

ED_JuanPabloCostas_MenuPilasDinamicas



- Universidad Autónoma De Querétaro
- Facultad de informática
- Estructura de datos
- M.I.S.D. Erika del Río Magaña

- Costas Rueda Juan Pablo
- EXP: 307081
- 4to Semestre

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

using namespace std;

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

void agregarCola(Nodo *&, Nodo *&, int);

void imprimirCola(Nodo *, Nodo *);

void vaciar(Nodo *&);

void eliminar(Nodo *&);
```

```
void buscar(Nodo *, int);

Nodo *head = NULL;
Nodo *tail = NULL;

int main(){

    bool loop = true;
    do
    {
        int n, o;

        cout << "\n\nSeleccione una opcion: \n";
        cout << "1- Insertar un valor\n";
        cout << "2- Mostrar la cola\n";
        cout << "3- Vaciar cola\n";
        cout << "4- Eliminar Nodo\n";
        cout << "5- Buscar Nodo\n";
        cout << "> ";
        cin >> o;
        switch (o)
        {
            case 1:
                cout << "Ingresa un valor entero\n> ";
                cin >> n;
                agregarCola(head, tail, n);
                break;

            case 2:
                cout << "Datos de la lista: \n";
                imprimirCola(head,tail);
                break;

            case 3:
                vaciar(head);
                break;

            case 4:
                eliminar(head);
                break;

            case 5:
                cout << "Ingresa el valor a buscar\n> ";
                cin >> n;
                buscar(head,n);
                break;

            default:
                loop = false;
                break;
        }
    } while (loop);
```

```
}

void agregarCola(Nodo *& head, Nodo *& tail, int n){
    Nodo *new_nodo = new Nodo();
    new_nodo -> data = n;
    new_nodo -> next = NULL;
    if (head == NULL)
    {
        head = new_nodo;
    }
    else
    {
        tail -> next = new_nodo;
    }
    tail = new_nodo;
    cout << "Dato Ingresado\n";
}

void imprimirCola(Nodo *head, Nodo *tail){
    Nodo *actual = new Nodo();
    actual = head;
    cout << "HEAD ";
    while (actual != NULL)
    {
        cout << "|" << actual -> data << "|" << " -> ";
        actual = actual -> next;
    }
    cout << "TAIL";
}

void vaciar(Nodo *& head){
    while (head != NULL)
    {
        Nodo * aBorrar = new Nodo();
        aBorrar = head;
        head = head -> next;
        delete aBorrar;
    }
    cout << "Cola vacia\n";
}

void eliminar(Nodo *& head){
    if (head != NULL)
    {
        Nodo * aBorrar = new Nodo();
        aBorrar = head;
        head = head -> next;
        cout << "|" << aBorrar -> data << "|" << " eliminado\n";
        delete aBorrar;
    }
    else
    {

```

```

        cout << "La cola esta vacia\n";
    }
}

void buscar(Nodo * head, int n){
    bool loop = true;
    while(head != NULL && loop){
        if (head -> data == n)
        {
            cout << head -> data << " encontrado en " << &head -> data << "\n";
            loop = false;
        }
        head = head -> next;
    }
    if (loop)
    {
        cout << "Dato no encontrado";
    }
}

```

```

Seleccione una opcion:
1- Insertar un valor
2- Mostrar la cola
3- Vaciar cola
4- Eliminar Nodo
5- Buscar Nodo
> 2
Datos de la lista:
HEAD |5| -> |7| -> TAIL

```

```

Seleccione una opcion:
1- Insertar un valor
2- Mostrar la cola
3- Vaciar cola
4- Eliminar Nodo
5- Buscar Nodo
> 5
Ingresa el valor a buscar
> 5
5 encontrado en 0x761960

```

```

Seleccione una opcion:
1- Insertar un valor
2- Mostrar la cola
3- Vaciar cola
4- Eliminar Nodo
5- Buscar Nodo
> 5
Ingresa el valor a buscar
> 2
dato no encontrado

```