



**UNIVERSIDAD NACIONAL
AUTÓNOMA DE MÉXICO**



**FACULTAD DE ESTUDIOS SUPERIORES
ARAGÓN**

Nombre del curso:
ESTRUCTURA DE DATOS

Profesor:
Jesús Hernández Cabrera

Tarea:
T4, LISTAS SEMPLIMENTE LIGADAS

Alumno:
Navarro Rocha Miguel Ángel

Fecha:
03-09-24

Código

Codigo Class Listas (Principal)

```
package ListasLigadas;

public class Listas<T> {

    public static void main(String[] args) {
        ListasLigadaADT<SmartPhone> lista = new ListasLigadaADT<>();

        // Crear y agregar 5 smartphones
        SmartPhone sp1 = new SmartPhone("Apple iPhone 14");
        SmartPhone sp2 = new SmartPhone("Samsung Galaxy S23");
        SmartPhone sp3 = new SmartPhone("Motorola Edge 40");
        SmartPhone sp4 = new SmartPhone("Apple iPhone 15");
        SmartPhone sp5 = new SmartPhone("Xiaomi Mi 11");

        lista.agregarAlFinal(sp1);
        lista.agregarAlFinal(sp2);
        lista.agregarAlFinal(sp3);
        lista.agregarAlFinal(sp4);
        lista.agregarAlFinal(sp5);

        // Imprimir contenido
        System.out.println("Contenido de la lista:");
        lista.transversal();

        // Eliminar el de la posición 2
        lista.eliminar(2);
        System.out.println("Después de eliminar el elemento en la posición 2:");
        lista.transversal();

        // Actualizar el segundo elemento
        SmartPhone nuevoSp2 = new SmartPhone("Samsung Galaxy S23 Ultra");
        lista.actualizar(sp2, nuevoSp2);
        System.out.println("Después de actualizar el segundo elemento:");
        lista.transversal();

        // Agregar un elemento al inicio y otro al final
        SmartPhone spInicio = new SmartPhone("Sony Xperia 5");
        SmartPhone spFinal = new SmartPhone("Huawei P50 Pro");
        lista.agregarAlInicio(spInicio);
        lista.agregarAlFinal(spFinal);
        System.out.println("Después de agregar un elemento al inicio y otro al final:");
        lista.transversal();

        // Eliminar el primero
        lista.eliminarPrimero();
        System.out.println("Después de eliminar el primer elemento:");
        lista.transversal();
    }
}
```

```
}  
}
```

Codigo Class ListasADT.

```
package ListasLigadas;  
  
public class ListasLigadaADT<T> {  
    private Nodo<T> head;  
    private int tamano;  
  
    public ListasLigadaADT() {  
        this.head = null;  
        this.tamano = 0;  
    }  
  
    public boolean estaVacia() {  
        return this.head == null;  
    }  
  
    public int getTamano() {  
        return this.tamano;  
    }  
  
    public void agregarAlFinal(T valor) {  
        Nodo<T> nuevo = new Nodo<>(valor);  
        if (this.head == null) {  
            this.head = nuevo;  
        } else {  
            Nodo<T> aux = this.head;  
            while (aux.getNext() != null) {  
                aux = aux.getNext();  
            }  
            aux.setNext(nuevo);  
        }  
        this.tamano++;  
    }  
  
    public void agregarAlInicio(T valor) {  
        Nodo<T> nuevo = new Nodo<>(valor);  
        if (this.head != null) {  
            nuevo.setNext(this.head);  
        }  
        this.head = nuevo;  
        this.tamano++;  
    }  
  
    public void agregarDespuesDe(T referencia, T valor) {  
        Nodo<T> aux = this.head;
```

```

while (aux != null && !aux.getDato().equals(referencia)) {
    aux = aux.getNext();
}
if (aux != null) {
    Nodo<T> nuevo = new Nodo<>(valor);
    nuevo.setNext(aux.getNext());
    aux.setNext(nuevo);
    this.tamanio++;
}
}

```

```

public void eliminar(int posicion) {
    if (posicion < 0 || posicion >= this.tamanio) {
        System.out.println("Posición inválida.");
        return;
    }
    if (posicion == 0) {
        this.head = this.head.getNext();
    } else {
        Nodo<T> aux = this.head;
        for (int i = 0; i < posicion - 1; i++) {
            aux = aux.getNext();
        }
        aux.setNext(aux.getNext().getNext());
    }
    this.tamanio--;
}

```

```

public void eliminarPrimero() {
    if (this.head != null) {
        this.head = this.head.getNext();
        this.tamanio--;
    }
}

```

```

public void eliminarFinal() {
    if (this.head == null) {
        return;
    }
    if (this.head.getNext() == null) {
        this.head = null;
    } else {
        Nodo<T> aux = this.head;
        while (aux.getNext().getNext() != null) {
            aux = aux.getNext();
        }
        aux.setNext(null);
    }
    this.tamanio--;
}

```

```

public void buscar(T valor) {
    Nodo<T> aux = this.head;
    int indice = 0;
    while (aux != null) {
        if (aux.getDato().equals(valor)) {
            System.out.println("El elemento " + valor + " se encuentra en la posición: "
+ indice);
            return;
        }
        aux = aux.getNext();
        indice++;
    }
    System.out.println("El elemento " + valor + " no se encontró en la lista.");
}

public void actualizar(T aBuscar, T valor) {
    Nodo<T> aux = this.head;
    while (aux != null && !aux.getDato().equals(aBuscar)) {
        aux = aux.getNext();
    }
    if (aux != null) {
        aux.setDato(valor);
        System.out.println("El elemento " + aBuscar + " fue actualizado a " + valor +
""");
    } else {
        System.out.println("El elemento " + aBuscar + " no se encontró en la lista.");
    }
}

public void transversal() {
    Nodo<T> aux = this.head;
    while (aux != null) {
        System.out.println(aux.getDato());
        aux = aux.getNext();
    }
    System.out.println();
}
}

```

Código Class Nodo

```

package ListasLigadas;

public class Nodo <T>{

    private T value;

    private Nodo <T> next;

```

```

public Nodo() {}

public Nodo(T valor) {
    this.value = valor;}

public Nodo(T dato, Nodo <T> siguiente) {
    this.value = dato;
    this.next = siguiente;}

public Nodo <T> getNext() {
    return next;}

public void setNext(Nodo <T> siguiente) {
    this.next = siguiente;}

public T getDato() {
    return value;}

public void setDato(T dato) {
    this.value = dato;}

@Override
public String toString() {
    return "/" + value + " ";}
}

```

Código Class SmartPhone

```

package ListasLigadas;

public class SmartPhone {
    private String modelo;

    public SmartPhone(String modelo) {
        this.modelo = modelo;
    }

    public String getModelo() {
        return modelo;
    }

    public void setModelo(String modelo) {
        this.modelo = modelo;
    }

    @Override
    public String toString() {
        return "SmartPhone [Modelo=" + modelo + " ]";
    }
}

```

```
}  
}
```

Capturas de ejecución

```
Run  Listas  
"C:\Program Files\Java\jdk-17\bin\java.exe" "-javaagent:C:\Program Files\JetBrains\IntelliJ IDEA 2024.2\lib\idea_rt.jar=56291:C:\Program Files\JetBrains\IntelliJ IDEA 2024.  
Contenido de la lista:  
SmartPhone [Modelo=Apple iPhone 14]  
SmartPhone [Modelo=Samsung Galaxy S23]  
SmartPhone [Modelo=Motorola Edge 40]  
SmartPhone [Modelo=Apple iPhone 15]  
SmartPhone [Modelo=Xiaomi Mi 11]  
  
Después de eliminar el elemento de la posición 2:  
SmartPhone [Modelo=Apple iPhone 14]  
SmartPhone [Modelo=Samsung Galaxy S23]  
SmartPhone [Modelo=Apple iPhone 15]  
SmartPhone [Modelo=Xiaomi Mi 11]  
  
El elemento 'SmartPhone [Modelo=Samsung Galaxy S23]' fue actualizado a 'SmartPhone [Modelo=Samsung Galaxy S23 Ultra]'  
Después de actualizar el segundo elemento:  
SmartPhone [Modelo=Apple iPhone 14]  
SmartPhone [Modelo=Samsung Galaxy S23 Ultra]  
SmartPhone [Modelo=Apple iPhone 15]  
SmartPhone [Modelo=Xiaomi Mi 11]  
  
Después de agregar un elemento al inicio y otro al final:  
SmartPhone [Modelo=Sony Xperia 5]  
SmartPhone [Modelo=Apple iPhone 14]  
SmartPhone [Modelo=Samsung Galaxy S23 Ultra]  
SmartPhone [Modelo=Apple iPhone 15]  
SmartPhone [Modelo=Xiaomi Mi 11]  
SmartPhone [Modelo=Huawei P50 Pro]
```

```
Después de eliminar el primer elemento:  
SmartPhone [Modelo=Apple iPhone 14]  
SmartPhone [Modelo=Samsung Galaxy S23 Ultra]  
SmartPhone [Modelo=Apple iPhone 15]  
SmartPhone [Modelo=Xiaomi Mi 11]  
SmartPhone [Modelo=Huawei P50 Pro]  
  
Process finished with exit code 0
```