



**Universidad Nacional Autónoma de México**



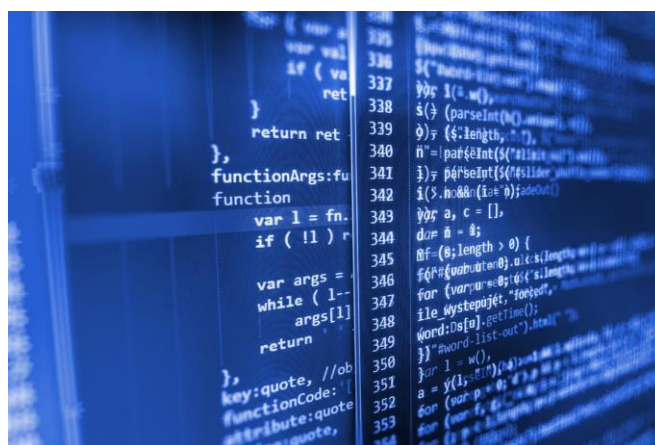
**Facultad de Estudios Superiores Aragón**

## **Ingeniería en Computación**

### **Estructura de Datos**

**Jesús Hernández Cabrera**

**Axel Yahir Moreno Rodríguez**



**Turno Vespertino**

**Grupo 1360**

```
... < -> Untitled (Workspace)
J Main.java x J ColaConPrioridadAcotada.java 2 J ColaADT.java J ClienteBanco.java
Estructura de Datos > Tareas > Tarea 8 > tarea8 > src > main > java > unam > mx > colas > J Main.java > Main > main(String[])
1 package unam.mx.colas;
2
3 public class Main {
4     Run | Debug
5     public static void main(String[] args) {
6         ColaConPrioridadAcotada<ClienteBanco> cliente = new ColaConPrioridadAcotada<>(maxPrioridad:5);
7
8         cliente.encolar(prioridad:4,new ClienteBanco(cliente:"Cliente Nuevo", nombre:"Diego Hernandez",saldo:5000));
9         cliente.encolar(prioridad:4,new ClienteBanco(cliente:"Cliente Nuevo", nombre:"Alex Herrera",saldo:1000));
10        cliente.encolar(prioridad:5,new ClienteBanco(cliente:"No es cliente",nombre:"Jose Perez", saldo:0));
11        cliente.encolar(prioridad:5,new ClienteBanco(cliente:"No es cliente",nombre:"Adrian Castillo", saldo:200));
12        cliente.encolar(prioridad:5,new ClienteBanco(cliente:"No es cliente",nombre:"Alejandro Suarez", saldo:5000));
13        cliente.encolar(prioridad:1,new ClienteBanco(cliente:"Celebridad",nombre:"Karol G",saldo:1900000));
14        System.out.println(cliente);
15        ClienteBanco clienteAtendido = cliente.desEncolar();
16        clienteAtendido.retirar(monto:10000);
17        cliente.encolar(prioridad:3,new ClienteBanco(cliente:"Cliente Frecuente",nombre:"Karen Rodriguez",saldo:50000));
18        cliente.encolar(prioridad:2,new ClienteBanco(cliente:"Cliente Premium",nombre:"Arturo Islas",saldo:1500000));
19        System.out.println("Atendiendo a : " + cliente.desEncolar());
20        System.out.println(cliente);
21        System.out.println("Atendiendo a : " + cliente.desEncolar());
22        System.out.println("Atendiendo a : " + cliente.desEncolar());
23        System.out.println("Atendiendo a : " + cliente.desEncolar());
24        System.out.println("Atendiendo a : " + cliente.desEncolar());
25        System.out.println("Atendiendo a : " + cliente.desEncolar());
26        System.out.println(cliente);
27    }
28 }
```

```
... < -> Untitled (Workspace)
J Main.java J ColaConPrioridadAcotada.java 2 J ColaADT.java J ClienteBanco.java
Estructura de Datos > Tareas > Tarea 8 > tarea8 > src > main > java > unam > mx > colas > J ColaConPrioridadAcotada.java > ColaConPrioridadAcotada<E> > estaVacia()
1 package unam.mx.colas;
2 import java.util.Arrays; The import java.util.Arrays is never used
3
4 public class ColaConPrioridadAcotada<E> {
5     private ColaADT<E> []colas;
6     private int maxPrioridad;
7
8     public ColaConPrioridadAcotada(int maxPrioridad) {
9         this.maxPrioridad = maxPrioridad;
10        this.colas = new ColaADT[maxPrioridad +1]; Type safety: The expression of type ColaADT[] needs unchecked conversion to
11        for (int i = 0; i < this.maxPrioridad +1; i++) {
12            this.colas[i] = new ColaADT<>();
13        }
14    }
15
16    public int longitud(){
17        int total = 0;
18        for (int i = 0; i < maxPrioridad+1; i++) {
19            total += this.colas[i].longitud();
20        }
21        return total;
22    }
23
24    public boolean estaVacia(){
25        return this.longitud() == 0;
26    }
27
28    public void encolar(int prioridad, E elemento){
29        if (prioridad >= 1 && prioridad <= this.maxPrioridad){
30            this.colas[prioridad].encolar(elemento);
31        }
32    }
33 }
```



```
PROBLEMS 18 OUTPUT DEBUG CONSOLE TERMINAL PORTS Run: Main + - [ ] [ ] ... ^ x
PS C:\Users\axelx\Documents\FES\3er semestre\Estructura de Datos>
PS C:\Users\axelx\Documents\FES\3er semestre\Estructura de Datos> c:; cd 'c:\Users\axelx\Documents\FES\3er semestre\Estructura de Datos'; &
'C:\Program Files\Java\jdk-21\bin\java.exe' '-XX:+ShowCodeDetailsInExceptionMessages' '-cp' 'C:\Users\axelx\Documents\FES\3er semestre\Estr
uctura de Datos\Tareas\Tarea 8\tarea8\target\classes' 'unam.mx.colas.Main'
Prioridad 1: [Cliente: 'Karol G', Saldo: $1900000.0, Tipo de cliente: Celebridad]
Prioridad 4: [Cliente: 'Diego Hernandez', Saldo: $5000.0, Tipo de cliente: Cliente Nuevo, Cliente: 'Alex Herrera', Saldo: $1000.0, Tipo de c
liente: Cliente Nuevo]
Prioridad 5: [Cliente: 'Jose Perez', Saldo: $0.0, Tipo de cliente: No es cliente, Cliente: 'Adrian Castillo', Saldo: $200.0, Tipo de cliente
: No es cliente, Cliente: 'Alejandro Suarez', Saldo: $5000.0, Tipo de cliente: No es cliente]

Karol G retiro $10000.0. Nuevo saldo: $1890000.0
Atendiendo a : Cliente: 'Arturo Islas', Saldo: $1500000.0, Tipo de cliente: Cliente Premium
Prioridad 3: [Cliente: 'Karen Rodriguez', Saldo: $50000.0, Tipo de cliente: Cliente Frecuente]
Prioridad 4: [Cliente: 'Diego Hernandez', Saldo: $5000.0, Tipo de cliente: Cliente Nuevo, Cliente: 'Alex Herrera', Saldo: $1000.0, Tipo de c
liente: Cliente Nuevo]
Prioridad 5: [Cliente: 'Jose Perez', Saldo: $0.0, Tipo de cliente: No es cliente, Cliente: 'Adrian Castillo', Saldo: $200.0, Tipo de cliente
: No es cliente, Cliente: 'Alejandro Suarez', Saldo: $5000.0, Tipo de cliente: No es cliente]

Atendiendo a : Cliente: 'Karen Rodriguez', Saldo: $50000.0, Tipo de cliente: Cliente Frecuente
Atendiendo a : Cliente: 'Diego Hernandez', Saldo: $5000.0, Tipo de cliente: Cliente Nuevo
Atendiendo a : Cliente: 'Alex Herrera', Saldo: $1000.0, Tipo de cliente: Cliente Nuevo
Atendiendo a : Cliente: 'Jose Perez', Saldo: $0.0, Tipo de cliente: No es cliente
Atendiendo a : Cliente: 'Adrian Castillo', Saldo: $200.0, Tipo de cliente: No es cliente
Atendiendo a : Cliente: 'Alejandro Suarez', Saldo: $5000.0, Tipo de cliente: No es cliente
La lista esta vacia :)

PS C:\Users\axelx\Documents\FES\3er semestre\Estructura de Datos> [ ]
```

```
package unam.mx.colas;

public class Main {
    public static void main(String[] args) {
        ColaConPrioridadAcotada<ClienteBanco> cliente = new
ColaConPrioridadAcotada<>(5);

        cliente.encolar(4,new ClienteBanco("Cliente Nuevo", "Diego
Hernandez",5000));
        cliente.encolar(4,new ClienteBanco("Cliente Nuevo", "Alex
Herrera",1000));
        cliente.encolar(5,new ClienteBanco("No es cliente","Jose Perez",
0));
        cliente.encolar(5,new ClienteBanco("No es cliente","Adrian
Castillo", 200));
        cliente.encolar(5,new ClienteBanco("No es cliente","Alejandro
Suarez", 5000));
        cliente.encolar(1,new ClienteBanco("Celebridad","Karol G",1900000));
        System.out.println(cliente);
        ClienteBanco clienteAtendido = cliente.desEncolar();
        clienteAtendido.retirar(10000);
        cliente.encolar(3,new ClienteBanco("Cliente Frecuente","Karen
Rodriguez",50000));
        cliente.encolar(2,new ClienteBanco("Cliente Premium","Arturo
Islas",1500000));
        System.out.println("Atendiendo a : " + cliente.desEncolar());
        System.out.println(cliente);
        System.out.println("Atendiendo a : " + cliente.desEncolar());
        System.out.println("Atendiendo a : " + cliente.desEncolar());
```

```

        System.out.println("Atendiendo a : " + cliente.desEncolar());
        System.out.println("Atendiendo a : " + cliente.desEncolar());
        System.out.println("Atendiendo a : " + cliente.desEncolar());
        System.out.println("Atendiendo a : " + cliente.desEncolar());
        System.out.println(cliente);
    }
}

```

```

package unam.mx.colas;
import java.util.Arrays;

public class ColaConPrioridadAcotada<E> {
    private ColaADT<E> []colas;
    private int maxPrioridad;

    public ColaConPrioridadAcotada(int maxPrioridad) {
        this.maxPrioridad = maxPrioridad;
        this.colas = new ColaADT[maxPrioridad +1];
        for (int i = 0; i < this.maxPrioridad +1; i++) {
            this.colas[i] = new ColaADT<>();
        }
    }

    public int longitud(){
        int total = 0;
        for (int i = 0; i < maxPrioridad+1; i++) {
            total += this.colas[i].longitud();
        }
        return total;
    }

    public boolean estaVacia(){
        return this.longitud() == 0;
    }

    public void encolar(int prioridad, E elemento){
        if (prioridad >= 1 && prioridad <= this.maxPrioridad){
            this.colas[prioridad].encolar(elemento);
        }
    }

    public E desEncolar(){
        if (this.estaVacia()){
            System.out.println("No hay mas elementos");
        }
    }
}

```

```

        return null;
    }else {
        for (int i = 0; i <= maxPrioridad; i++) {
            if (!this.colas[i].estaVacia()) {
                return this.colas[i].desEncolar();
            }
        }
    }
    return null;
}

@Override
public String toString() {
    StringBuilder sb = new StringBuilder();

    for (int i = 0; i <= maxPrioridad; i++) {
        if (!this.colas[i].estaVacia()) {
            sb.append("Prioridad ").append(i).append(":
").append(this.colas[i].toString()).append("\n");
        }
    }
    if(this.estaVacia())
    {
        System.out.println("La lista esta vacia :");
    }
    return sb.toString();
}
}

```

```

package unam.mx.colas;

import java.util.LinkedList;

public class ColaADT<E> {
    private LinkedList<E> data;

    public ColaADT() {
        this.data = new LinkedList<>();
    }

    public boolean estaVacia(){
        boolean res = false;
        if (this.data.size() == 0){

```

```

        res = true;
    }
    return res;
}

public int longitud(){
    return this.data.size();
}

public E frente(){
    return this.data.getFirst();
}

public void encolar(E valor){
    this.data.addLast(valor);
}

public E desEncolar(){
    return this.data.removeFirst();
}

@Override
public String toString() {
    return data.toString();
}
}

```

```

package unam.mx.colas;

public class ClienteBanco {
    private String cliente;
    private String nombre;
    private double saldo;

    public ClienteBanco() {
    }

    public ClienteBanco(String cliente, String nombre, double saldo) {
        this.cliente = cliente;
        this.nombre = nombre;
        this.saldo = saldo;
    }

    public String getCliente() {

```

```
        return cliente;
    }

    public void setCliente(String cliente) {
        this.cliente = cliente;
    }

    public String getNombre() {
        return nombre;
    }

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

    public double getSaldo() {
        return saldo;
    }

    public void setSaldo(double saldo) {
        this.saldo = saldo;
    }

    public void retirar(double monto) {
        if (this.saldo >= monto) {
            this.saldo -= monto;
            System.out.println(nombre + " retiro $" + monto + ". Nuevo
saldo: $" + saldo);
        }
    }

    @Override
    public String toString() {
        return "Cliente: '" + nombre + "', Saldo: $" + saldo + ", Tipo de
cliente: " + cliente;
    }
}
```