

## **Ingeniero en Software y tecnologías emergentes**

**Materia:** Programación Estructurada / Clave 36276

**Alumno:** lizeth Barrios Retana

**Matrícula:** 372813

**Maestro:** Pedro Núñez Yépiz

**Actividad No. : 9 Código y Capturas de pantalla**

**Tema - Unidad : librerías en c, métodos de ordenación y búsqueda**

**Ensenada, Baja California a 08 de Octubre de 2022.**

## Código principal

```
/*LIZETH BARRIOS RETANA 372813
```

```
2-oct-23
```

```
ACTIVIDAD 9
```

```
    MENÚ
```

- 1.- LLENAR VECTOR
- 2.- LLENAR MATRIZ
- 3.- IMPRIMIR VECTOR
- 4.- IMPRIMIR MATRIZ
- 5.- ORDENAR VECTOR
- 6.- BUSCAR VALOR EN VECTOR
- 0.- SALIR

NOTA: El programa deberá repetirse cuantas veces lo desee el usuario, Validado el menú con la función vali\_num

```
    INSTRUCCIONES:
```

1.- LLENAR VECTOR .- Llenar vector con 15 números, los números generados aleatoriamente, los números entre el rango de 100 al 200 (no repetidos)

2.- LLENAR MATRIZ .- Llenar la matriz de 4x4 con con números generados aleatoriamente, números entre el rango de 1 al 16 (no repetidos)

3.- IMPRIMIR VECTOR .- Imprime el vector que se envíe, donde la función recibe como parámetro el vector,tamaño, nombre del vector.

4.- IMPRIMIR MATRIZ.- Imprime la matriz sin importar el tamaño de la matriz recibiendo como parámetros la matriz, la cantidad de renglones y columnas, así como nombre que se le dará a la matriz.

5.- ORDENAR VECTOR.- Usar función que ordene el vector por el método de ordenación de la Burbuja mejorada.

6.- BUSCAR VALOR EN VECTOR.- Buscar un valor en el vector usando el método de búsqueda secuencial.

```
    0.- SALIR
```

```
    BRL_act9_1_932
```

```
*/
```

```
#include <stdio.h>
```

```
#include <time.h>
```

```
#include <stdlib.h>
```

```
#include "Lizeth.h"
```

```
int vec[15];
```

```
int mat[4][4];
```

```

int main()
{
    srand(time(NULL));
    int menu;
    int opc, n;
    do
    {
        menu = 0;
        opc = validarNumeros("MENU\n1.- LLENAR VECTOR\n2.- LLENAR
MATRIZ\n3.- IMPRIMIR VECTOR\n4.- IMPRIMIR MATRIZ\n5.- ORDENAR
VECTOR\n6.- BUSCAR VALOR EN VECTOR\n0.- SALIR\n", 0, 6);
        switch (opc)
        {
            case 0:
                system("cls");
                printf("FIN DEL PROGRAMA");
                break;
            case 1:
                VectSinRep(vec, 15, 100, 200);
                menu = repetir();
                break;
            case 2:
                matriz4x4(mat, 1, 16);
                menu = repetir();
                break;
            case 3:
                imprimirVec("==vector==", vec, 15);
                menu = repetir();
                break;
            case 4:
                imprimirMatriz(mat, 4, 4, "==Matriz==");
                menu = repetir();
                break;
            case 5:
                ordenarVec(vec, 15);
                menu = repetir();
                break;
            case 6:
                buscarValor(vec, 15);
                menu = repetir();
                break;
        }
    }
}

```

```

    } while (menu == 1);
}

```

### Libreria 'Lizeth.h'

```

// (Funciones de validar números, método de búsqueda)
#include <stdio.h>
#include <stdlib.h>
#include <time.h>
#include <stdbool.h>

int validarNumeros(char mensaje[], int ri, int rf);
int Buscar(int vect[], int n, int num);
int repetir();
int VectSinRep(int vec[], int n, int ri, int rf);
int random(int ri, int rf);
int imprimirVec(char msg[], int vec[], int n);
void matriz4x4(int matriz[4][4], int ri, int rf);
void imprimirMatriz(int matriz[][4], int m, int n, char mensaje[]);
void ordenarVec(int vec[], int n);
void buscarValor();

void buscarValor(int vec[], int n)
{
    int i, encontrado = 0, numero;
    numero = validarNumeros("INGRESE EL NUMERO: ", 0, 10000);
    for (i = 0; i < n; i++)
    {
        if (vec[i] == numero)
        {
            printf("El numero %d se encuentra en el lugar [%d]\n",
numero, i);
            encontrado = 1;
        }
    }
    if (encontrado == 0)
    {
        printf("El numero no se encuentra en el vector\n");
    }
}

// para ordenar el vector
void ordenarVec(int vec[], int n)
{

```

```

int i, j, temp;
int ordenado = 0;

for (i = 0; i < n - 1; i++)
{
    ordenado = 1;
    for (j = 0; j < n - i - 1; j++)
    {
        if (vec[j] > vec[j + 1])
        {
            temp = vec[j];
            vec[j] = vec[j + 1];
            vec[j + 1] = temp;
            ordenado = 0;
        }
    }
}

// para imprimir la matriz
void imprimirMatriz(int matriz[][4], int m, int n, char mensaje[])
{
    int i, j;
    printf("%s:\n", mensaje);
    for (i = 0; i < m; i++)
    {
        for (j = 0; j < n; j++)
        {
            printf("[%3d] ", matriz[i][j]);
        }
        printf("\n");
    }
}

// para llenar la matriz
void matriz4x4(int matriz[4][4], int ri, int rf)
{
    srand(time(NULL));
    int i, j, repe, vec[16];
    for (i = 0; i < 16; i++)
    {
        do
        {

```

```

        vec[i] = random(ri, rf);
        repe = 0;
        for (j = 0; j < i; j++)
        {
            if (vec[i] == vec[j])
            {
                repe = 1;
            }
        }
    } while (repe == 1);
}
for (i = 0; i < 4; i++)
{
    for (j = 0; j < 4; j++)
    {
        matriz[i][j] = vec[i * 4 + j];
    }
}
}

// valida la entrada de numeros
int validarNumeros(char mensaje[], int ri, int rf)
{
    char entrada[100];
    int num;
    do
    {
        printf("%s", mensaje);
        fgets(entrada, sizeof(entrada), stdin);
        num = atoi(entrada);
        if (num < ri || num > rf)
        {
            printf("INGRESE EL NUMERO OTRA VEZ PORFAVOR\n");
        }
    } while (num < ri || num > rf);
    return num;
}

// para repetir hasta que el usuario decida salir
int repetir()
{
    int repetir;

```

```

        repetir = validarNumeros("QUIERES HACER OTRA OPCION\nSI=1\nNO=2\n",
1, 2);
        if (repetir == 1)
        {
            return 1;
        }
        else
        {
            printf("FIN DEL POGRAMA");
            return 0;
        }
    }

// busca los numeros repetidos en vector
int Buscar(int vect[], int n, int num)
{
    int i;
    for (i = 0; i < n; i++)
    {
        if (vect[i] == num)
        {
            return i;
        }
    }
    return -1;
}

// buscar los numeros repetidos en matriz

// llenar vector sin repetir num
int VectSinRep(int vec[], int n, int ri, int rf)
{
    int i, num;
    for (i = 0; i < n; i++)
    {
        do
        {
            num = random(ri, rf);
        } while (Buscar(vec, i, num) != -1);
        vec[i] = num;
    }
}

```

```

// funcion para hacer los numeros aleatorios
int random(int ri, int rf)
{
    int r;
    r = (rf - ri) + 1;
    return (rand() % r) + ri;
}

// funcion para imprimir el vector
int imprimirVec(char msg[], int vec[], int n)
{
    int i;
    printf("%s\n", msg);
    for (i = 0; i < n; i++)
    {
        printf("[%1d]\n", vec[i]);
    }
}

```

## Ejemplos de capturas de pantalla

<p>MENU</p> <p>1.- LLENAR VECTOR</p> <p>2.- LLENAR MATRIZ</p> <p>3.- IMPRIMIR VECTOR</p> <p>4.- IMPRIMIR MATRIZ</p> <p>5.- ORDENAR VECTOR</p> <p>6.- BUSCAR VALOR EN VECTOR</p> <p>0.- SALIR</p> <p>1</p> <p>QUIERES HACER OTRA OPCION</p> <p>SI=1</p> <p>NO=2</p> <p>1</p> <p>MENU</p> <p>1.- LLENAR VECTOR</p> <p>2.- LLENAR MATRIZ</p> <p>3.- IMPRIMIR VECTOR</p> <p>4.- IMPRIMIR MATRIZ</p> <p>5.- ORDENAR VECTOR</p> <p>6.- BUSCAR VALOR EN VECTOR</p> <p>0.- SALIR</p> <p>3</p> <p>==vector==</p> <p>[144]</p> <p>[109]</p> <p>[128]</p> <p>[112]</p> <p>[165]</p> <p>[148]</p> <p>[186]</p> <p>[197]</p> <p>[152]</p> <p>[130]</p> <p>[167]</p> <p>[172]</p> <p>[115]</p> <p>[159]</p> <p>[139]</p> <p>QUIERES HACER OTRA OPCION</p> <p>SI=1</p> <p>NO=2</p>	<p>MENU</p> <p>1.- LLENAR VECTOR</p> <p>2.- LLENAR MATRIZ</p> <p>3.- IMPRIMIR VECTOR</p> <p>4.- IMPRIMIR MATRIZ</p> <p>5.- ORDENAR VECTOR</p> <p>6.- BUSCAR VALOR EN VECTOR</p> <p>0.- SALIR</p> <p>2</p> <p>QUIERES HACER OTRA OPCION</p> <p>SI=1</p> <p>NO=2</p> <p>1</p> <p>MENU</p> <p>1.- LLENAR VECTOR</p> <p>2.- LLENAR MATRIZ</p> <p>3.- IMPRIMIR VECTOR</p> <p>4.- IMPRIMIR MATRIZ</p> <p>5.- ORDENAR VECTOR</p> <p>6.- BUSCAR VALOR EN VECTOR</p> <p>0.- SALIR</p> <p>4</p> <p>==Matriz==:</p> <p>[ 2] [ 4] [ 9] [ 12]</p> <p>[ 10] [ 14] [ 11] [ 7]</p> <p>[ 8] [ 3] [ 16] [ 1]</p> <p>[ 15] [ 13] [ 6] [ 5]</p> <p>QUIERES HACER OTRA OPCION</p> <p>SI=1</p> <p>NO=2</p>	<p>MENU</p> <p>1.- LLENAR VECTOR</p> <p>2.- LLENAR MATRIZ</p> <p>3.- IMPRIMIR VECTOR</p> <p>4.- IMPRIMIR MATRIZ</p> <p>5.- ORDENAR VECTOR</p> <p>6.- BUSCAR VALOR EN VECTOR</p> <p>0.- SALIR</p> <p>3</p> <p>==vector==</p> <p>[109]</p> <p>[112]</p> <p>[115]</p> <p>[128]</p> <p>[130]</p> <p>[139]</p> <p>[144]</p> <p>[148]</p> <p>[152]</p> <p>[159]</p> <p>[165]</p> <p>[167]</p> <p>[172]</p> <p>[186]</p> <p>[197]</p> <p>QUIERES HACER OTRA OPCION</p> <p>SI=1</p> <p>NO=2</p>	<p>MENU</p> <p>1.- LLENAR VECTOR</p> <p>2.- LLENAR MATRIZ</p> <p>3.- IMPRIMIR VECTOR</p> <p>4.- IMPRIMIR MATRIZ</p> <p>5.- ORDENAR VECTOR</p> <p>6.- BUSCAR VALOR EN VECTOR</p> <p>0.- SALIR</p> <p>6</p> <p>INGRESE EL NUMERO: 172</p> <p>El numero 172 se encuentra en el lugar [12]</p> <p>QUIERES HACER OTRA OPCION</p> <p>SI=1</p> <p>NO=2</p> <p>1</p> <p>MENU</p> <p>1.- LLENAR VECTOR</p> <p>2.- LLENAR MATRIZ</p> <p>3.- IMPRIMIR VECTOR</p> <p>4.- IMPRIMIR MATRIZ</p> <p>5.- ORDENAR VECTOR</p> <p>6.- BUSCAR VALOR EN VECTOR</p> <p>0.- SALIR</p> <p>6</p> <p>INGRESE EL NUMERO: 162</p> <p>El numero no se encuentra en el vector</p> <p>QUIERES HACER OTRA OPCION</p> <p>SI=1</p> <p>NO=2</p>
---	--	---	---