

## ***Incorporar un nuevo Producto***

### **Precondiciones:**

El producto debe tener una etiqueta, nombre, precio y stock  
El Almacen debe estar inicializado

### **Postcondiciones:**

Si el producto no existe en la lista, se agrega con su stock inicial.  
Si el producto ya existe, se agrega al stock existente del mismo producto.

### **Lenguaje Natural:**

Si el producto no está registrado, lo agrego a la lista de productos. Si ya está registrado, solo se actualiza su stock, aumentando la cantidad disponible.

### **Seudocódigo:**

Comienzo

Buscar el producto por su etiqueta

Si no existe el producto

    Agregar el producto a la lista

    Sino

        Actualizar el stock del producto

    FinSi

Fin

## ***Agregar stock a un producto existente***

### **Precondiciones:**

El producto debe existir en la lista.

### **Postcondiciones:**

El stock del producto se incrementa según la cantidad especificada.  
Si el producto no existe, no se realiza ninguna operación.

### **Lenguaje Natural:**

Primero, busco el producto por su etiqueta. Si lo encuentro, incremento su stock con la cantidad indicada. Si no lo encuentro, no hago nada.

### **Seudocódigo:**

```
boolean AñadirStock (clave,cantidad)
    Si producto.buscar esta en la lista
        producto.stock=producto.stock+nuevostock
        devolver true
    FinSi
devolver false
Fin
```

## ***Reducir el stock de un producto existente***

### ***Precondiciones:***

El producto debe existir en la lista.

### **Postcondiciones:**

El stock del producto se reduce en la cantidad indicada.

Si el stock es menor que cero, se ajusta a cero, pero el producto no se elimina.

**Lenguaje Natural:** Busco el producto por su etiqueta. Si lo encuentro, reduzco su stock por la cantidad indicada. Si el stock queda por debajo de cero, lo ajusto a cero. Si no encuentro el producto, no hago nada.

### **Seudocódigo**

```
Integer restarStock(clave,cantidad)

    si(producto esta en la lista)

        producto.stock=producto.stock-cantidad

        si(producto.stock es menor a 0)

            producto.stock=0;

        FinSi

    FinSi

Fin
```

## ***Listar todos los productos registrados, ordenados por nombre, presentando además su stock***

### **Precondiciones:**

- El almacén debe tener productos registrados en su lista.

### **Postcondiciones:**

- Se devuelve la lista de productos, ordenada alfabéticamente por su nombre, junto con su stock.

**Lenguaje Natural:** Recorro la lista de productos y los ordeno alfabéticamente por nombre. Luego, imprimo el nombre y el stock de cada producto.

### **Seudocódigo**

ListarOrdenadoPorNombre()

```
nuevaLista;
Nodo<Producto> actual=lista.primerO
mientras(actual no este vacio)
    clonado=actual
    si(nueva lista esta vacia)
        nuevalista.primerO=clonado
    sino
        aux1=nuevalista.primerO
        aux2=aux1.siguiente

        si(clonado va antes que aux1)
            clonado.siguiente=aux1
            nuevalista.primerO=clonado
        sino
            agregado=false
            mientras (aux2 no este vacio)
                si (clonado va antes que aux2)
                    aux1.siguiente=clonado
                    clonado.siguiente=aux2
                    agregado=true
                    break
                FinSi
                aux1=aux1.siguiente
                aux2=aux2.siguiente
            FinMientras

            Si (agregado=false)
                aux1.siguiene=clonado
            FinSi
        FinSi
    FinSi
actual=actual.siguiente
FinMientras
lista=nuevalista
nodo=lista.primerO
mientras(nodo no sea null)
    Imprimir (nodo.dato.nombre +” Stock “+ nodo.dato.stock)
    nodo=nodo.siguiente
FinMientras
```

Fin

