

Universidad ORT Uruguay
Facultad de Ingeniería
Escuela de Tecnología

OBLIGATORIO - ALGORITMOS Y ESTRUCT. DE DATOS



Taramasco, Sabrina – 180403



Villar, Florencia - 218320



Zanetti, Florencia – 283076

Grupo N3C, 2022

Docente: Guillermo Viera

Analista en Tecnologías de la información - Analista Programador

17/11/2022

Índice

Diagrama de Estructura de Datos	3
Clase Camion	3
Clase Cliente	3
Clase Producto	3
Clase Stock	3
Clase Retiro Y Orden De Espera	3
Juego de pruebas	4
2.1 Crear Sistema de Distribucion	4
.2 Crear Cliente	4
2.4 Agregar Camion	4
2.6 Registrar Producto	4
3.1 Registrar Alta de Stock de Producto	5
3.2 Retiro de Productos para Envio a Cliente	5
2.3 Eliminar Cliente	5
2.5 Eliminar Camion	5
4.4 Ultimo Producto Registrado	5
4.5 Listar Envíos De Producto	6
4.6 Listar Ordenes Pendientes	6
4.7 Reporte de Envios Por Producto	6
Firmas de la interface del sistema con las pre y post condiciones.	7
//2.1	7
//2.2	7
//2.3	7
//2.4	7
//2.5	7
//2.6	8
//3.1	8
//3.2	8
//4.1	8
//4.2	8
//4.3	8
//4.4	9
//4.5	9
//4.6	9
//4.7	9

Diagrama de Estructura de Datos

Camion	Cliente	OrdenDeEspera	Producto	Retiro	Stock
matricula	Nombre	MatriculaCam	Nombre	MatriculaCam	MaticulaCamion
toneladasMax	Rut	RutCliente	Descripcion	Rutcliente	CodigoProd
	tel	CodProducto	Codigo	CodProducto	NumeroCaja
	direccion	Cantidad	sockDisponible	Cantidad	CantidadUnidades
			ultimoCodigo		

Clase Camion

Camion tiene sus atributos propios y una matricula identificadora

Clase Cliente

Cliente tiene sus atributos propios y un rut identificador

Clase Producto

Producto tiene sus atributos propios y un codigo autogenerado identificador

Clase Stock

Stock tiene sus atributos propios y un nroCaja identificador

Clase Retiro Y Orden De Espera

Creamos 2 clases ya que sus listas son independientes, de todas formas creemos que estas 2 clases podrian ser una sola: "ORDEN", debido a que sus atributos son los mismos al igual que metodos y forma de identificarlos.

Juego de pruebas

2.1 Crear Sistema de Distribucion

```
IObligatorio obli = new Sistema();  
  
obli.crearSistemaDeDistribucion(200);
```

.2 Crear Cliente

```
obli.agregarCliente("Juan Perez", "1234567898765", 24876543, "8 de Octubre 2105");  
obli.agregarCliente("Romina Lopez", "12345672345", 26226250, "18 de Julio 2056");  
obli.agregarCliente("Floencia Ramirez", "6574367898765", 2678729, "Nin y Silva  
6574");  
obli.agregarCliente("Diego Martinez", "22224567898765", 24356789, "Rivera 4567");  
obli.agregarCliente("Julia Quintana", "222456756743", 26509876, "Bolivia 4352");  
obli.agregarCliente("Santiago Rivero", "5554567898976", 24878382, "Av Italia 1234");  
  
obli.listarClientesOrdenado();
```

2.4 Agregar Camion

```
obli.agregarCamion("ABC1234", 5);  
obli.agregarCamion("ABB1111", 9);  
obli.agregarCamion("ACC2222", 6);  
obli.agregarCamion("ADD3333", 4);  
obli.agregarCamion("AEE3333", 4);  
obli.listarCamiones();
```

2.6 Registrar Producto

```
obli.registrarProducto("Perfume Gardenia", "Aroma floral");  
obli.registrarProducto("TV 50pulgadas", "Smart TV HD ");  
obli.registrarProducto("Mesa de madera", "Madera de Nogal");  
obli.registrarProducto("Jabon liquido", "Lavanda 100ml");  
obli.listarProductos();
```

3.1 Registrar Alta de Stock de Producto

```
obli.altaDeStockDeProducto("ADD3333", 1, 1, 20);
obli.altaDeStockDeProducto("AEE3333", 2, 2, 13);
obli.altaDeStockDeProducto("ADD3333", 2, 3, 6);
obli.altaDeStockDeProducto("ACC2222", 3, 4, 8);
obli.altaDeStockDeProducto("ABC1234", 1, 4, 12);
obli.altaDeStockDeProducto("ABB1111", 3, 2, 9);
obli.listarProductos();
```

3.2 Retiro de Productos para Envio a Cliente

```
obli.retiroDeProducto("ADD3333", "1234567898765", 1, 20);
obli.retiroDeProducto("ABB1111", "222456756743", 2, 24);
obli.retiroDeProducto("ADD3333", "1234567898765", 3, 12);
obli.retiroDeProducto("ADD3333", "222456756743", 1, 3);
obli.retiroDeProducto("AEE3333", "6574367898765", 2, 5);
```

2.3 Eliminar Cliente

```
obli.eliminarCliente("1234567898765");
obli.eliminarCliente("12345672345");

obli.listarClientesOrdenado(); //VER ESTE ERROR
```

2.5 Eliminar Camion

```
obli.eliminarCamion("AEE3333");
obli.eliminarCamion("ACC2222");

obli.listarCamiones();
```

4.4 Ultimo Producto Registrado

```
obli.ultimoProductoRegistrado();
```

4.5 Listar Envíos De Producto

```
obli.listarEnvíosDeProducto(1);  
obli.listarEnvíosDeProducto(2);  
obli.listarEnvíosDeProducto(3);  
obli.listarEnvíosDeProducto(4);
```

4.6 Listar Ordenes Pendientes

```
obli.listarOrdenesPendientes(1);  
obli.listarOrdenesPendientes(2);  
obli.listarOrdenesPendientes(3);  
obli.listarOrdenesPendientes(4);
```

4.7 Reporte de Envios Por Producto

```
int[][] matriz = obli.reporteDeEnviosDeProductos();
```

Firmas de la interface del sistema con las pre y post condiciones.

//2.1

```
//pre: capacidadCajas != 0 && capacidadCajas > 0
//post: Retorna ERROR_1 si la capacidadCajas es menor o igual a cero, retorna OK si se
pudo inicializar el sistema correctamente
public Retorno crearSistemaDeDistribucion(int capacidadCajas);
```

//2.2

```
//pre: nombre != null && rut != null && tel != 0 && direccion != null
//post: Retorna OK si se logra registrar el cliente, ERROR_1 si ya existe un cliente con el
mismo rut
public Retorno agregarCliente(String nombre, String rut,int tel, String direccion);
```

//2.3

```
//pre: rut != null
//post: Retorna OK si se pudo eliminar el cliente, ERROR_1 si existe un cliente con el
mismo rut que otro, ERROR_2 si tiene entregas registradas
public Retorno eliminarCliente(String rut);
```

//2.4

```
//pre: matricula != null && toneladasMaxSoportadas != 0 && toneladasMaxSoportadas >
0
//post: Retorna OK si se pudo agregar el camión, ERROR_1 si se repite matricula,
ERROR_2 si las toneladasMaxSoportadas es menor o igual a 0
public Retorno agregarCamion(String matricula, int toneladasMaxSoportadas);
```

//2.5

```
//pre: matricula != null
//post: Retorna OK si se pudo eliminar el camión, ERROR_1 si no existe la matricula,
ERROR_2 si el camión tiene entregas realizadas
public Retorno eliminarCamion(String matricula);
```

//2.6

```
//pre: nombre != null && descripcion != null
//post: Retorna OK si se pudo registrar el producto, ERROR_1 si hay un producto que se
llame igual, ERROR_2 si la descripcion es vacia
public Retorno registrarProducto(String nombre, String descripcion);
```

//3.1

```
//pre: matriculaCamion != null && codigoProd != null && nroCaja != 0, cantUnidades >=
0
//post: Retorna OK si se pudo despachar el producto, ERROR_1 si no existe la matricula,
//ERROR_2 si no existe el codigoProd, ERROR_3 si la cantUnidades es menor o
igual a cero, ERROR_4 si ya existe el numero de caja,
//ERROR_5 si no existe capacidad en la fabrica para despachar
public Retorno altaDeStockDeProducto(String matriculaCamion, int codigoProd, int
nroCaja, int cantUnidades);
```

//3.2

```
//pre: matriculaCam != null && rutCliente != null && codProducto != 0
//post: Retorna OK si se pudo retirar total o parcialmente, ERROR_1 si no existe la
matricula,
//ERROR_2 si no existe el rutCliente, ERROR_3 si no existe el codProducto
public Retorno retiroDeProducto(String matriculaCam, String rutCliente, int codProducto,
int cant);
```

//4.1

```
//pre: post: Retorna la lista de camiones
public Retorno listarCamiones();
```

//4.2

```
//pre: post: Retorna la lista de clientes ordenados alfabeticamente
public Retorno listarClientesOrdenado();
```

//4.3

```
//pre: post: Retorna la lista de productos con el stock disponible
public Retorno listarProductos();
```


//4.4

//pre: post: Retorna el ultimo producto registrado
public Retorno ultimoProductoRegistrado();

//4.5

//pre: codProd != null post: Retorna OK si se muestran los envios correctamente,
ERROR_1 si no existe el codProd
public Retorno listarEnvíosDeProducto(int codProd);

//4.6

//pre: codProd != null post: Retorna OK si se listan las ordenes pendientes de manera
correcta, ERROR_1 si no existe el codProd
public Retorno listarOrdenesPendientes(int codProd);

//4.7

//pre: post: Retorna el reporte de manera correcta
public int[][] reporteDeEnviosDeProductos();