Scott Krstyen

U0760822

CS2420

October 19, 2016

Analysis Document: Assignment 7

1. I have not yet worked with more than one partner, but I am working with a new partner on the next assignment.

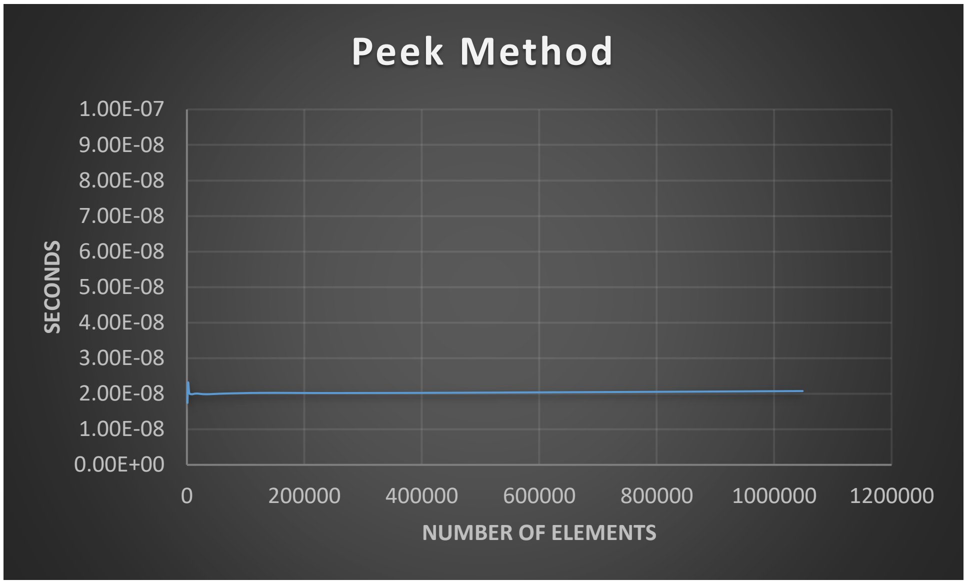
2. It would slightly be better to use a singly linked list instead of a doubly linked list because to can do the same thing as a doubly linked list without needed the references pointing backwards. The only caveat is that if you use a singly linked list, you have to add and remove from the front of the list. If you try to make the end the "top" of the stack, deleting will take O(N) time. A doubly linked list can have either the front and the end be the "top" of the stack with O(c) time. So I would say as long as you make the front the "top" of the stack, it would be slightly better to use a singly linked list because then you don't need the backwards references that you have in a doubly linked list.

3. Yes it is possible to replace my instance of Doubly Linked List with Java's Linked List, because it is a Doubly Linked List so it contains all the methods and structure you need to use it to back our LinkedListStack.

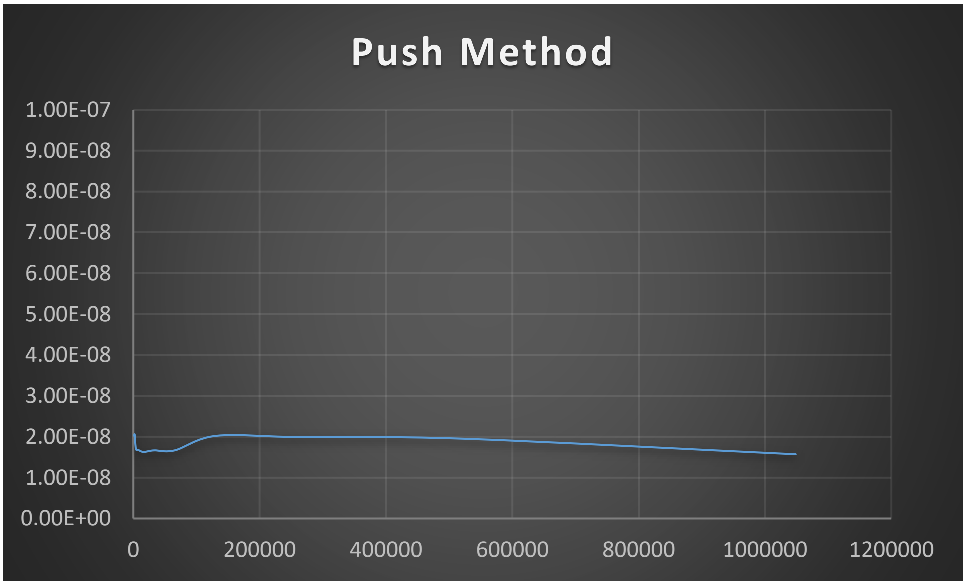
4. Because the LinkedListStack was backed with the Doubly Linked List class I made it was very easy to develop the LinkedListStack class because I could use methods from my Doubly Linked List class to fulfill the methods in my LinkedListStack class. So the LinkedListStack class was easier to develop. On this assignment I spent most of the time working on the BalancedSymbolChecker class.

5. You could always use variables to keep track of which column and row each opening symbol was found but that could get a little messy with many repeating symbols. The way I would do it is to create a new class that you can create objects from that can hold the "data" (symbol) and also the column and line number where it was found. You can create a LinkedListStack of the type of this class and use these special "objects" to store the original position of the data along with the symbol. This would ensure that you have the correct symbol with the correct line and column number.

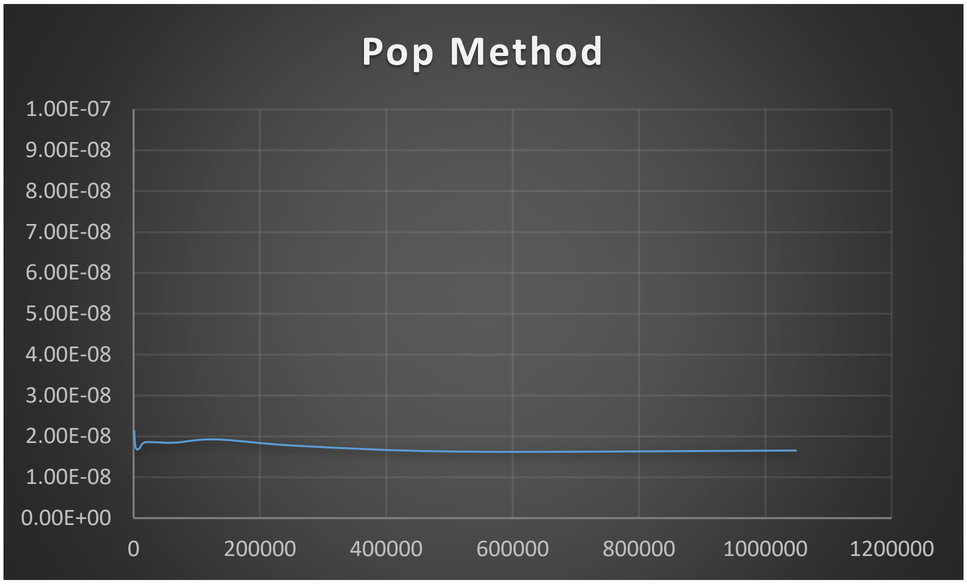
6.



As you can see above the peek method is O(c). That means as N grows the running time does not grow. There are little squiggles in the line but this is 2.00E^(-8) so this is an extremely small amount of time, so the those are probably due to the computer running other processes at the same time so it runs slightly longer or slower. This peek method is constant whether there are 1024 elements or over a million elements.



As you can see above the push method is O(c). That means as N grows the running time does not grow. There are little squiggles in the line but this is 2.00E^(-8) so this is an extremely small amount of time, so the those are probably due to the computer running other processes at the same time so it runs slightly longer or slower. This push method is constant whether there are 1024 elements or over a million elements.



As you can see above the pop method is O(c). That means as N grows the running time does not grow. There are little squiggles in the line but this is 2.00E^(-8) so this is an extremely small amount of time, so the those are probably due to the computer running other processes at the same time so it runs slightly longer or slower. This pop method is constant whether there are 1024 elements or over a million elements.

7. I spent about 11 or 12 hours on this assignment.