

UT3_PD3

▼ Property	COMPLETED
📅 Date	@April 30, 2022
☰ BLOCKED	

Crear los elementos y la lista

- LENGUAJE NATURAL

Creo una lista sin elementos

- PRECONDICIONES
 - No tener elementos
- POSTCONDICIONES
 - La lista esta sin elementos
- PSEUDO

Insertar un elemento en una lista (al final, la lista no está ordenada)

- LENGUAJE NATURAL

Reviso si la lista esta vacia, si lo esta la creo y agrego el elemento y lo declaro como el primero. Sino recorro la lista hasta el ultimo nodo, y luego lo inserto.

- PRECONDICIONES
 - La lista existe, este vacia o no.
- POSTCONDICIONES
 - El largo de la lista se incremento en 1
 - La lista se ve inalterada
 - El ultimo elemento efectivamente es el elemento que se deseaba insertar
- PSEUDO

```

insertar(Nodo : cabeza, carga)
  actual <- cabeza
  nuevoNodo <- nuevoNodo(carga,null)
  MIENTRAS actual.siguiete <> null
    actual <- actual.siguiete
  FIN MIENTRAS
  actual.siguiete <- nuevoNodo

```

Buscar un elemento en una lista

- LENGUAJE NATURAL

Reviso si la lista esta vacia, si esta vacia retorno null. Sino recorro la lista revisando la carga y cuando lo encuentro devuelvo el nodo

- PRECONDICIONES
 - La lista existe
- POSTCONDICIONES
 - La lista se mantiene inalterada
 - Se devuelve efectivamente un nodo
- PSEUDO

```

buscarElemento(clave):
  SI esVacia()
    return null
  SINO
    actual = primero
    MIENTRAS actual <> null
      SI actual.getEtiqueta().equals(clave)
        return actual
      actual = actual.siguiete
    return null

```

Listar (imprimir) todos los elementos de una lista

- LENGUAJE NATURAL

Verifico que la lista existe. Si existe recorro todos los elementos de la lista imprimiendo la carga de cada uno. sino existe retorno null

- PRECONDICIONES
 - Existe la lista

- POSTCONDICIONES
 - La lista no se ve alterada
 - Los elementos se imprimen en el orden en que estaba la lista originalmente
- PSEUDO

```
imprimirElementos(Nodo: cabeza)
  actual <- cabeza
  MIENTRAS actual <> null
    imprimir actual.carga
    actual <- actual.siguiente
  FIN MIENTRAS
```

Eliminar un elemento de una lista, dada su clave.

- LENGUAJE NATURAL

Verifico que la lista existe y luego voy buscando consultando el siguiente

- PRECONDICIONES
 - La lista no esta vacia
- POSTCONDICIONES
 - La cantidad de elementos es -1
 - La lista queda correctamente encadenada
- PSEUDO

```
eliminarElemento(etiqueta)
  SI ! esVacia
    SI primero.getEtiqueta().equals(clave)
      primero = primero.getSiguiente();
      return true;
    actual = primero
    MIENTRAS actual.siguiente <> null
      SI actual.getSiguiente().getEtiqueta().equals(clave)
        actual.setSiguiente(actual.getSiguiente().getSiguiente());
        return true;
      actual = actual.siguiente
  return false
```

[CODIGO IMPLEMENTADO EN EL PROYECTO DE JAVA PD3]