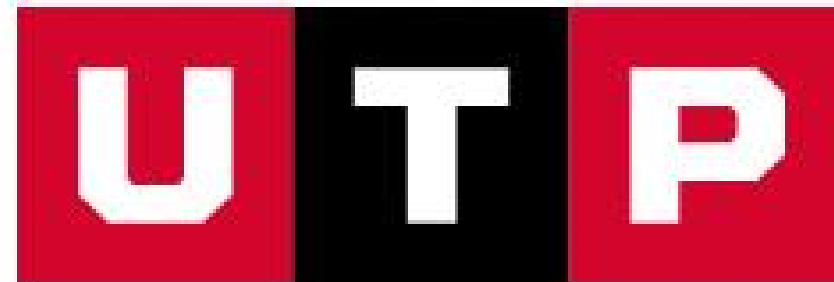


# **ALGORITMOS Y ESTRUCTURA DE DATOS**

**Ing. Frank Huamanchumo Casanova.**



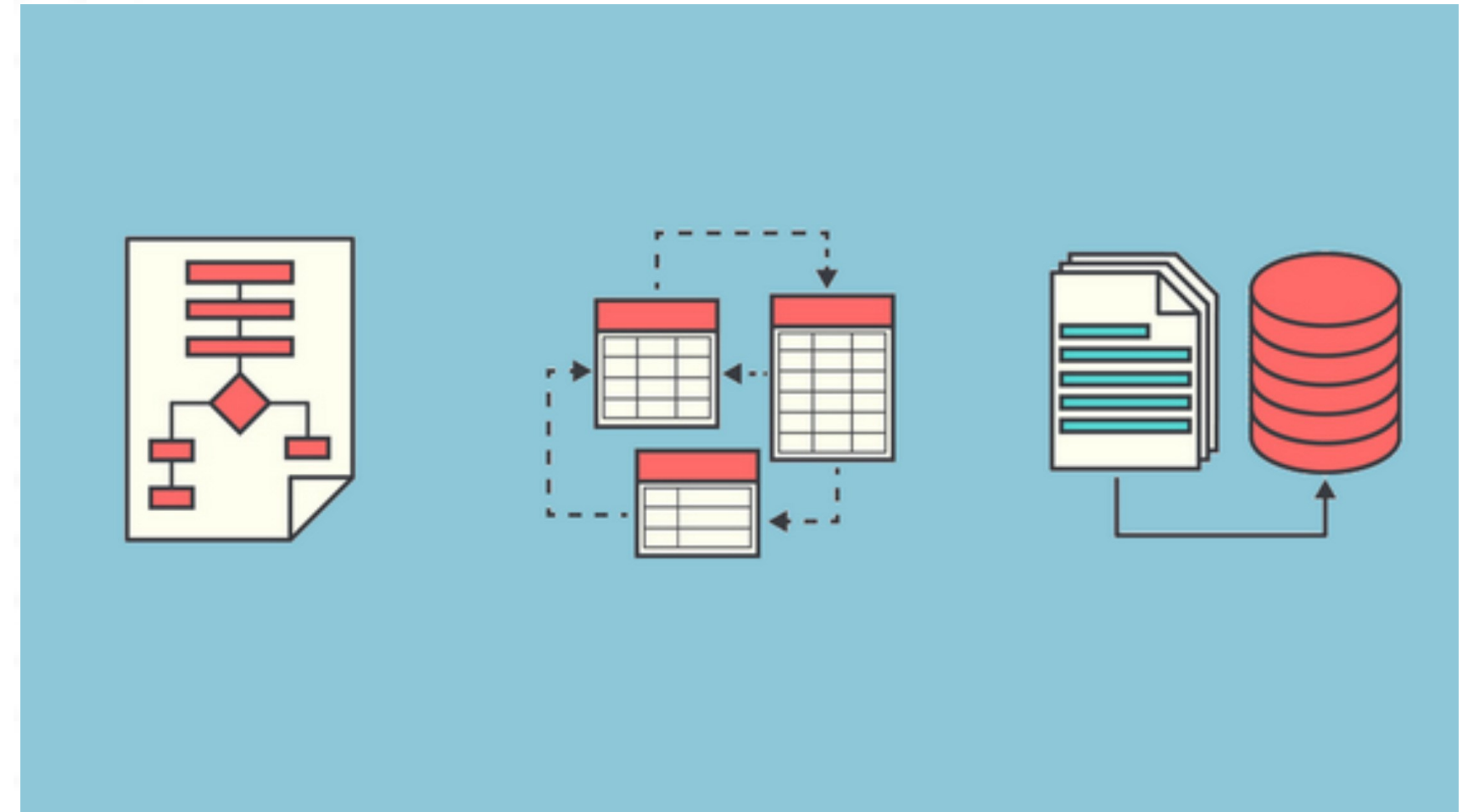
**Universidad  
Tecnológica  
del Perú**

# ¿Alguna consulta de la clase anterior?



# Recordemos

- Estructuras dinámicas
- Listas enlazadas simples
- Operaciones básicas con listas enlazadas



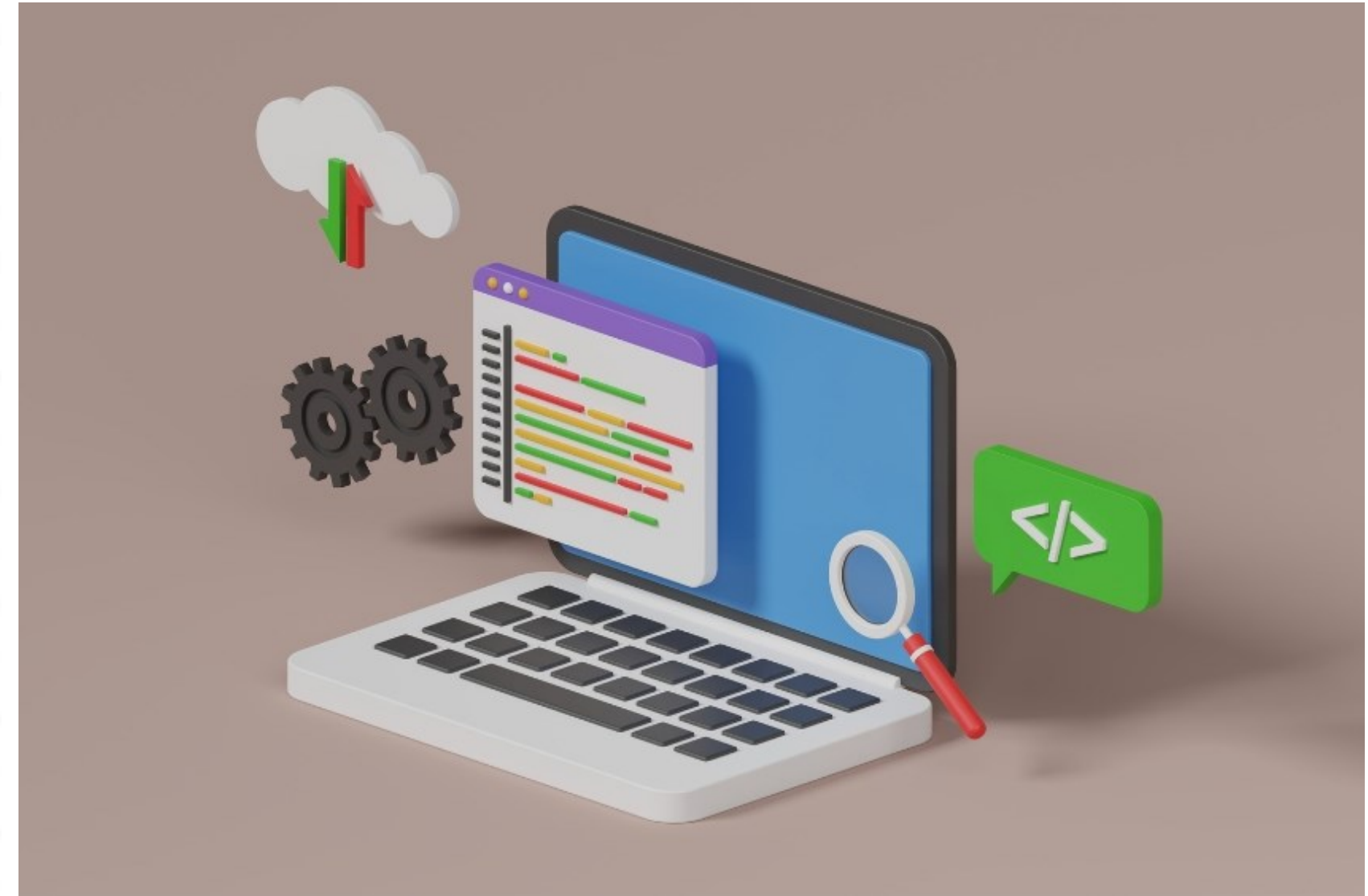
# Utilidad

- En qué situaciones debo almacenar datos en listas enlazadas



# Logro de la sesión

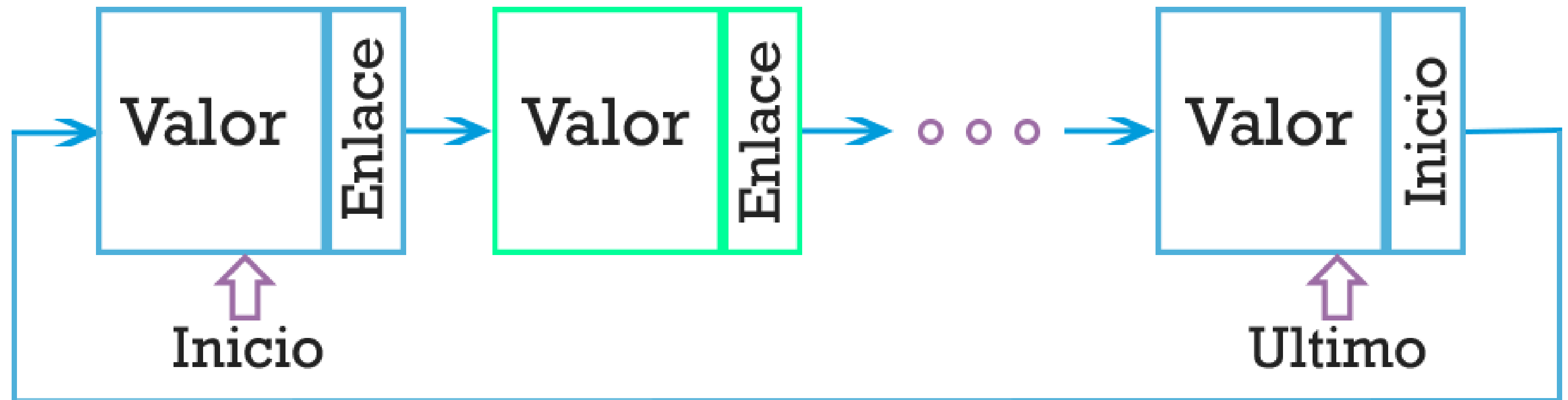
Al finalizar la sesión, el alumno es capaz de entender los conceptos de listas doblemente enlazadas y sus operaciones básicas en ejercicios de programación orientada a objetos





# Recordemos... Listas enlazadas simples

- Estructura
- Operaciones Básicas
- Funcionamiento



# Lista doblemente enlazada

A diferencia de las listas enlazadas simples, donde cada nodo tiene un enlace al siguiente nodo, en una lista doblemente enlazada cada nodo tiene dos enlaces: uno al nodo anterior y otro al nodo siguiente.

Esto permite un recorrido bidireccional por la lista, es decir, se puede navegar tanto hacia adelante como hacia atrás.

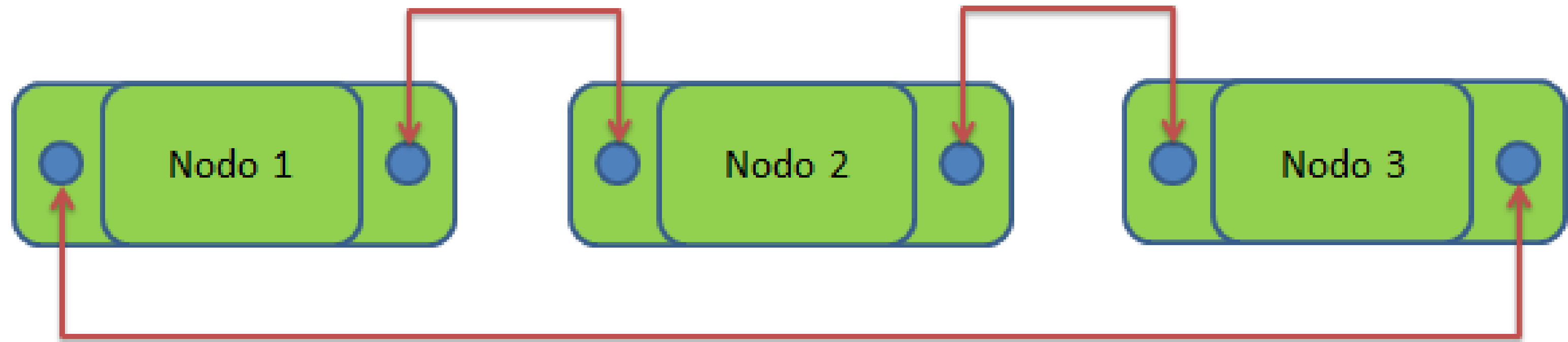
# Características de una lista doblemente enlazada Universidad Tecnológica del Perú

- 1. Nodo:** Cada nodo de una lista doblemente enlazada contiene tres partes:
  - **Dato:** El valor almacenado en el nodo.
  - **Enlace anterior:** Un puntero o referencia al nodo anterior en la lista.
  - **Enlace siguiente:** Un puntero o referencia al nodo siguiente en la lista.
- 2. Cabeza (Head):** Es el primer nodo de la lista, cuya referencia anterior es null.
- 3. Cola (Tail):** Es el último nodo de la lista, cuya referencia siguiente es null



# Estructura

- Nodo
- Enlace anterior
- Enlace siguiente



Lista doblemente logada circular – Podemos recorrer la estructura en ambas direcciones

# Operaciones básicas en una lista doblemente enlazada:

- Inserción
- Eliminación
- Recorrido
- Búsqueda



# Operaciones básicas en una lista doblemente enlazada:

## Inserción

- **Al principio:** Insertar un nuevo nodo al inicio de la lista.
- **Al final:** Insertar un nuevo nodo al final de la lista.
- **En medio:** Insertar un nuevo nodo en una posición intermedia.

# Operaciones básicas en una lista doblemente enlazada:

## Eliminación

- **Del principio:** Eliminar el primer nodo de la lista.
- **Del final:** Eliminar el último nodo de la lista.
- **De una posición específica:** Eliminar un nodo de una posición intermedia.

# Operaciones básicas en una lista doblemente enlazada:

## Recorrido

- **Hacia adelante:** Navegar desde el primer nodo hasta el último.
- **Hacia atrás:** Navegar desde el último nodo hasta el primero.

# Operaciones básicas en una lista doblemente enlazada:

## Búsqueda

- Buscar un elemento específico dentro de la lista.



# Vamos a programar



# Clase Nodo

```
public class Nodo {  
    int dato;  
    Nodo anterior;  
    Nodo siguiente;  
  
    public Nodo(int dato) {  
        this.dato = dato;  
        this.anterior = null;  
        this.siguiente = null;  
    }  
}
```

```
public class ListaDoblemente {
    Nodo cabeza;
    Nodo cola;

    public ListaDoblemente() {
        cabeza = null;
        cola = null;
    }
    public void insertarAlFinal(int dato) {
        Nodo nuevoNodo = new Nodo(dato);
        if (cabeza == null) {
            cabeza = cola = nuevoNodo;
        } else {
            cola.siguiente = nuevoNodo;
            nuevoNodo.anterior = cola;
            cola = nuevoNodo;
        }
    }
    public void recorrerHaciaAdelante() {
        Nodo actual = cabeza;
        while (actual != null) {
            System.out.print(actual.dato + " ");
            actual = actual.siguiente;
        }
        System.out.println();
    }
}
```

# Clase Doblemente Enlazada

# Clase Main

```
public class Ed091 {  
    public static void main(String[] args) {  
        ListaDoblemente lista = new ListaDoblemente();  
        lista.insertarAlFinal(1);  
        lista.insertarAlFinal(2);  
        lista.insertarAlFinal(3);  
  
        System.out.println("Recorrido hacia adelante:");  
        lista.recorrerHaciaAdelante();  
    }  
}
```

# Práctica

**¿Qué distingue a una lista doblemente enlazada de una lista simplemente enlazada?**

- a) Cada nodo tiene una referencia al nodo siguiente.
- b) Cada nodo tiene referencias al nodo anterior y al siguiente.
- c) No se pueden insertar nodos en una lista doblemente enlazada.
- d) Los nodos solo pueden ser recorridos en una dirección.

# Práctica

**¿Cuál es la ventaja principal de una lista doblemente enlazada?**

- a) Permite la búsqueda binaria.
- b) Permite recorrer la lista en ambas direcciones (hacia adelante y hacia atrás).
- c) Ocupa menos memoria que una lista simplemente enlazada.
- d) Facilita el uso de índices para acceder a los elementos.



# Práctica

**En una lista doblemente enlazada, si el nodo nuevoNodo se inserta al final de la lista, ¿qué se debe hacer con la referencia siguiente del nodo que era la cola?**

- a) Debe apuntar al nodo cabeza.
- b) Debe apuntar a null.
- c) Debe apuntar al nuevoNodo.
- d) No se modifica..

# Práctica

**¿Cuál de las siguientes operaciones NO es una operación básica en una lista doblemente enlazada?**

- A) Inserción de un nodo
- B) Eliminación de un nodo.
- C) Ordenación de los nodos.
- D) Recorrido de la lista hacia adelante y hacia atrás.

# Práctica

**Si una lista doblemente enlazada tiene tres nodos, y se elimina el nodo del medio, ¿qué sucede con las referencias de los nodos restantes?**

- A) No cambia nada, las referencias permanecen igual.
- B) El primer nodo ahora apunta directamente al último nodo, y el último nodo apunta al primero.
- C) La lista queda rota y no se puede seguir utilizando.
- D) El primer nodo apunta al último nodo mediante su referencia siguiente, y el último nodo apunta al primero mediante su referencia anterior.

# ¿Tienen alguna consulta o duda?



# Cierre

¿Qué hemos aprendido hoy?



- Características de una lista doblemente enlazada
- Estructura o Partes
- Operaciones básicas



**Universidad  
Tecnológica  
del Perú**