



# EVALUACION PROCESUAL

HITO 4

ESTRUCTURA DE DATOS

INGENIERIA DE SISTEMAS

VICTOR HUGO QUISPE TORREZ

# Parte Conceptual

1. ¿A que se refiere cuando se habla de ESTRUCTURA DE DATOS?

Las estructuras de datos son una forma de organizar los datos en la computadora, de tal manera que nos permita realizar unas operaciones con ellas de forma muy eficiente.

2. ¿Que significa FIFO?

Es un termino escrito en ingles que signiica: “First In, First Out”, dando a entender que es el “Primero a entrar, primero en salir”

3. ¿Muestra la diferencia entre LIFO y FIFO?

LIFO, es una terminología para pilas que También esta escrito en ingles que dice: "Last In, First Out" que da entender que es: "Ultimo en entrar, Primero en Salir". Mientras que FIFO es una terminología para colas.

4. ¿Qué es una COLA?

Una cola es una estructura de datos que almacena elementos en una lista y permite acceder a los datos por uno de los dos extremos de la lista.

5. ¿Qué es QUEUE en JAVA, una QUEUE será lo mismo que una COLA?

Una queue es una estructura de datos lineal que sigue el principio FIFO (primero en entrar, primero en salir). Eso significa que el objeto insertado primero será el primero en salir, seguido del objeto insertado a continuación.

6. ¿Qué es INI o REAR en una COLA?

Es el extremo donde los datos entran llamado el inicio de la cola.

7. ¿Qué es FIN o FRONT en una COLA

Es el otro extremo donde se inserta los datos de la cola.

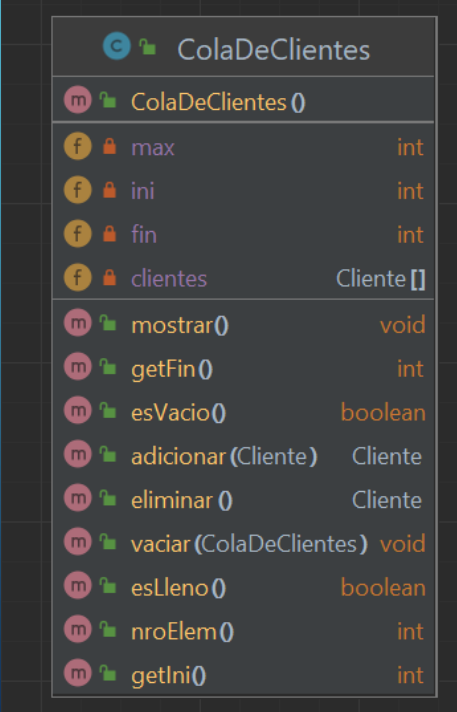
8. ¿A que se refiere los métodos esVacia() y esLlena() en una COLA?

```
public boolean esVacio(){  
    if(ini==0&&fin==0){  
        return true;}else{  
            return false;  
        }  
}  
  
1 usage  
public boolean esLleno(){  
    if(fin==max){  
        return true;}  
    else{  
        return false;  
    }  
}
```

9. ¿Qué son los métodos estáticos en JAVA?

Las variables y métodos estáticos en Java sirven para que puedan ser accedidos desde cualquier parte del código (inclusive desde otras clases) sin tener que crear un objeto.

10. ¿A través de un gráfico, muestre los métodos mínimos que debería de tener una COLA?



The screenshot shows a Java IDE window titled 'ColaDeClientes'. It displays the class structure with variables and methods. The variables are 'max', 'ini', 'fin', and 'clientes', all of type 'int' or 'Cliente[]'. The methods are 'mostrar()', 'getFin()', 'esVacio()', 'adicionar(Cliente)', 'eliminar()', 'vaciar(ColaDeClientes)', 'esLleno()', 'nroElem()', and 'getIni()'. The return types for the methods are 'void', 'int', 'boolean', 'Cliente', and 'int'.

Variable	Valor	Tip
max		int
ini		int
fin		int
clientes		Cliente[]

Método	Retorno
mostrar()	void
getFin()	int
esVacio()	boolean
adicionar(Cliente)	Cliente
eliminar()	Cliente
vaciar(ColaDeClientes)	void
esLleno()	boolean
nroElem()	int
getIni()	int

# Parte Practica



## 11. Crear las clases necesarias para la PILA DE CLIENTES.



- Crear la clase **Cliente**
- Crear la clase **ColaDeClientes**
- Crear la clase **Main**.

## 12. Inicializar la cola de clientes.

- Crear una cola con 5 clientes.
  - En la clase MAIN deberán estar los 5 clientes.
  - Mostrar todos los datos de la cola de clientes

```
public class MainColas {  
    public static void main(String[] args) {  
  
        ColaDeClientes colaClientes=new ColaDeClientes();  
        ColaDeClientes colaB=new ColaDeClientes();  
  
        Cliente cliente1=new Cliente( nombres: "Manuel", apellidos: "Lopez", edad: 62, pais: "Bolivia", genero: "Masculino", tipo: "GOLD");  
        Cliente cliente2=new Cliente( nombres: "Noelia", apellidos: "Garcia", edad: 24, pais: "Chile", genero: "Femenino", tipo: "PLATINUM");  
        Cliente cliente3=new Cliente( nombres: "Gabriela", apellidos: "Fernandez", edad: 28, pais: "Bolivia", genero: "Femenino", tipo: "GOLD");  
        Cliente cliente4=new Cliente( nombres: "Saul", apellidos: "Gonzales", edad: 63, pais: "Bolivia", genero: "Masculino", tipo: "SILVER");  
        Cliente cliente5=new Cliente( nombres: "Javier", apellidos: "Perez", edad: 26, pais: "Chile", genero: "Masculino", tipo: "SILVER");  
  
        colaClientes.adicionar(cliente1);  
        colaClientes.adicionar(cliente2);  
        colaClientes.adicionar(cliente3);  
        colaClientes.adicionar(cliente4);  
        colaClientes.adicionar(cliente5);  
    }  
}
```

### 13.Promoción para usuarios de Bolivia.

- En el mes de diciembre a todos los clientes de Bolivia se les dará una promoción en cuanto a precios en viajes a nivel nacional.
  - A todos los clientes que sean de nacionalidad boliviana y además el tipo de cliente GOLD, convertir a estos clientes en VIP
  - Es decir si es de Bolivia y es GOLD deberá ser ahora un cliente VIP
- El método estático dentro de la clase MAIN recibe 3 atributos
  - La cola de clientes
  - El tipo de cliente
  - La nacionalidad del cliente.

```
public static void promoverCliente(ColaDeClientes cola,String tipo, String nacionalidad){
    ColaDeClientes aux=new ColaDeClientes();
    Cliente item;
    while(!cola.esVacio()){
        item=cola.eliminar();
        if(Objects.equals(item.getPais(),nacionalidad) && Objects.equals(item.getTipo(),tipo)){
            item.setTipo("VIP");
        }
        aux.adicionar(item);
    }
    cola.vaciar(aux);
    cola.mostrar();
}
```

## 14.Moviendo clientes en la cola.

- Mover al inicio todos los clientes mayores a 60 años.
  - Es decir si el cliente es mayor a 60 deberá de moverlo al inicio de la cola.
- El método recibe 2 parámetros
  - La Cola de Clientes
  - El valor(int) de la edad.

```
public static void moverMayores(ColaDeClientes cola, int edad){
    ColaDeClientes aux=new ColaDeClientes();
    ColaDeClientes orden=new ColaDeClientes();
    Cliente item;
    while (!cola.esVacio()){
        item=cola.eliminar();
        if(item.getEdad()>edad){
            aux.adicionar(item);
        }else {
            orden.adicionar(item);
        }
    }
    cola.vaciar(aux);
    cola.vaciar(orden);
    cola.mostrar();
}
```

## 15. Moviendo clientes entre 2 colas.

- Por razones de promociones de vuelo, es necesario cambiar de vuelo a ciertos clientes.
  - Crear 2 colas con 5 clientes.
  - Todos los clientes cuyo nombre sea **Saul** deberán ser agregados a la cola **B al inicio**.

```
public static void moverCliente(ColaDeClientes colaA, ColaDeClientes colaB, String nombre){
    ColaDeClientes auxA=new ColaDeClientes();
    ColaDeClientes auxB=new ColaDeClientes();
    Cliente item;
    Cliente itemB;
    while (!colaA.esVacio()){
        item=colaA.eliminar();
        itemB=colaB.eliminar();
        if(Objects.equals(item.getNombres(), nombre)){
            auxB.adicionar(item);
        }
        else {
            auxA.adicionar(item);
        }
        auxB.adicionar(itemB);
    }
    colaA.vaciar(auxA);
    colaB.vaciar(auxB);
    colaA.mostrar();
    System.out.println("Mostrando COLA B");
    colaB.mostrar();
}
```