



Universidad Autónoma de Baja California

FACULTAD DE CIENCIAS QUÍMICAS E INGENIERÍA



Reporte de Laboratorio

Lenguaje C

Práctica No. 8

Manejo de arreglos

Alumno

Joshua Osorio Osorio - 1293271

Docente

Fernando Elihonai Saucedo Lares

20 de marzo de 2025

ÍNDICE

Objetivo.....	2
Introducción.....	2
Lista de materiales.....	2
Desarrollo.....	3
Calcular el perímetro de cualquier tipo de figura o polígono regular.....	3
Identificar si un número es primo.....	4
Identificar si el número capturado es múltiplo de algún número base, definir Cuál es y cuántas veces cabe en dicho número.....	4
Identificar si un número tiene raíz cuadrada exacta.....	4
Conclusiones.....	4
Bibliografía.....	5

Objetivo

Que los estudiantes adquieran habilidades prácticas en el manejo de estructuras de datos como arreglos unidimensionales y matrices bidimensionales en un lenguaje de programación.

Introducción

Ejercicio 1: Escribe un programa que solicite al usuario dos arreglos de números enteros de tamaño n (definido por el usuario) y luego los sume posición por posición, almacenando el resultado en un tercer arreglo.

Ejercicio 2: Escribe un programa que solicite una matriz de tamaño $m \times n$, la almacene y luego calcule su transpuesta.

Ejercicio 3: Escribe un programa que cuente cuántos números pares e impares hay en una matriz

Lista de materiales

- Computadora
- software: mingw
- Editor de texto.

Desarrollo

```

C/C++
/*-----
-----
-LABORATORIO: 7 Matrices y arreglos.
-Nombre: Joshua Osorio
-Materia: LENGUAJE C 531
-Fecha: Marzo/12/2025
-----*/

#include <stdio.h>
#include <time.h>
#include <stdlib.h>
#include <string.h>

#define renglon 3
#define columna 3

int aleatorio(void);
void RellenarArr(int arr[renglon][columna]);
void SumarArreglo(int arr1[renglon][columna], int arr2[renglon][columna], int
res[renglon][columna]);

void Transpuesta(int arr[renglon][columna]);
void ContarNumParesArr(int arr[renglon][columna]);

int main(void)
{
    int opc = 0;

    int res[renglon][columna];

    int arr1[renglon][columna];
    RellenarArr(arr1);
    int arr2[renglon][columna];
    RellenarArr(arr2);

    /*
    renglon = 0, columna = 0,  renglon1 = 0, columna1 = 0;

    printf("\nIndica el tamaño del arreglo 1 Renglon x columna:");

```

```

printf("\nRenglon:\t");
scanf("%d", &renglon);
printf("columna:\t");
scanf("%d", &columna);

int arr1[renglon][columna];
RellenarArr(arr1);

printf("\nIndica el tamaño del arreglo 1 Renglon x columna:");

printf("\nRenglon:\t");
scanf("%d", &renglon1);
printf("columna:\t");
scanf("%d", &columna1);
*/
printf("\n");
do
{
    /* Menu de la practica. */
    printf("\n-----Menu-----");
    printf("\n\t1) Sumar Arreglo posición x posición");
    printf("\n\t2) Transpuesta del arreglo 2");
    printf("\n\t3) Contar cuantos numeros pares hay");
    printf("\n\t4) Salir");

    /* Capturar opción*/
    printf("\nSelecciona una opcion:\t");
    scanf("%d", &opc);

    switch (opc)
    {
    case 1:
        printf("\n\tSumar Arreglo posición x posición\n");
        SumarArreglo(arr1, arr2, res);
        break;
    case 2:
        printf("\n\tTranspuesta del arreglo 2\n");
        Transpuesta(arr2);
        break;
    case 3:
        printf("\n\tContar cuantos numeros pares hay");
        ContarNumParesArr(arr1);

```

```

        break;
    case 4:
        printf("\nSaliendo del programa");
        break;
    default:
        printf("\nOpcion no disponible =(");
        break;
    }
    printf("\n\nPresione cualquier tecla para continua");
    fflush(stdin);
    getchar();
} while (opc != 4);

return 0;
}

int aleatorio(void)
{
    int num;
    num = rand() % 10 + 1;
    return num;
}

void RellenarArr(int arr[renglon][columna])
{
    int num = 0;
    printf("\n Capturar numeros del arreglo.\n");
    for (int r = 0; r < renglon; r++)
    {
        for (int c = 0; c < columna; c++)
        {
            /*
            printf("%d, %d = ", r, c);
            scanf("%d", &num);
            */
            arr[r][c] = aleatorio();
            printf("%d ", arr[r][c]);
        }
        printf("\n");
    }
}

void SumarArreglo(int arr1[renglon][columna], int arr2[renglon][columna], int
res[renglon][columna])

```

```

{
    for (int r = 0; r < renglon; r++)
    {
        for (int c = 0; c < columna; c++)
        {
            res[r][c] = arr1[r][c] + arr2[r][c];
            printf("%d + %d =%d\t", arr1[r][c], arr2[r][c], res[r][c]);
        }
        printf("\n");
    }
}

void Transpuesta(int arr[renglon][columna])
{
    printf("\n Arreglo original\n");
    for (int r = 0; r < renglon; r++)
    {
        for (int c = 0; c < columna; c++)
        {
            printf("%d\t", arr[r][c]);
        }
        printf("\n");
    }

    printf("\n Transpuesta\n");
    for (int r = 0; r < renglon; r++)
    {
        for (int c = 0; c < columna; c++)
        {
            printf("%d\t", arr[c][r]);
        }
        printf("\n");
    }
}

void ContarNumParesArr(int arr[renglon][columna])
{
    int con = 0;
    int con2 = 0;

    printf("\n Arreglo\n");
    for (int r = 0; r < renglon; r++)
    {
        for (int c = 0; c < columna; c++)

```

```

        {
            printf("%d\t", arr[r][c]);
            if (arr[r][c] % 2 == 0)
            {
                con++;
            }
            else
            {
                con2++;
            }
        }
        printf("\n");
    }
    printf("\n Cantidad de numeros pares: %d\n", con);
    printf("\n Cantidad de numeros impares: %d\n", con2);
}

```

Menu:

```
C:\Users\Okuyt\Desktop\Lenguaje C\Laboratorio\P8>gcc main.c -o main.exe
```

```
C:\Users\Okuyt\Desktop\Lenguaje C\Laboratorio\P8>main.exe
```

Capturar numeros del arreglo.

```
2  8  5
1  10 5
9  9  3
```

Capturar numeros del arreglo.

```
5  6  6
2  8  2
2  6  3
```

-----Menu-----

- 1) Sumar Arreglo posici|n x posici|n
- 2) Transpuesta del arreglo 2
- 3) Contar cuantos numeros pares hay
- 4) Salir

Selecciona una opcion:

Sumar Arreglo posición x posición

```

Selecciona una opcion: 1

                Sumar Arreglo posici|n x posici|n
2 + 5 =7          8 + 6 =14          5 + 6 =11
1 + 2 =3          10 + 8 =18         5 + 2 =7
9 + 2 =11         9 + 6 =15          3 + 3 =6

Presione cualquier tecla para continua|
    
```

Transpuesta del arreglo 2

```

-----Menu-----
1) Sumar Arreglo posici|n x posici|n
2) Transpuesta del arreglo 2
3) Contar cuantos numeros pares hay
4) Salir
Selecciona una opcion: 2

                Transpuesta del arreglo 2

Arreglo original
5      6      6
2      8      2
2      6      3

Transpuesta
5      2      2
6      8      6
6      2      3

Presione cualquier tecla para continua|
    
```


Contar cuantos números pares hay.

```
Selecciona una opcion: 3

                Contar cuantos numeros pares hay
Arreglo
2           8           5
1           10          5
9           9           3

Cantidad de numeros pares: 3

Cantidad de numeros impares: 6

Presione cualquier tecla para continua|
```

Conclusiones

En esta práctica se trabajó con matrices y arreglos en lenguaje C, implementando funciones para rellenar matrices con valores aleatorios, sumar dos matrices, calcular la transpuesta de una matriz y contar números pares e impares. Se utilizó un menú interactivo para que el usuario seleccionara la operación deseada, demostrando el manejo básico de matrices y la modularización del código mediante funciones. Esta práctica reforzó conceptos clave como el uso de bucles, condicionales y la manipulación de arreglos bidimensionales en C.

Bibliografía

Solo se usaron prácticas anteriores. =)