

LPC

Cola | **FIFO**: Primero que entra, primero que sale Pila | **LIFO**: El ultimo que entra, el primero que sale

Los getters y los setters se escriben en la medida que sea necesario especificar alguna función con parámetros específicos o sólo se declaren métodos específicos debido a características específicas del problema/ejercicio. Si hay muchos se completan los getters/setters faltantes con "..." (tomar en cuenta que esto se hace en papel para conservar espacio, energía y tiempo).

Clase Nodo

```
Class Nodo<Element>
  Attributes:
    Private:
      Element: info
      pointer to Nodo<Element>: next
  Methods:
    Public:
      proc construir(Element:info, pointer to Nodo<Element>: next)
      //Getters
      ...
      //Setters
      ...
endClass
```

Clase Lista

```
Class Lista<Element>
  Attributes:
    Private:
      pointer to Nodo<Element>: head, tail
      int: long
  Methods:
    Public:
      proc construir()
      proc construir(ref Lista<Element>: target) //Constructor copia
      //Getters
      func getLong():int

      proc insertar(Element: e, int: pos)
      func consultar(int: pos):Element
      proc eliminar(int: pos)
      func buscar(Element: e): int
      proc invertir()
      func copiar(): Lista<Element>
```

```

        proc vaciar()
        func esVacia(): bool
        proc intercambiar(int: pos1, pos2)
        func concatenar(Lista<Element>: target): Lista<Element>
        proc destruir()
        proc modificar(int: pos, Element: e)

endClass

```

Clase Cola

```

Class Cola
  Attributes:
    Private:
      pointer to Nodo<Element>: frente, ultimo
      int: long
  Methods:
    Public:
      proc construir()
      proc construir(ref Cola<Element>: target) //Constructor copia
      func getFrente():Element
      func getUltimo():Element
      func getLong(): int
      proc encolar(Element: e)
      proc desencolar()
      func esVacia(): bool
      proc vaciar()
      proc destruir()

endClass

```

Clase Pila

```

Class Pila
  Attributes:
    Private:
      pointer to Nodo<Element>: tope
      int: long
  Methods:
    Public:
      proc construir()
      proc construir(ref Pila<Element>: target)
      func getTope(): Element
      proc apilar(Element: e)
      proc desapilar()
      func esVacia():bool
      proc vaciar()
      proc destruir()

endClass

```

