

Estructura de Datos

Amy Cárdenas Silva

GRupo: 1360

Carrera: ingenieria en computacion

Tarea 4, muestra de funcionamiento

```

import java.util.List;

public class Main {
    public static void main(String[] args) {

        Smartphone telefono1 = new Smartphone("iPhone 4", "IOS");
        Smartphone telefono2 = new Smartphone("Samsung Galaxy s24",
"android");
        Smartphone telefono3 = new Smartphone("motorola razr 40", "android");
        Smartphone telefono4 = new Smartphone("iPhone 13" , "IOS");
        Smartphone telefono5 = new Smartphone("xiaomi 13T", "android");
        ListaLigada ll = new ListaLigada<>();
        ll.agregarAlFinal(telefono1);
        ll.agregarAlFinal(telefono2);
        ll.agregarAlFinal(telefono3);
        ll.agregarAlFinal(telefono4);
        ll.agregarAlFinal(telefono5);
        ll.transversal();
        ll.eliminar(2);
        ll.transversal();
        ll.actualizar(telefono3,new Smartphone("iPhone4s", "IOS"));
        ll.agregarAlInicio(new Smartphone("lumia", "Windows Phone"));
        ll.agregarAlFinal(new Smartphone("Honor 200","android"));
        ll.transversal();
        ll.eliminarElPrimero();
        ll.transversal();
    }
}

```

```

public class Smartphone {
    private String nombre;
    private String sistemaOperativo;
    public Smartphone() {
        this.nombre = null;
        this.sistemaOperativo = null;
    }
    public Smartphone(String nombre, String sistemaOperativo){
        this.nombre = nombre;
        this.sistemaOperativo = sistemaOperativo;
    }
    public String getNombre() {

```

```

        return this.nombre;
    }

    public String getSistemaOperativo() {
        return this.sistemaOperativo;
    }

    public void setNombre(String nombre) {
        this.nombre=nombre;
    }

    public void setSistemaOperativo(String sistemaOperativo) {
        this.sistemaOperativo=sistemaOperativo;
    }

    @Override
    public String toString() {
        return "{" +
            nombre + '\n' +
            ",SO=" + sistemaOperativo +
            "'";
    }
}

```

```

public class ListaLigada<T> {
    private Nodo<T> head;
    private int tamano;

    public ListaLigada() {
    }

    public boolean estaVacía() {

        boolean resultado = false;

        if (this.head == null) {

            resultado = true;

        }

        return resultado;
    }
}

```

```
public int getTamanio(){
    Nodo aux = this.head;
    int contador = 0;
    while (aux!=null){
        contador++;
        aux = aux.getSiguiente();
    }
    return contador;
}

public void agregarAlFinal(T dato){
    Nodo<T> nuevo = new Nodo<>(dato);
    if(head == null){
        head = nuevo;
    }else {
        Nodo<T> aux = this.head;
        while (aux.getSiguiente() != null){
            aux = aux.getSiguiente();
        }
        aux.setSiguiente(nuevo);
    }
}

public void agregarAlInicio(T dato){

    this.head = new Nodo<>(dato, this.head);

}

public void agregarDespuesDe(T referencia, T dato){

    Nodo<T> auxiliar = this.head;

    while (auxiliar.getDato() != referencia){

        auxiliar = auxiliar.getSiguiente();

    }
}
```

```
        auxiliar.setSiguiente(new Nodo<T>(dato, auxiliar.getSiguiente()));
    }

    public void eliminar(Integer pos){

        Nodo<T> aux = this.head;

        if(pos != 0) {
            for (int i = 0; i == pos; i++) {
                aux = aux.getSiguiente();
            }
            aux.setSiguiente(aux.getSiguiente().getSiguiente());
        }else{
            this.head = this.head.getSiguiente();
        }
    }

    public void eliminarElPrimero(){

        this.head = this.head.getSiguiente();

    }

    public void eliminarElFinal(){
        Nodo<T> auxiliar = this.head;
        if (getTamanio() > 1){

            while (auxiliar.getSiguiente().getSiguiente() != null){

                auxiliar = auxiliar.getSiguiente();

            }

            auxiliar.setSiguiente(null);

        }else {

            this.head = null;
        }
    }
}
```

```

    }
}

public int buscar(T valor) {

    if (estaVacia() == true) {
        return -1;
    }
    Nodo<T> auxiliar = this.head;

    int contador = 0;

    while (auxiliar.getDato() != valor) {

        auxiliar = auxiliar.getSiguiente();

        contador++;

    }

    return contador;

}

public void actualizar(T aBuscar, T dato) {

    Nodo<T> auxiliar = this.head;

    while (auxiliar.getDato() != aBuscar) {

        auxiliar = auxiliar.getSiguiente();

    }

    auxiliar.setDato(dato);

}

public void transversal() {
    Nodo<T> aux = this.head;
    while (aux != null) {

```

```

        System.out.print(aux.getDato() + " --> ");
        aux = aux.getSiguiente();
    }
    System.out.print("null");
    System.out.println("");
}
}

```

```

public class Nodo<T> {
    private T dato;
    private Nodo<T> siguiente;

    public Nodo() {
    }

    public Nodo(T dato) {
        this.dato = dato;
    }

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

    public T getDato() {
        return dato;
    }

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

    public Nodo<T> getSiguiente() {
        return siguiente;
    }

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

```

```

    }

    @Override
    public String toString() {
        return "Nodo{" +
            "dato=" + dato +
            ", siguiente=" + siguiente +
            '}';
    }
}

```

```

{iPhone 4',S0=IOS} --> {Samsung Galaxy s24',S0=android} --> {motorola razr 40',S0=android} --> {iPhone 13',S0=IOS} --> {xiaomi 13T',S0=android} --> null
{iPhone 4',S0=IOS} --> {motorola razr 40',S0=android} --> {iPhone 13',S0=IOS} --> {xiaomi 13T',S0=android} --> null
{lumia',S0=Windows Phone} --> {iPhone 4',S0=IOS} --> {iPhone4s',S0=IOS} --> {iPhone 13',S0=IOS} --> {xiaomi 13T',S0=android} --> {Honor 200',S0=android} --> null
{iPhone 4',S0=IOS} --> {iPhone4s',S0=IOS} --> {iPhone 13',S0=IOS} --> {xiaomi 13T',S0=android} --> {Honor 200',S0=android} --> null

```