



Universidad Autónoma de Baja California

Facultad de Ingeniería, Arquitectura y Diseño

Ingeniero en Computación

Asignatura:

Programación Estructurada

Actividad 9:

FUNCIONES Y METODOS DE ORDENACION Y BUSQUEDA

Brayan Arturo Rocha Meneses

Matricula:

371049

Ensenada Baja California 10 de Octubre del 2023

```
include <stdio.h>
include <stdlib.h>
include "4life.h"

nt msges();
oid menu();

nt BuscarVector(int vector1[], int n);

nt main()

    menu();
    return 0;

nt msges()

    int op;
    system ("CLS");
    printf ("\n MENU \n");
    printf("1.- LLENAR EL VECTOR \n");
    printf("2.- LLENAR LA MATRIZ \n");
    printf("3.- IMPRIMIR EL VECTOR \n");
    printf("4.- IMPRIMIR LA MATRIZ \n");
```

```
nt msges()

    int op;
    system ("CLS");
    printf ("\n MENU \n");
    printf("1.- LLENAR EL VECTOR \n");
    printf("2.- LLENAR LA MATRIZ \n");
    printf("3.- IMPRIMIR EL VECTOR \n");
    printf("4.- IMPRIMIR LA MATRIZ \n");
    printf("5.- ORDENAR EL VECTOR \n");
    printf("6.- BUSCAR VALOR EN EL VECTOR \n");
    printf("7.- SALIR \n");

    op = ValidarCadena("Escribe el numero de opciones que seleccionaste: \n", 0, 6);

    return op;

oid menu()

    int op;
    int n = 15;
    int vector1[15];
    int matriz[4][4];
```

```
do{
    system("CLS");
    op=msges();

    switch (op)
    {
    case 1:
        LlenarVectorSinRepetir(vector1, n, 100, 200);
        printf("Vector a sido llenado.\n");
        EsperarUsuario();
        break;

    case 2:
        LlenarMatrizSinRepetir(4, 4, matriz, 1, 16);
        printf("Matriz a sido llenada.\n");
        EsperarUsuario();
        break;

    case 3:
        ImprimirVector(vector1, n);
        EsperarUsuario();
        break;

    case 4:
        ImprimirMatriz(4, 4, matriz);
        EsperarUsuario();
        break;

    case 5:
        OrdenarVector(vector1, n);
        printf("Vector ordenado, vuelva a seleccionar la opcion numero 3.\n");
        EsperarUsuario();
    }
```

```

        break;
    case 5:
        OrdenarVector(vector1, n);
        printf("Vector ordenado, vuelva a seleccionar la opcion numero 3.\n");
        EsperarUsuario();
        break;
    case 6:
        BuscarVector(vector1, n);
        EsperarUsuario();
        break;
    case 0:
        printf("Leave the program.\n");
        break;
    }
} while (op != 0);
}

int BuscarVector(int vector1[], int n)
{
    int ValorBuscar, posicion;

    ValorBuscar = ValidarCadena("Ingresa el valor que usted busca (Entre 100 y 200): ", 100, 200);

    posicion = BusqSecVector(vector1, n, ValorBuscar);

    if (posicion != -1)

```

```

    }
} while (op != 0);
}

int BuscarVector(int vector1[], int n)
{
    int ValorBuscar, posicion;

    ValorBuscar = ValidarCadena("Ingresa el valor que usted busca (Entre 100 y 200): ", 100, 200);

    posicion = BusqSecVector(vector1, n, ValorBuscar);

    if (posicion != -1)
    {
        printf("El numero se encontro, esta en la posicion: %d\n", posicion);
    }
    else
    {
        printf("El numero proporcionado no esta en el vector\n");
    }

    return posicion;
}

```