



UNIVERSIDAD AUTÓNOMA DE QUERÉTARO  
Facultad de informática



## Tarea: Programa listas dinámicas.

### ESTRUCTURA DE DATOS

CARLOS NOGUEZ JUAREZ

315398

GRUPO 35

FECHA 09/04/2024



Codigo:

```
#include <iostream>
#include <cstdlib>

using namespace std;

struct Nodo
{
    int data;
    Nodo *next;
};

void insertList(Nodo *&, int);
void mostrarLista(Nodo *);
void buscarLista(Nodo *, int);
bool existeEnLista(Nodo *, int);

Nodo *head = NULL;

int main() {
    system("cls");
    int opcion, n;

    do {
        cout << "*****" << endl;
        cout << "Menu de opciones:" << endl;
        cout << "1. Insertar elemento" << endl;
        cout << "2. Mostrar lista" << endl;
        cout << "3. Buscar elemento" << endl;
        cout << "4. Salir" << endl;
        cout << "Ingrese una opcion: ";
        cin >> opcion;

        switch (opcion) {
            case 1:
                cout << "Ingrese el contenido de data: ";
                cin >> n;
                if (existeEnLista(head, n)) {
```



```
        cout << "El elemento " << n << " ya esta en la lista. No se puede agregar nuevamente." << endl;
    } else {
        insertList(head, n);
        cout << "Elemento insertado correctamente." << endl;
    }
    break;
case 2:
    mostrarLista(head);
    break;
case 3:
    cout << "Ingresa el elemento a buscar: ";
    cin >> n;
    buscarLista(head, n);
    break;
case 4:
    cout << "Saliendo del programa..." << endl;
    break;
default:
    cout << "Opción invalida. Intente nuevamente." << endl;
}
} while (opcion != 4);

return 0;
}

void mostrarLista(Nodo *head) {
    Nodo *actual = head;
    if (actual == NULL) {
        cout << "La lista está vacia." << endl;
    } else {
        cout << "Los numeros ingresados son: ";
        while (actual != NULL) {
            cout << "|" << actual->data << "| -> ";
            actual = actual->next;
        }
        cout << "NULL" << endl;
    }
}

void insertList(Nodo *&head, int n) {
    Nodo *new_nodo = new Nodo();
    new_nodo->data = n;
    Nodo *aux1 = head;
    Nodo *aux2 = NULL;
```



```
while ((aux1 != NULL) && (aux1->data < n)) {
    aux2 = aux1;
    aux1 = aux1->next;
}

if (head == aux1) {
    head = new_nodo;
} else {
    aux2->next = new_nodo;
}

new_nodo->next = aux1;
}

void buscarLista(Nodo *head, int n) {
    bool flag = false;
    Nodo *actual = new Nodo();
    Nodo *address;
    actual = head;

    while ((actual != NULL) && (actual->data <= n)) {
        if (actual->data == n) {
            flag = true;
            address = actual;
        }
        actual = actual->next;
    }

    if (flag == true) {
        cout << "El elemento " << n << " ha sido encontrado en la posicion " << address << endl;
    } else {
        cout << "El elemento " << n << " no ha sido encontrado en la lista." << endl;
    }
}

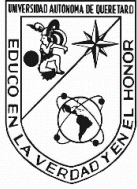
bool existeEnLista(Nodo *head, int n) {
    Nodo *actual = head;
    while (actual != NULL) {
        if (actual->data == n) {
            return true;
        }
        actual = actual->next;
    }
    return false;
}
```



Resultado:

```
Menu de opciones:
1. Insertar elemento
2. Mostrar lista
3. Buscar elemento
4. Salir
Ingrese una opcion: 1
Ingrese el contenido de data: 4
Elemento insertado correctamente.
*****
Menu de opciones:
1. Insertar elemento
2. Mostrar lista
3. Buscar elemento
4. Salir
Ingrese una opcion: 1
Ingrese el contenido de data: 5
Elemento insertado correctamente.
*****
Menu de opciones:
1. Insertar elemento
2. Mostrar lista
3. Buscar elemento
4. Salir
Ingrese una opcion: 1
Ingrese el contenido de data: 6
Elemento insertado correctamente.
```

```
Menu de opciones:
1. Insertar elemento
2. Mostrar lista
3. Buscar elemento
4. Salir
Ingrese una opcion: 2
Los numeros ingresados son: |4| -> |5| -> |6| -> NULL
```



Menu de opciones:

1. Insertar elemento
2. Mostrar lista
3. Buscar elemento
4. Salir

Ingrese una opcion: 3

Ingresa el elemento a buscar: 2

El elemento 2 no ha sido encontrado en la lista.

\*\*\*\*\*

Menu de opciones:

1. Insertar elemento
2. Mostrar lista
3. Buscar elemento
4. Salir

Ingrese una opcion: 3

Ingresa el elemento a buscar: 5

El elemento 5 ha sido encontrado en la posicion 0x23a5b6a14d0