

Laboratorio Nro III: Listas Enlazadas (Linked List) y Listas Hechas con Arreglos (ArrayList)

Juan Camilo Guerrero Alarcón
Universidad Eafit
Medellín, Colombia
jcguerrera@eafit.edu.co

Santiago Pulgarin Vasquez
Universidad Eafit
Medellín, Colombia
spulgarinv@eafit.edu.co

3) Simulacro de preguntas de sustentación de proyectos

3.1

	ArrayList	LinkedList
Ejercicio 1.1	$O(n^2)$	$O(n)$
Ejercicio 1.2	$O(n)$	$O(n)$
Ejercicio 1.3	$O(n^2)$	$O(n)$
Ejercicio 1.4	$O(n)$	$O(n)$

3.2

El ejercicio 2.1 denominado teclado roto consiste básicamente en encontrar los caracteres especiales “[o]” en caso de que el carácter sea “[“ el programa debe enviar la cadena o carácter al inicio de una lista que este contenida mientras vuelva a encontrar otro carácter de ese tipo, si el carácter especial es ”]” enviara al final de la lista.

La implementación de nuestro algoritmo cuenta con un método denominado ordenar() el cual recibe como parámetro un String, cuando se ingresa al método este cuenta con dos variables booleanas llamadas first y last las cuales indican el estado para proceder a analizar el String, en caso de que first tenga verdadero como estado quiere decir que encontró el primer carácter especial y empezara a añadir caracteres al inicio de la lista hasta encontrar el otro carácter, que puede volver a ser el primero o el segundo, en caso de que el segundo sea el que sigue solo se encargara de añadir los caracteres al final de dicha lista.

3.3

Keyboard

$$T(n) = (n) + c + c1 + c2 + c3 + c4n + c5 + c6 + 1 + 1$$

$$T(n) \text{ es } O((n) + c + c1 + c2 + c3 + c4n + c5 + c6 + 1 + 1)$$

Por definición de O. $T(n)$ es $O((n) + c)$ Por regla de la suma.

$T(n)$ es $O(n)$ Por regla de la suma.

3.4

En el calculo de complejidad anterior, la variable str que es de tipo String, es la cadena que se va a ordenar. El tiempo de ejecucion del algoritmo depende del tamaño de la cadena ingresada.

4) Simulacro de parcial en el informe PDF

- 1). A
- 2). C
- 3). while(q.size() < 2)
 i <= num
 q.add(i)
 return q.get[0]
- 4). lista.size()
 lista.add(auxiliar.pop())
- 5). auxiliar1.size() > 0
 auxiliar2.size() > 0
 personas.offer(edad)
- 6). O(n^2)
- 7). O(n^3)
- 8). O(1)
- 9). O(K)
 12
 O(1)
- 10). O(n)
 6
 O(n)