

## Laboratorio Nro. 4: Listas Enlazadas

**Johanna Sarai Caicedo Mejia**

Universidad Eafit  
Medellín, Colombia  
jscaicedom@eafit.edu.co

### 3) Simulacro de preguntas de sustentación de Proyectos

#### 1. Test

BlueJ: Probar Resultados

- ✓ LinkedListMauricioTest.testContains (2ms)
- ✓ LinkedListMauricioTest.testInsert (6ms)
- ✓ LinkedListMauricioTest.testRemove (1ms)

Ejecuciones: 3/3      Errores:0      Fallos:0      Tiempo Total: 9ms

2. El problema principalmente es crear un programa que tenga 5 métodos los cuales permite darle órdenes a un brazo robótico simulado para que realice diferentes tipos de acciones, el brazo robótico se simula con una cantidad  $n$  de bloques, estos van desde el 0 hasta el  $n-1$ .

Las acciones que puede realizar este brazo robótico son:

**Mover a onto b:** Donde sabemos que  $a$  y  $b$  son dos bloques específicos, y lo que debe realizar es poner el bloque  $a$  sobre el  $b$ , y devolver los demás a sus posiciones originales.

**Mover a over b:** Acá  $a$  y  $b$  siguen siendo bloques específicos, y realiza un proceso donde pone el bloque  $a$  donde está la pila del bloque  $b$  y los demás vuelven a sus posiciones.

**Pile a onto b:** Este proceso toma  $a$  y  $b$  como bloques específicos y mueve todos los bloques apilados en  $a$ , incluyendo  $a$  hacia encima de  $b$  y los bloques de  $b$  son movidos a su posición inicial antes de apilarse, luego los bloques de  $a$  llegan a su orden inicial.

**Quit:** este es el último método y es la salida del programa, al dar en este finaliza la ejecución.

**PD:**  $A$  y  $B$  no pueden estar en la misma pila, ya que los métodos serían inválidos.

3. Complejidad:

Método	Complejidad
Stack	$O(n)$
Moveaontob	$O(n^4)$
Moveaoverb	$O(n^2)$
Pileaontob	$O(n^2)$
Pileaoverb	$O(n^2)$
Quit	$O(n)$

4.  $n$  y  $m$  son las repeticiones las cuales hace una parte del algoritmo, cuando las repeticiones son iguales, se toma como  $n^2$ ,  $n^3$ , etc, pero cuando son diferentes repeticiones como lo podrían ser ciclos anidados, entonces se toma un iterador diferente, en este caso  $m$ .  $m$  no aparece en la complejidad de ningún algoritmo, ya que siempre era una potencia menor a  $n$ , y por regla del producto la  $m$  desaparecía.

#### 4) Simulacro de Parcial

1. A) `lista.size()`  
B) `lista.add(auxiliar.pop());`
2. A) `auxiliar1.size() > 0, auxiliar2.size()>0`  
B) `personas.offer(edad);`
3.  $O(n^2)$