1. Who is your programming partner? Which of you submitted the source code of your program?
   My programming partner was Aaron Smith, he submitted our code.

2. Have you worked with more than one partner yet? Remember, you are required to switch at least once this semester.
   No I have not, I plan on switching after spring break.

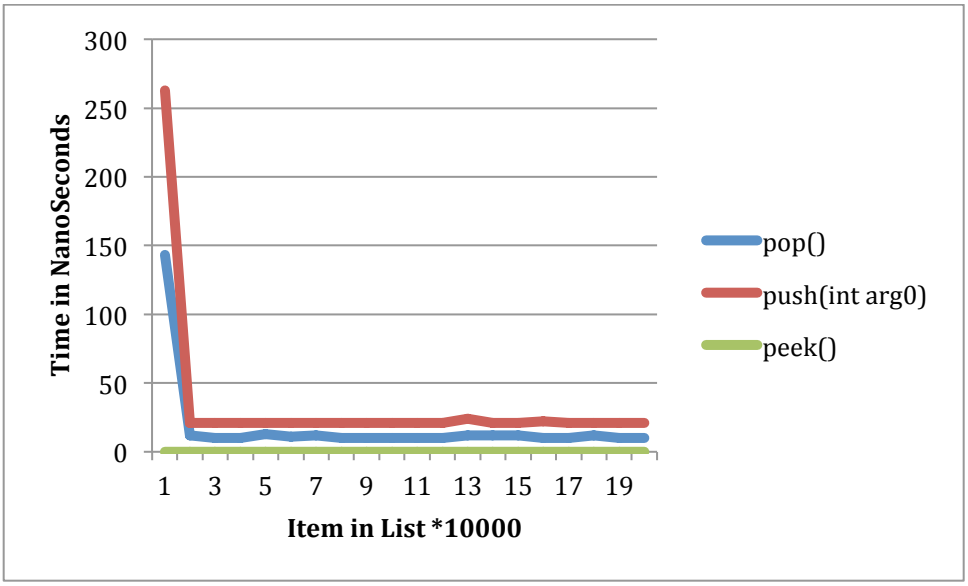
3. In the `MyStack` class, the stack data structure is implemented using a doubly-linked list. Would it be better to use a singly-linked list instead? Defend your answer.
   I believe that it is best to use a doubly-linked list. This is because when you remove an item from the stack you must make the item before the tail the new tail, in order to iterate to this in a singly-linked list you must iterate through all the items before it, in a doubly-linked list you just have to call tail.prev.

4. Would it be possible to replace the instance of `MyLinkedList` in the `MyStack` class with an instance of Java's `LinkedList`? Why or why not?
   Yes, LinkedList contains all the methods that we call in our MyStack class so we could replace our MyLinkedList class with Java's built in LinkedList.

5. Comment on the efficiency of your time spent developing the `MyStack` class.
   We were very effective at developing our Mystack class. We already had the methods necessary from our last assignment coded so that we could very easily implement the Stack method, which is like a LinkedList but with more limitations.

6. Note that the line and column number given by `BalancedSymbolChecker` indicate the location in a file where an unmatched symbol is detected (i.e., where the closing symbol is expected). Explain how you would also keep track of the line and column number of the unmatched opening symbol. For example, in Class1.java, the unmatched symbol is detected at line 6 and column 1, but the original '(' is located at line 2 and column 24.
   You could create objects that contain the character read, and the original location, then add these to the stack. When one of the object does not have it opposing character then you can read the openings character and read it location.

7. Collect and plot running times in order to determine if the running times of the `MyStack` methods `push`, `pop`, and`peek` are *O*(1) as expected.




All of our plots appear to show constant growth, non of them seem to increase at all over time regardless of the input size.

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

We spent about 5 hours on this assignment.