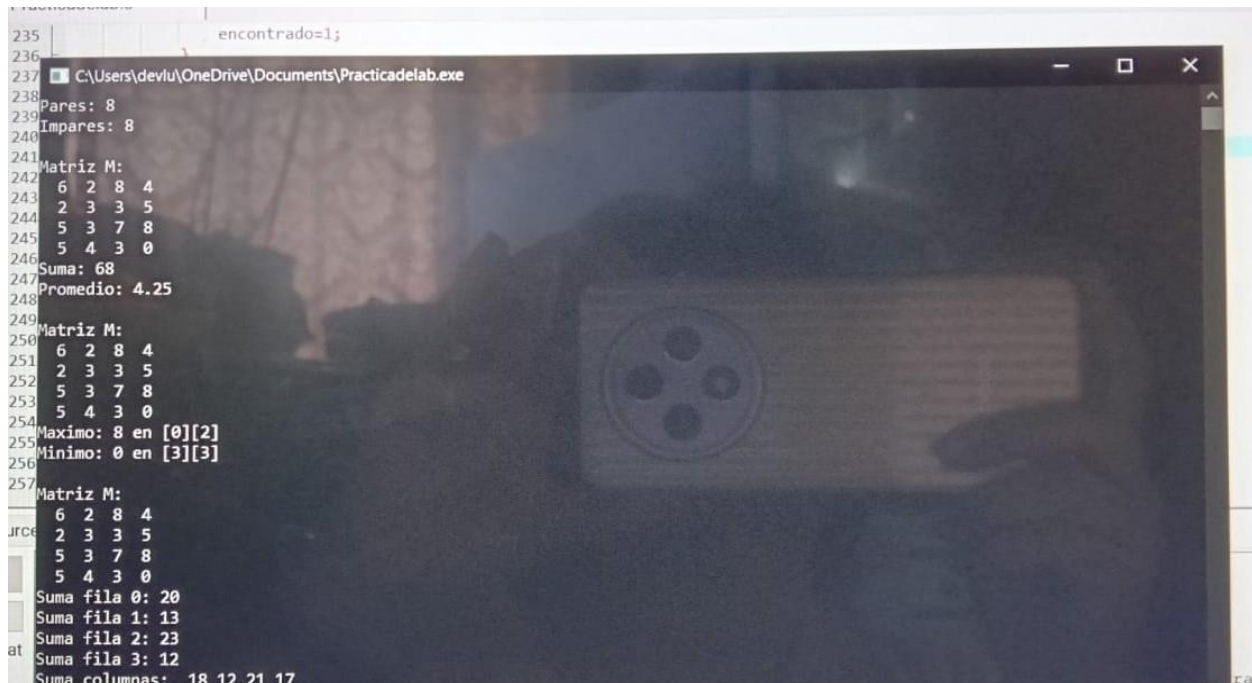
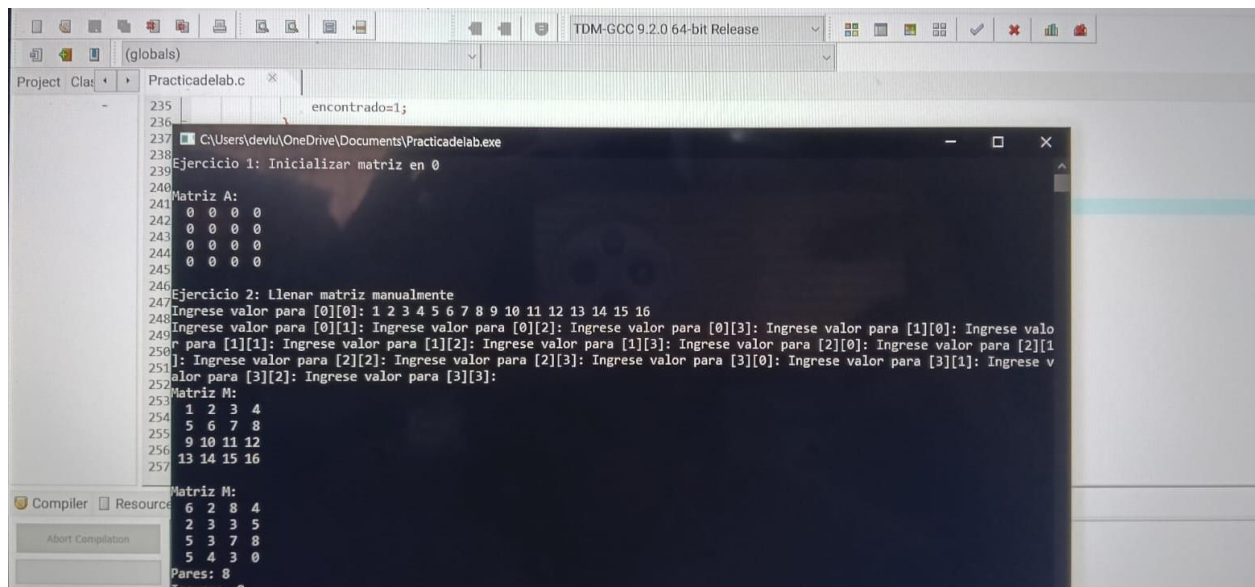
Diagonal principal: 0 9 4 6
Suma diagonal principal: 19

Diagonal secundaria: 1 5 5 5
Suma diagonal secundaria: 16

-----
Process exited after 0.08949 seconds with return value 0
Press any key to continue . . .
```



Código fuente compilado




```
237 C:\Users\devi\OneDrive\Documents\PracticadelaBase
238 Suma columnas: 18 12 21 17
239
240 Matriz A:
241 6 2 8 4
242 2 3 3 5
243 5 3 7 8
244 5 4 3 0
245
246 Matriz B:
247 5 2 0 1
248 7 9 9 1
249 2 2 6 6
250 0 7 9 2
251
252 Matriz C:
253 11 4 8 5
254 9 12 12 6
255 7 5 13 14
256 5 11 12 2
257
258 Matriz A:
259 6 2 8 4
260 2 3 3 5
261 5 3 7 8
262 5 4 3 0
263
264 Matriz B:
265 5 2 0 1
266 7 9 9 1
267 2 2 6 6
268
269 - Errores: 0
270 - Advertencias: 0
```

```
0 7 9 2
Matriz C:
1 0 8 3
-5 -6 -6 4
3 1 1 2
5 -3 -6 -2
Matriz A:
6 2 8
4 2 3
Matriz B:
3 5
5 3
7 8
Matriz C:
04100
4) 50
Matriz A:
6 2 8 4
2 3 3 5
5 3 7 8
Matriz T:
6 2 5
2 3 3
8 1 7
```