



U
P
T

UNIVERSIDAD POLITÉCNICA DE TULANCINGO

Portafolio de Evidencias

por

Michael Cabrera García

**Ingeniería en Sistemas
Computacionales**

Asignatura:

Estructura de datos

Nombre del Catedrático:

Uriel Edgardo Escobar Franco

Cuarto Cuatrimestre

Grupo **ISC41**

Tulancingo de Bravo, Hidalgo

Septiembre-Diciembre 2021



Contenido

1	a) Listar los nombres de todos los pacientes hospitalizados.....	3
	b) Obtener la cantidad de pacientes hospitalizado en las siguientes categorías: niños (0-13 años), jóvenes (13-30 años) y adultos (mayores de 30).	3
2	c) Además deberá poder buscar un paciente por su nombre.	4
3	d) Eliminar pacientes de la lista	5
4	Clase App y Pruebas.....	6

1

a) Listar los nombres de todos los pacientes hospitalizados.

```
public String Nombres() {
    String cadena = "Nombres pacientes{";

    if (!isEmpty()) {
        // no está vacía (aquí si hacemos algo)
        Nodo<E> aux = primer_nodo;
        do {
            cadena += aux.getDato().getNombre().toString() + ",";
            aux = aux.getAnterior();

        } while (aux != null);
    }

    return cadena += "}";
}
```

1.1

b) Obtener la cantidad de pacientes hospitalizado en las siguientes categorías: niños (0-13 años), jóvenes (13-30 años) y adultos (mayores de 30).

```

public String Rangos() {

    String nin = "",ado="",adult="";
    String cadena = "Nombres pacientes 0-13 años{";

    if (!isEmpty()) {
        // no está vacía (aquí si hacemos algo)
        Nodo<E> aux = primer_nodo;

        do {

            if (aux.getDato().getEdad()<14 && aux.getDato().getEdad()>0)
            {
                nin += " ";
            }
            else if (aux.getDato().getEdad()<31 && aux.getDato().getEdad()>0) {
                ado += " ";
            }
            else if (aux.getDato().getEdad()>30 && aux.getDato().getEdad()>0) {
                adult += " ";
            }
            aux = aux.getAnterior();

        } while (aux != null);
    }

    return cadena += "Cantidad pacientes: " + "\n" + "Niños: " + nin + "\n" + "Jovenes: " + ado + "\n" + "Adultos: " + adul
}

```

2

c) Además deberá poder buscar un paciente por su nombre.

```

public String BuscarPorNombre(String nombre) {

    String cadena = "{";
    int enc = 0;
    if (!isEmpty()) {
        // no está vacía (aquí si hacemos algo)
        Nodo<E> aux = primer_nodo;

        do {
            if (nombre.equals(aux.getDato().getNombre()))
            {
                cadena += "Paciente encontrado " + aux.getDato();
                enc = 1;
            }
            aux = aux.getAnterior();

        } while (aux != null);
        if(enc==0){
            cadena = "(Paciente no encontrado: " + nombre;
        }
    }

    return cadena += "}";
}

```

3

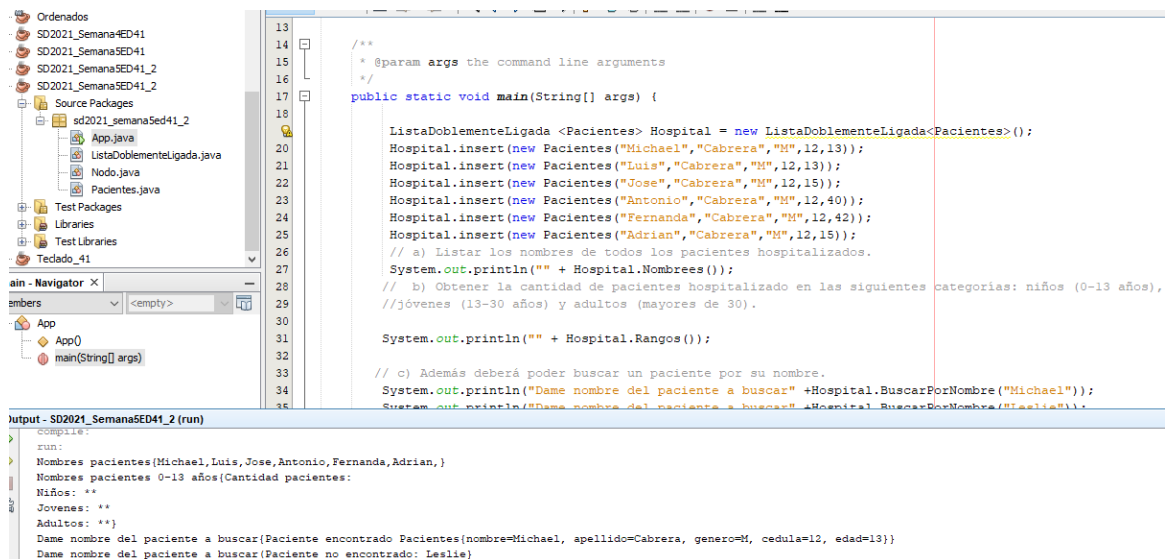
d) Eliminar pacientes de la lista

```

public String Eliminar(String nombre){
    E dato_eliminado =null;
    if(!isEmpty()){
        Nodo<E> aux=ultimo_nodo;
        boolean encontrado=false;
        while(aux != null && !encontrado){
            encontrado = (aux.getDatos().getNombre().equals(nombre));
            if(!encontrado){
                aux= aux.getSiguiente();
            }
            dato_eliminado=aux.getDatos();
        }
        if(encontrado){
            if(aux.equals(ultimo_nodo)){
                ultimo_nodo = aux.getSiguiente();
            }else{
                aux.getAnterior().setSiguiente(aux.getSiguiente());
                if(aux.getSiguiente() != null){
                    aux.getSiguiente().setAnterior(aux.getAnterior());
                }
            }
            aux=null;
        }
    }
    return nombre;
}

```

4 Clase App y Pruebas



The screenshot shows an IDE with a project named 'sd2021_semana5ed41_2'. The project structure includes 'App.java', 'ListaDoblementeLigada.java', 'Nodo.java', and 'Pacientes.java'. The 'App.java' file is open, showing the following code:

```

13
14
15 /**
16  * @param args the command line arguments
17  */
18 public static void main(String[] args) {
19
20     ListaDoblementeLigada <Pacientes> Hospital = new ListaDoblementeLigada<Pacientes>();
21     Hospital.insert(new Pacientes("Michael", "Cabrera", "M", 12, 13));
22     Hospital.insert(new Pacientes("Luis", "Cabrera", "M", 12, 13));
23     Hospital.insert(new Pacientes("Jose", "Cabrera", "M", 12, 15));
24     Hospital.insert(new Pacientes("Antonio", "Cabrera", "M", 12, 40));
25     Hospital.insert(new Pacientes("Fernanda", "Cabrera", "M", 12, 42));
26     Hospital.insert(new Pacientes("Adrian", "Cabrera", "M", 12, 15));
27     // a) Listar los nombres de todos los pacientes hospitalizados.
28     System.out.println("a) " + Hospital.Nombres());
29     // b) Obtener la cantidad de pacientes hospitalizado en las siguientes categorias: niños (0-13 años),
30     // jóvenes (13-30 años) y adultos (mayores de 30).
31     System.out.println("b) " + Hospital.Rangos());
32
33     // c) Además deberá poder buscar un paciente por su nombre.
34     System.out.println("c) Dame nombre del paciente a buscar" + Hospital.BuscarPorNombre("Michael"));
35     System.out.println("c) Dame nombre del paciente a buscar" + Hospital.BuscarPorNombre("Leslie"));
36 }

```

The output window shows the following results:

```

run:
Nombres pacientes[Michael,Luis,Jose,Antonio,Fernanda,Adrian,]
Nombres pacientes 0-13 años{Cantidad pacientes:
Niños: **
Jóvenes: **
Adultos: **}
Dame nombre del paciente a buscar{Paciente encontrado Pacientes(nombre=Michael, apellido=Cabrera, genero=M, cedula=12, edad=13)}
Dame nombre del paciente a buscar{Paciente no encontrado: Leslie}

```

```
Dame nombre del paciente a buscar(Paciente no encontrado: Leslie)
EliminadoMichael
BUILD SUCCESSFUL (total time: 0 seconds)
```