

## Laboratorio Nro. X: Escribir el Tema del Laboratorio

**Santiago Valencia Arango**

Universidad Eafit

Medellín, Colombia

[svalenciaa@eafit.edu.co](mailto:svalenciaa@eafit.edu.co)

**Andres Almanzar Restrepo**

Universidad Eafit

Medellín, Colombia

[aalmanzarr@eafit.edu.co](mailto:aalmanzarr@eafit.edu.co)

**Santiago Hincapié Murillo**

Universidad Eafit

Medellín, Colombia

[shincapiem@eafit.edu.co](mailto:shincapiem@eafit.edu.co)

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

| 1. ARRAY      | LINKED     |
|---------------|------------|
| 2. $O(n)$     | $O(n)$     |
| 3. $O(n)$     | $O(n)$     |
| 4. $O(n)$     | $O(n^2)$   |
| 5. $O(n^2*m)$ | $O(n^2*m)$ |

2.

3. Punto 2,2

El algoritmo consta de 3 métodos. El main donde se aloja el llamado de los otros dos métodos y una lista enlazada llamado lista. En el método reorganizar, se recibe la lista y la palabra (La palabra ya incluye los caracteres especiales '[' y ']'). iniciamos con un ciclo que recorra caracter por caracter la palabra. Si el caracter es igual a el caracter especial '[' lo que le sigue deberá ir al principio. Por otro lado si el caracter que es igual a ']' lo que le siga deberá ir al final de la palabra. Esto se lleva acabo con el tamaño de la lista o lista.size(); En el caso que el caracter no fue ni '[' ni ']' se debe de seguir común y corriente el recorrido de la palabra. El otro ciclo llamado salida, es el encargado de ir asignando a la lista el nueva palabra. Para hacerlo se asigna una cadena vacía que se llenará y formará la nueva palabra. Recorremos con un ciclo for la lista y a cada posición le vamos asignando el caracter. Por último imprimimos la nueva palabra o palabra modificada.

4.  $O(n)$

5. N o M son las variables que se pueden dar para realizar el cálculo de complejidad, como por ejemplo. Si hay un ciclo, el código se haría N veces hasta su condición de parada


### 4) Simulacro de Parcial

- 1) C) Las dos tiene la misma complejidad asintótica  $O(n)$
- 2) C) La complejidad asintótica en el peor de los casos es  $O(n)$
- 3) A) while (q.size > 1) b) for (int i = 1; i <= num; i++) c) q.add(q.remove()); d) return q.remove();
- 4) a) lista.size b) LISTA.ADD(auxiliar.pop());
- 5) a) auxiliar1. Size() > 0 , auxiliar2.size() > 0 b) Persona.offer(edad);

**DOCENTE MAURICIO TORO BERMÚDEZ**

Teléfono: (+57) (4) 261 95 00 Ext. 9473. Oficina: 19 - 627

Correo: mtorobe@eafit.edu.co

|   |   |                              |
|---|---|------------------------------|
|  | <p style="text-align: center;">UNIVERSIDAD EAFIT<br/>ESCUELA DE INGENIERÍA<br/>DEPARTAMENTO DE INFORMÁTICA Y SISTEMAS</p> | <p>Código: ST245</p>         |
|   |   | <p>Estructura de Datos 1</p> |

6) c) La complejidad asintótica en el peor de los casos es  $O(n^2)$

**5) Lectura recomendada (opcional)**

- a) Título
- b) Ideas principales
- c) Mapa de Conceptos

**6) Trabajo en Equipo y Progreso Gradual (Opcional)**

- a) Actas de reunión
- b) El reporte de cambios en el código
- c) El reporte de cambios del informe de laboratorio