Anthony Iovino
Analysis Document

1.
No, however I currently have a new partner for the next assignment.

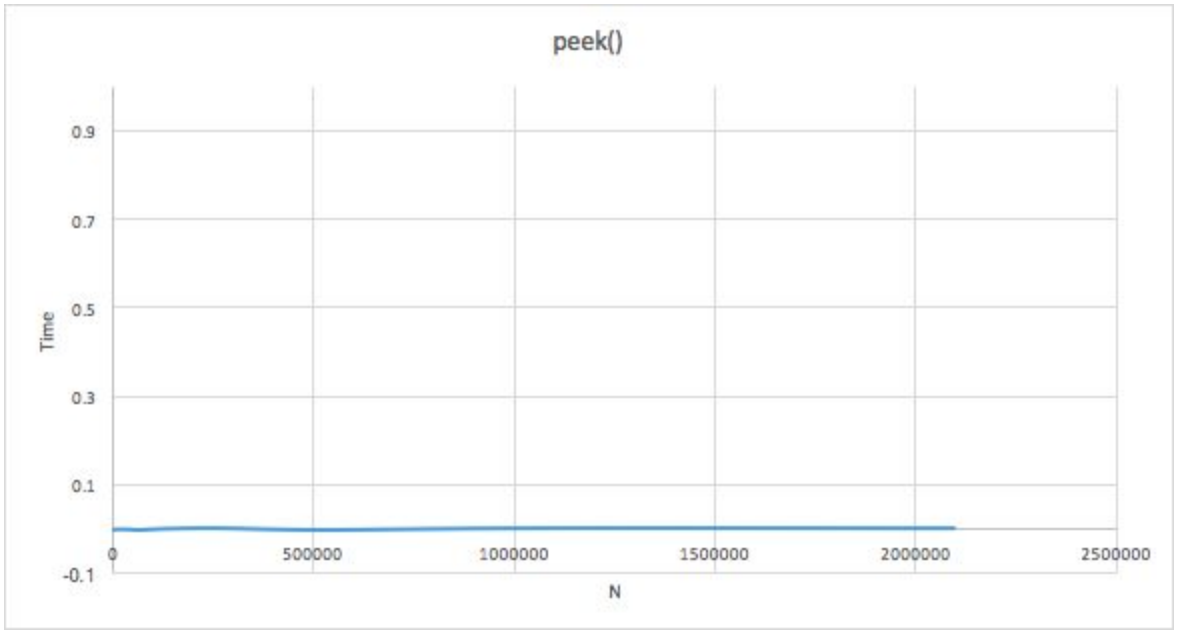
2.
Both the get last and the add last methods could have been done using as singly linked list. The only method that uses the unique functionality of the doubly linked list is the pop() method which uses the removeLast() method. Similar performance could be achieved with a reference to the 2nd to last node in a singly linked list. The singly linked list would use an overall smaller memory footprint because it ultimately stores less information per node. This would arguably make it better.

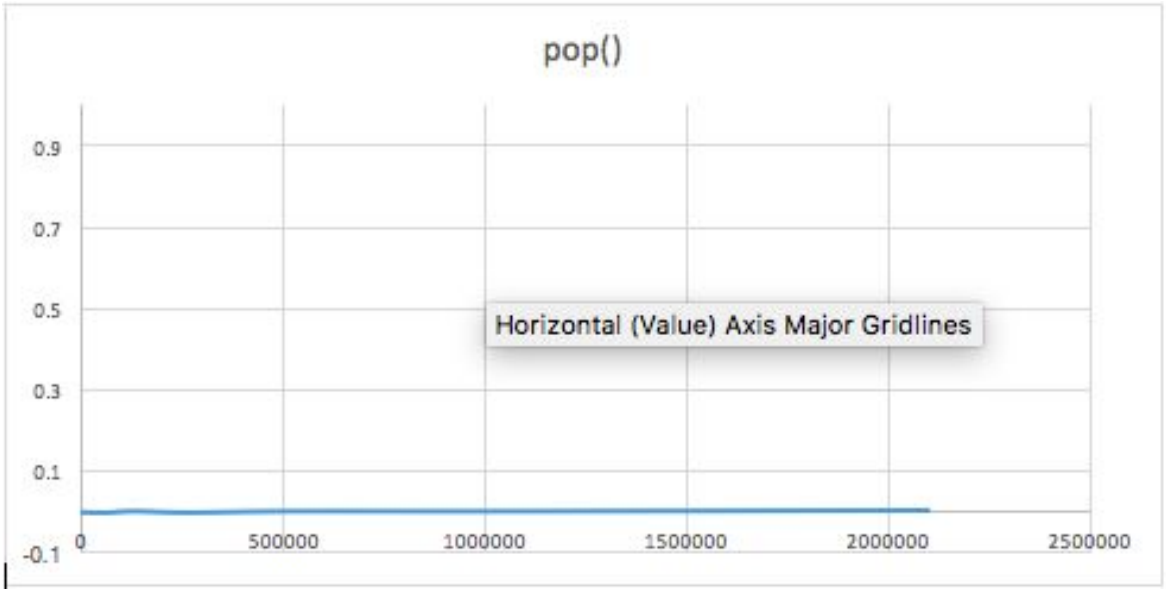
3.
Yes, it would be possible. The top of the stack in my implementation is
at the end of the linked list. The top of the stack in Java's implementation
is at the front of the linked list. Therefore, the implementation of LinkedListStack
would have to change with respect to that difference.

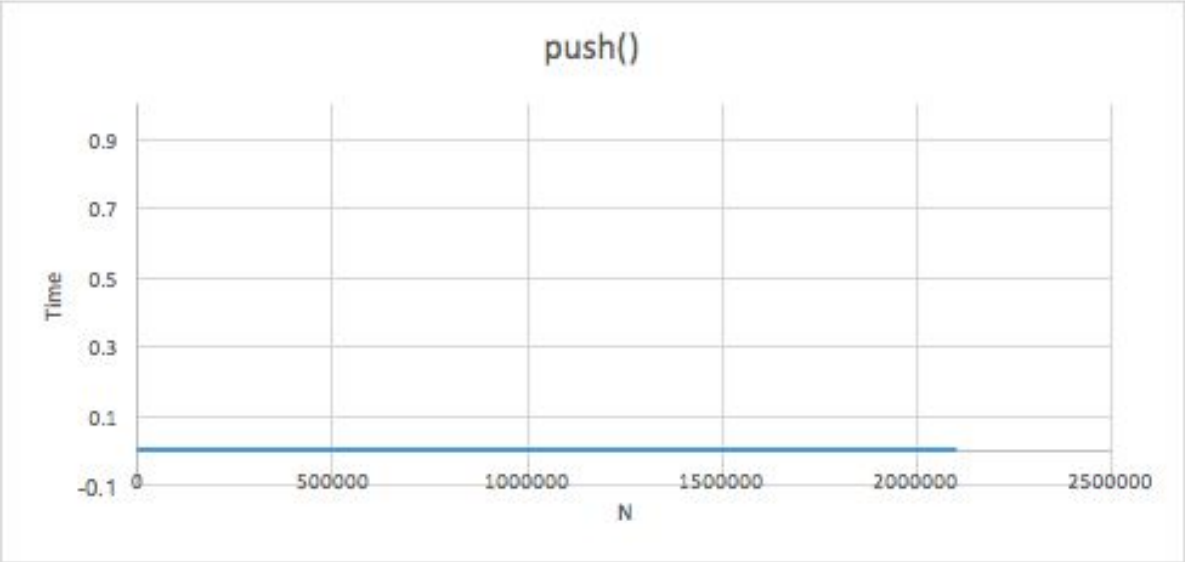
4.
Thanks to the prewritten DoublyLinkedList class the time spent developing the LinkedListStack class was very efficient.

5.
You could keep track of the location of the unmatched opening symbol by storing its row and column number on the stack with the character or possibly on a different stack relative to the existing stack. When the error is found you could then access the original location of the unmatched opening symbol.

6.


peek()
0.9
0.7
0.5
0.3
0.1
-0.1
Time
0
500000
1000000
1500000
2000000
2500000
N


pop()
0.9
0.7
0.5
0.3
0.1
-0.1
Horizontal (Value) Axis Major Gridlines
0
500000
1000000
1500000
2000000
2500000



Yes, all are constant time as expected.

7. How many hours did you spend on this assignment?

~ 10 hours