



INSTITUTO  
**KHIPU**

**Semestre III**

**Sesión 14**

**CARRERA PROFESIONAL**

# **DESARROLLO DE SISTEMAS DE INFORMACIÓN**

**HERRAMIENTAS DE  
PROGRAMACION C#**

**Tema**

**IMPLEMENTACION DE UNA LISTA  
ENLAZADA**

## METODOS COMUNES EN LAS LISTAS ENLAZADAS

**1. Insertar al Inicio (push\_front)**

Este método agrega un nuevo nodo al principio de la lista. Es una operación eficiente, ya que simplemente se ajusta la referencia de la cabeza al nuevo nodo.

```
void insertarInicio(int valor) {  
    Nodo nuevoNodo = new Nodo(valor);  
    nuevoNodo.siguiente = cabeza;  
    cabeza = nuevoNodo;  
}
```

**2. Insertar al Final (push\_back)**

Agrega un nuevo nodo al final de la lista, recorriendo todos los nodos hasta encontrar el último y luego enlazando el nuevo nodo al final.

```
void insertarFinal(int valor) {  
    Nodo nuevoNodo = new Nodo(valor);  
    si (cabeza == null) {  
        cabeza = nuevoNodo;  
    } sino {  
        Nodo actual = cabeza;  
        mientras (actual.siguiente != null) {  
            actual = actual.siguiente;  
        }  
        actual.siguiente = nuevoNodo;  
    }  
}
```

### 3. Eliminar el Primer Nodo (pop\_front)

Elimina el primer nodo de la lista, ajustando la referencia de la cabeza al siguiente nodo.

```
void eliminarInicio() {  
    si (cabeza != null) {  
        cabeza = cabeza.siguiente;  
    }  
}
```

### 4. Buscar un Nodo (search)

Recorre la lista nodo por nodo buscando un valor específico.

```
bool buscar(int valor) {  
    Nodo actual = cabeza;  
    mientras (actual != null) {  
        si (actual.dato == valor) {  
            return true;  
        }  
        actual = actual.siguiente;  
    }  
    return false;  
}
```

### 5. Invertir la Lista (reverse)

Cambia el orden de los nodos de la lista, invirtiendo los enlaces.

```
void invertir() {  
    Nodo anterior = null;  
    Nodo actual = cabeza;  
    Nodo siguiente = null;  
    mientras (actual != null) {  
        siguiente = actual.siguiente;  
        actual.siguiente = anterior;  
        anterior = actual;  
        actual = siguiente;  
    }  
    cabeza = anterior;  
}
```

### IMPLEMENTACIÓN DE UNA LISTA ENLAZADA SIMPLE EN C#

#### 1. CREAR LA CLASE “Nodo”

```
class Nodo {  
    public int dato;  
    public Nodo siguiente;  
  
    public Nodo(int valor) {  
        dato = valor;  
        siguiente = null;  
    }  
}
```

## 2. CREAR LA CLASE “ListaEnlazada”

```
class ListaEnlazada {
    private Nodo cabeza;

    public ListaEnlazada() {
        cabeza = null;
    }

    public void insertarInicio(int valor) {
        Nodo nuevoNodo = new Nodo(valor);
        nuevoNodo.siguiente = cabeza;
        cabeza = nuevoNodo;
    }

    public void insertarFinal(int valor) {
        Nodo nuevoNodo = new Nodo(valor);
        if (cabeza == null) {
            cabeza = nuevoNodo;
        } else {
            Nodo actual = cabeza;
            while (actual.siguiente != null) {
                actual = actual.siguiente;
            }
            actual.siguiente = nuevoNodo;
        }
    }

    public bool buscar(int valor) {
        Nodo actual = cabeza;
        while (actual != null) {
            if (actual.dato == valor) {
                return true;
            }
            actual = actual.siguiente;
        }
        return false;
    }

    public void imprimirLista() {
        Nodo actual = cabeza;
        while (actual != null) {
            Console.WriteLine(actual.dato);
            actual = actual.siguiente;
        }
    }
}
```

### 3. Usar

```
ListaEnlazada lista = new ListaEnlazada();  
lista.insertarInicio(5);  
lista.insertarFinal(10);  
lista.imprimirLista(); // Salida: 5, 10
```





INSTITUTO  
**KHIPU**