

**Listas Doblemente Enlazadas**

**Jasser Lozano**

**Estructuras de Información**

**402M**

**Ingeniería de Sistemas y Computación**

**Facultad de Ingeniería**

**Universidad de Cundinamarca**

**Octubre, 2025**

## **Contenido**

Introducción .....	2
Resumen y Desarrollo del Algoritmo .....	3
Punto 1: Algoritmo de lista de números .....	3
Punto 2: Algoritmo de Transmilenio .....	4
Punto 3: Algoritmo de lista de números con mis funciones propias .....	5
Resultados .....	7

## **Introducción**

En el siguiente programa se implementa una lista doblemente enlazada utilizando el lenguaje java. el objetivo es mostrar de forma clara y funcional cómo se pueden realizar operaciones básicas como añadir, eliminar e imprimir elementos tanto en orden normal como inverso. este ejercicio busca reforzar la comprensión de las estructuras dinámicas de datos, reemplazando el uso de funciones nativas por métodos creados desde cero, para entender mejor cómo se enlazan y recorren los nodos dentro de una lista.

## Resumen y Desarrollo del Algoritmo

### Punto 1: Algoritmo de lista de números

Figura 1.

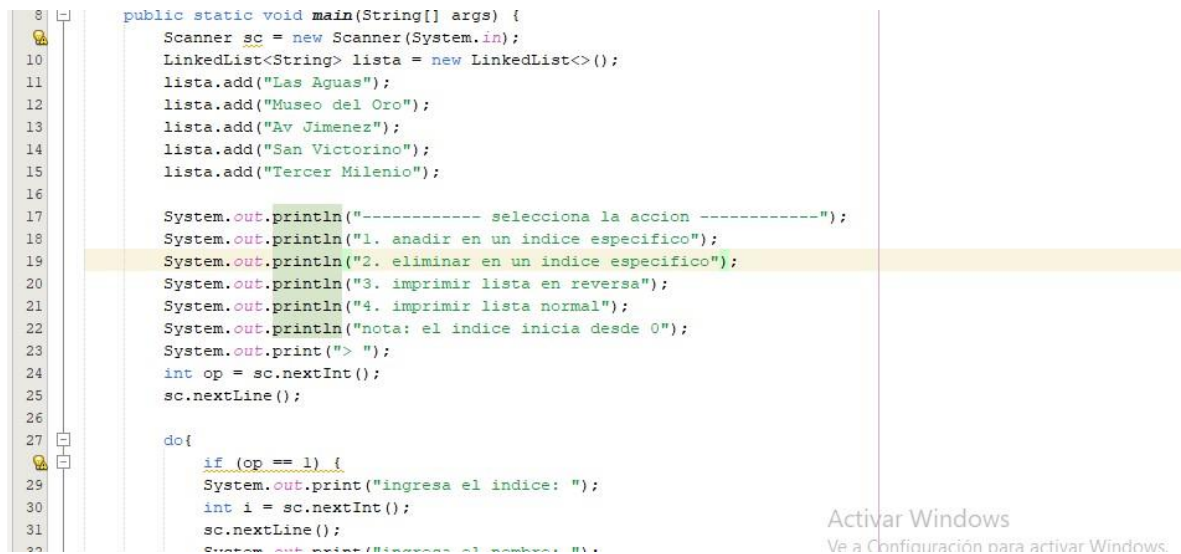
```
12      lista.add(3);
13      lista.add(7);
14      lista.add(9);
15      lista.add(12);
16      lista.add(15);
17
18      System.out.println("----- selecciona la accion -----");
19      System.out.println("1. anadir en un indice especifico");
20      System.out.println("2. eliminar en un indice especifico");
21      System.out.println("3. imprimir lista en reversa");
22      System.out.println("4. imprimir lista normal");
23      System.out.println("nota: el indice inicia desde 0");
24      System.out.print("> ");
25      int op = sc.nextInt();
26
27      if (op == 1) {
28          System.out.print("ingresa el indice: ");
29          int i = sc.nextInt();
30          System.out.print("ingresa el valor: ");
31          int v = sc.nextInt();
32          if (i >= 0 && i <= lista.size()) {
```

Activar Windows  
Ve a Configuración para activar Windows.

*Nota: Fragmento de Código: Método Main del algoritmo.*

## Punto 2: Algoritmo de Transmilenio

*Figura 2.*



```
8 public static void main(String[] args) {
9     Scanner sc = new Scanner(System.in);
10    LinkedList<String> lista = new LinkedList<>();
11    lista.add("Las Aguas");
12    lista.add("Museo del Oro");
13    lista.add("Av Jimenez");
14    lista.add("San Victorino");
15    lista.add("Tercer Milenio");
16
17    System.out.println("----- selecciona la accion -----");
18    System.out.println("1. anadir en un indice especifico");
19    System.out.println("2. eliminar en un indice especifico");
20    System.out.println("3. imprimir lista en reversa");
21    System.out.println("4. imprimir lista normal");
22    System.out.println("nota: el indice inicia desde 0");
23    System.out.print("> ");
24    int op = sc.nextInt();
25    sc.nextLine();
26
27    do{
28        if (op == 1) {
29            System.out.print("ingresa el indice: ");
30            int i = sc.nextInt();
31            sc.nextLine();
32            System.out.print("ingresa el nombre: ");
```

*Nota: Fragmento de Código: Clase 2do algoritmo*

### Punto 3: Algoritmo de lista de números con mis funciones propias

Figura 3.

```
LinkedList<Integer> lista = new LinkedList<>();
lista.add(3);
lista.add(7);
lista.add(9);
lista.add(12);
lista.add(15);

System.out.println("----- selecciona la accion -----");
System.out.println("1. anadir en un indice especifico");
System.out.println("2. eliminar en un indice especifico");
System.out.println("3. imprimir lista en reversa");
System.out.println("4. imprimir lista normal");
System.out.println("nota: el indice inicia desde 0");
System.out.print("> ");
int op = sc.nextInt();

if (op == 1) {
    System.out.print("ingresa el indice: ");
    int i = sc.nextInt();
    System.out.print("ingresa el valor: ");
```

Activar Windows

*Nota: Fragmento deCodigo: Clase 3er algoritmo*

### **Enlace**

.Enlace al Repositorio en GitHub del Proyecto:

<https://github.com/Jass-dotcom/ListasDobles>

**3. ¿Que casos cotidianos hay como ejemplos de listas doblemente enlazadas?**

En el archivo .txt en GitHub estan descritos los ejemplos.

## **Resultados**

Con la realizacion de este programa se comprendio mejor el funcionamiento interno de una lista doblemente enlazada, asi como los procesos de insercion, eliminacion y recorrido de datos en ambas direcciones. al desarrollar los metodos manualmente sin depender de funciones predefinidas, fue posible entender de manera mas profunda la logica que hay detras de las estructuras dinamicas y su importancia para optimizar la manipulacion de informacion en diferentes aplicaciones.