Laboratory practice No. 3: LinkedLists and Lists Made with Arrays (ArrayList)

Verónica Mendoza Iguarán
Universidad Eafit
Medellín, Colombia
vmendozai@eafit.edu.co

Pablo Alberto Osorio Marulanda
Universidad Eafit
Medellín, Colombia
paosoriom@eafit.edu.co

3) Practice for final project defense presentation

1. In this case, we just going to check the complexity for the exercise number 1.2

ArrayList	LinkedList
O(n^2)	O(n^3)

For this special case, it is better to use the ArrayList than LinkedList. This is just for a the method .get() included throught the algorithm

2. Firstable, the algorithm calls the a method named *linkedMethod*, then, this method starts a LinkedList named *copy*(it will be useful later),it starts two boolean variables which represent the end and the start of the string and it starts a integer that represent a count too. The two boolean variables will serve us to determinate if the position which we are visiting with the loop is the start or it is the end of the principal String. So, the string will be search, letter by letter , in the linked list before named. After all this, the algorithm begins a loop which have the porpouse to sort all the string letters in the correct places. In that way, the loop checks if the first character is a "[" or a "]". If the first case is correct, it means that all the next letters will be in the beginning of the principal word, so the boolean variable changes to true, the end variable change to false and the count is equal to zero(because is the start of something). and, if the second case is true, we just need to say that the boolean variable *start* is false and the boolen variable *end* is true. Then, if boolean variable *start* is true, to all the cases when this variable will keep on true, we organize the letters at the start of the array, keeping the integer count running.if the boolean variable end is true, we just put the letter at the end of the linkedlist. Before the loop ends, we just remove the first letter and sum plus 1 to the count variable.Then we check if the linked list doesn't have ""[" or"]". Thus we just return the array with the correct organization.
3. 2.1: The complexity in the worst case is O(n)
4. 2.1: The "n" variable mens the total positions in the String,that is, the number of character in the String

4) Practice for midterms

1. *a*
2. c
3.
   a) q.size() > 1
   b) <=
   c) q.getLast()
   d) q.remove()
4.
   a) lista.size() > 0
   b) lista.add(auxiliar.pop())
5.
   a) auxiliar1.size() > 0 -- auxiliar2.size() > 0
   b) personas.offer(edad)
6. c
7. c
8. d
9.
   9.1) a
   9.2) c
   9.3) c
10.
    10.1) d
    10.2) a
    10.3) b

6) Team work and gradual progress (optional)

Member	Date	Done	Doing	To do
Pablo	26/09/2018 Start time: 2:30pm	· Implementation of methods that simulate a linkedList simply linked	· Creation of a method to place a pivot in the optimal position	· Improve the linkedList

PROFESSOR MAURICIO TORO BERMÚDEZ
Phone: (+57) (4) 261 95 00 Ext. 9473. Office: 19 - 627
E-mail: mtorobe@eafit.edu.co

Verónica	26/09/2018 Start time: 2:30pm	· Implementation of methods thet simulate a linkedList simply and doubly linked (methods de SmartInsert, add, remove and contains)		· Improve the linkedList
Pablo	28/09/2018 Start time: 12:30pm	· Completion of method to place a pivot	· Method creation for a broken keyboard	
Verónica	28/09/2018 Start time: 12:30pm		· Exercises of practice for midterms	· Exercise 5 of the partial simulation is missing
Pablo	02/10/2018 Start time: 3pm	· Refrigerator storage program implementation · Continuation of method for a broken keyboard		
Verónica	02/10/2018 Start time: 3pm	· Continuation practice for midterms · Analysis of the methods and calculation of their complexity	· development of optional exercises	

Pablo	03/10/2018 Start time: 3pm	· Verification of Exercises of practice for midterms · Explication of the implementation of exercise 2.1 and calculation of complexity	· optional exercises	
Verónica	03/10/2018 Start time: 3pm	· Revision of the behavior of the methods with Arraylist and with linkedList		
Pablo	07/10/2018 Start time: 3pm	· Explanation of the variables of the complexity calculation	· Compilation of all exercises and improvements	
Verónica	07/10/2018 Start time: 3pm	· Explanation of the variables of the complexity calculation	· Compilation of all exercises and improvements	