

Laboratorio Nro. 3

Listas Enlazadas (*Linked List*) y Listas Hechas con Arreglos (*Array List*)

Simón Marín Giraldo
Universidad Eafit
Medellín, Colombia
smaring1@eafit.edu.co

Miguel Fernando Ramos García
Universidad Eafit
Medellín, Colombia
mframesg@eafit.edu.co

3) Simulacro de preguntas de sustentación de Proyectos

3.1

LinkedList	ArrayList
$O(n^2)$	$O(n)$

3.2 En este ejercicio nos plantean la situación de un teclado que internamente tiene un problema. Presiona las teclas inicio y fin sin que el usuario lo indique. Para este ejercicio, la tecla inicio se representa con el corchete que abre “[” y la tecla fin se representa con el corchete que cierra “]”. Para cumplir con el objetivo, primero que todo, se debe declarar una LinkedList, dado que esta permite agregar elementos en cualquier posición. En particular, nos interesa agregar al inicio y al final de la lista. Cuando se presione la tecla inicio, se agregará al comienzo de la lista, lógicamente. De la misma forma, al presionarse la tecla fin, se agregará al final de la lista. Especialmente nos interesa encontrar sub cadenas de texto entre un corchete y otro, dado que se tienen que tomar completas para agregarlas como debe ser. Entonces, posterior a la creación de la lista, creamos un booleano alFinal que se encarga de establecer un modo de operación. Cuando es verdadero, estará agregando al final de la lista. De lo contrario lo hará al inicio de la lista. Luego, inicializamos la variable k, que será la posición a la izquierda del substring, que irá cambiando para formar las palabras o substrings entre un corchete y otro. Luego hacemos un for desde 0 hasta la longitud de la cadena de texto que se le pase como parámetro a la función. En este for, inicialmente se busca si el caracter en la posición es el corchete que abre “[”. Si se cumple esto, se procede con los pasos para tomar el substring. Se pregunta si el modo de operación es para agregar al final de la lista. Si es verdadero, se procede a agregar desde la posición por la izquierda hasta la posición correspondiente al valor que tenga i. Si esta

ESTRUCTURA DE DATOS 1

Código ST0245

pregunta tiene respuesta falsa, entonces se agregará al inicio. Para cerrar este condicional, se aumenta el valor por la izquierda en 1 y lo que tenemos por derecha. También se establece modo de operación para que agregue por el inicio. Luego de todo esto, se hace la misma operación, pero con el corchete de cierre “]”. Termina el ciclo. Se procede a preguntar si el modo de operación está establecido para operar en el final de la lista. Si es así, se agrega al final de la lista lo que se tenga desde la posición k. De lo contrario, se procede a agregar, pero al inicio de la lista. Finalmente, se inicializa un String que se retornará como resultado de todo. Este String se llenará con los valores en cada posición de la lista. Por último se devuelve el resultado.

3.3 La complejidad del punto 2.1 es $O(n^2)$

3.4 Es nuestro caso al hacer el método brokenKeyBoard usamos dos ciclos anidados for y

4) Simulacro de Parcial

4.1 c

4.2 b

4.3 **4.3.1** q.size() > 1 **4.3.2** <= **4.3.3** q.remove() **4.3.4** q.remove()

4.4 **4.4.1** lista.size() **4.4.2** lista.addLast(auxiliar.pop())

4.5 **4.5.1** auxiliar1.size() > 0 **4.5.2** auxiliar2.size() > 0 **4.5.3** personas.offer(edad)

4.6 a

4.7 c

4.8 **4.8.1** a **4.8.2** c **4.8.3** c

4.9 **4.9.1** d **4.9.2** a **4.9.3** b

4.10 **4.10.1** b **4.10.2** b

4.11 **4.11.1** s1.size() > 1 **4.11.2** s1.pop() **4.11.3** s2

4.12 **4.12.1** iv **4.12.2** ii

4.13 **4.13.1** i **4.13.2** i

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