

## Laboratorio Nro. 3

# Listas Enlazadas, Vectores Dinámicos y Tablas de Hash

**Stiven Yepes**  
Universidad Eafit  
Medellín, Colombia  
correoinegrante1@eafit.edu.co

**Sara Rodríguez**  
Universidad Eafit  
Medellín, Colombia  
Correointegrante2@eafit.edu.co

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

#### 3.1 Complejidad de cada ejercicio:

	<b>ArrayList</b>	<b>LinkedList</b>
<b>1.1</b>	<b><math>O(n)</math></b> : Agregar (Worst Case) <b><math>O(n)</math></b> : Consultas	<b><math>O(1)</math></b> : Agregar <b><math>O(n)</math></b> : Consultas
<b>1.3</b>	(Se utilizan pilas y colas) <b><math>O(n,m)</math></b> -n: tamaño de la pila, -m: tamaño de la cola	

¿En qué ejercicios es mejor usar una estructura o la otra? ¿Es alguna de las dos eficiente para este problema o se necesitará otra estructura de datos aún más eficiente?

En un problema donde la posición de los datos sea de mucha importancia es mejor usar un ArrayList o un Array, que maneja posiciones y es mucho más fácil buscar un elemento.

Si la principal operación es agregar y quitar datos el linked-list puede resultar mucho más favorable.

En el ejercicio 1.3 es necesario usar pilas y colas, por la manera en que está planteado el problema, ya que el uso de una sola estructura lo haría muy ineficiente.

#### 3.2

##### Funcionamiento del ejercicio 2.1:

El algoritmo empieza inicializando las variables que se van a utilizar, después entra a un ciclo que analiza cada carácter de la entrada, dando lugar a 2 posibles casos:

## ESTRUCTURA DE DATOS 1

### Código ST0245

El primero es que el carácter sea “[”, para este caso pondrá como verdadero la variable inicio, aumentará el iterador del ciclo (esto para indicar que el carácter que se va a agregar no es el corchete sino el que hay después) e inicializará el contador en 0.

El segundo es que el carácter sea “]”, para este caso pondrá como falso la variable inicio y aumentará el iterador del ciclo.

Luego de esto verificará si la variable inicio es verdadera (si es verdadera quiere decir que se empezó a escribir desde el inicio del texto), en cuyo caso agregará el carácter a la posición correspondiente al valor del contador, y seguidamente aumentará este. De no ser verdadera, solamente agregará el carácter al final.

Después se eliminarán todos los corchetes de la lista y se retornará el resultado.

### 3.3 y 3.4 Complejidad del ejercicio realizado en el numeral 2.1

La complejidad total del algoritmo es  **$O(n)$** , donde  $n$  es la longitud de la cadena de caracteres.

#### 4) Simulacro de Parcial

4.1.1 b) Suponiendo que  $I_1$  y  $I_2$  están ordenadas, una nueva lista con los elementos de ambas listas ordenados.

4.1.2 a)  $O(n \times m)$

4.2 b)  $O(1)$

4.3 b) que inician con la misma letra colisionan

d)  $O(1)$

4.4.1 `stack.push()`

4.4.2 c)  $O(1)$

4.5 a) [7,8,3,1,2,9]

4.6 \* Se asume la llamada a `print` como al método “imprimir”

a)  $O(n^3)$

4.7 - No existe este punto

4.8 c)  $O(n)$

4.9 4.9.1 c)  $O(n \log k)$

4.9.2 b) 9

**PhD. Mauricio Toro Bermúdez**

Docente | Escuela de Ingeniería | Informática y Sistemas  
Correo: mtorobe@eafit.edu.co | Oficina: Bloque 19 – 627  
Tel: (+57) (4) 261 95 00 Ext. 9473



## ESTRUCTURA DE DATOS 1

### Código ST0245

### 4.9.3 (c) $O(1)$

#### 4.10.1. d) $O(n)$

10.2. a) 6

10.3. b)  $O(n)$

4.11.1. c)  $O(\max(\text{list}) \times n^2)$

11.2. b)  $O(n)$

#### 4.12.1. !s1.isEmpty()

## 12.2. s1.pop()

### 12.3. s2.pop()

4.13 .1. iv) 0, 2, 4, 6, 8, 10

13.2. i)  $O(1)$

4.14 i

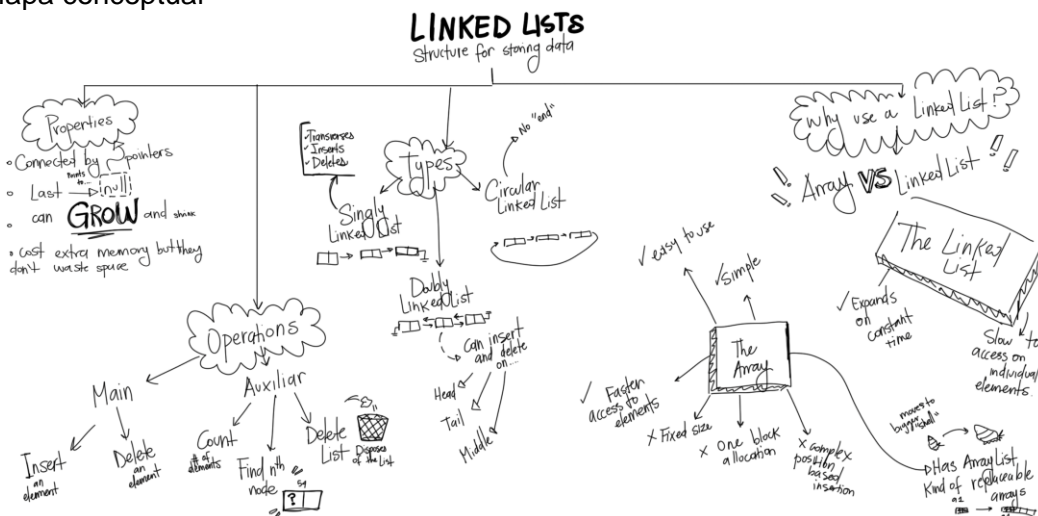
#### 4.14.1 i) $O(n)$

4.14.2 i)  $O(n)$

4.15 iv) 5, 4, 3, 2

**5) Lectura recomendada (opcional)**

## Mapa conceptual



**PhD. Mauricio Toro Bermúdez**

Docente | Escuela de Ingeniería | Informática y Sistemas  
Correo: mtorobe@eafit.edu.co | Oficina: Bloque 19 – 627  
Tel: (+57) (4) 261 95 00 Ext. 9473

**ESTRUCTURA DE DATOS 1**  
**Código ST0245**

Tamaño

Completo:

[https://eafit-my.sharepoint.com/:i:/g/personal/srodriguev\\_eafit\\_edu\\_co/EeFYAt35AohNoNRQC-zDfhIBveHHsPYivk6BWlepWguSwg?e=OwocT8](https://eafit-my.sharepoint.com/:i:/g/personal/srodriguev_eafit_edu_co/EeFYAt35AohNoNRQC-zDfhIBveHHsPYivk6BWlepWguSwg?e=OwocT8)

**PhD. Mauricio Toro Bermúdez**

Docente | Escuela de Ingeniería | Informática y Sistemas  
Correo: mtorobe@eafit.edu.co | Oficina: Bloque 19 – 627  
Tel: (+57) (4) 261 95 00 Ext. 9473

