

Joshua Johnson
Prof. Cheryl Resch
COP3530
18 September 2018

Programming Assignment 1: Line Editor Commentary

What is the computational complexity of the methods in the implementation?

<i>print</i>	- $O(n)$	<i>insertEnd</i>	- $O(n)$
<i>insertAt</i>	- $O(n)$	<i>edit</i>	- $O(n)$
<i>removeAt</i>	- $O(n)$	<i>search</i>	- $O(n)$
<i>LinkedList()</i>	- $O(1)$		

Your thoughts on the use of linked lists for implementing a line editor. What are the advantages and disadvantages?

Advantages

The list can grow as needed, as opposed to an array based implementation, in which a new array would need to be created if the space fills up.

Neutral

Adding and removing within a linked list is constant time if the iterator is where we are within the list. However, this likely will not be the case, since with a line editor, there is no cursor to keep track of where you are within a document. You must specify the location of each line you would like to add or remove and iteratively move to that location. Time complexity will be linear, similar to an array based implementation.

Disadvantages

Accessing elements within the list is a linear time operation as opposed to constant time with an array based implementation.

Linked list elements must store a pointer along with the data (small storage disadvantage).

What did you learn from this assignment and what would you do differently if you had to start over?

I learned more about string functionality (substring, npos, stoi, find). I created my first singly linked list using a doubly linked list that I coded as reference. I learned how to efficiently iterate through the list.

The line editor class seemed unnecessary since it just acts as a middle man and calls the linked list functions. If I started over on this assignment, I would have put this functionality in main(). I also would have started coding in an IDE sooner. While coding the first few functions, I was running my code through Stepik. The compiling time was slow, so I switched over to my IDE and was able to cut down my testing time dramatically.