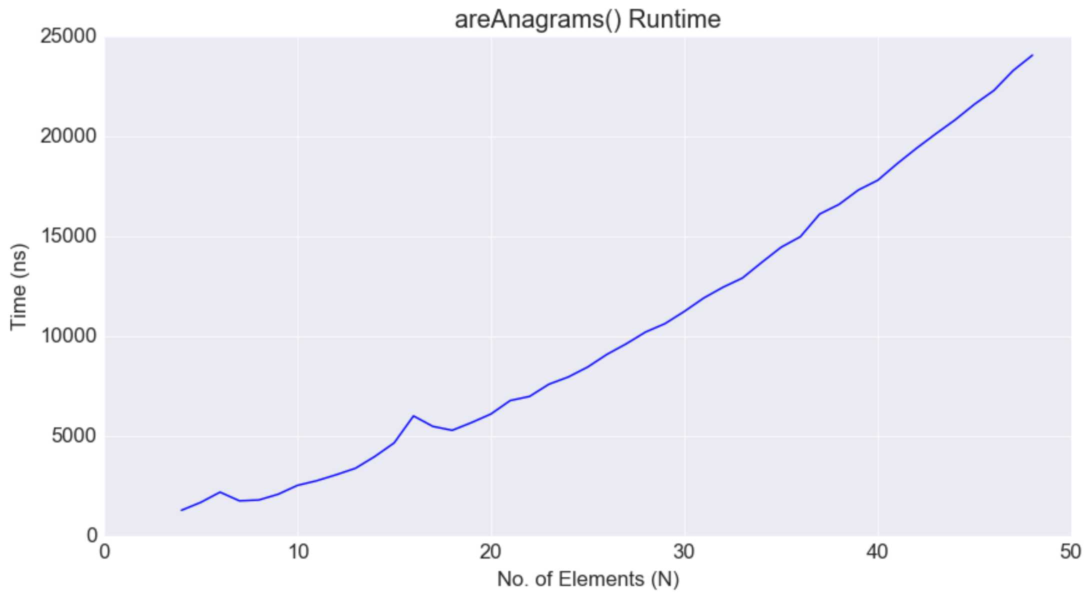
Lingxi Zhong
U0770136

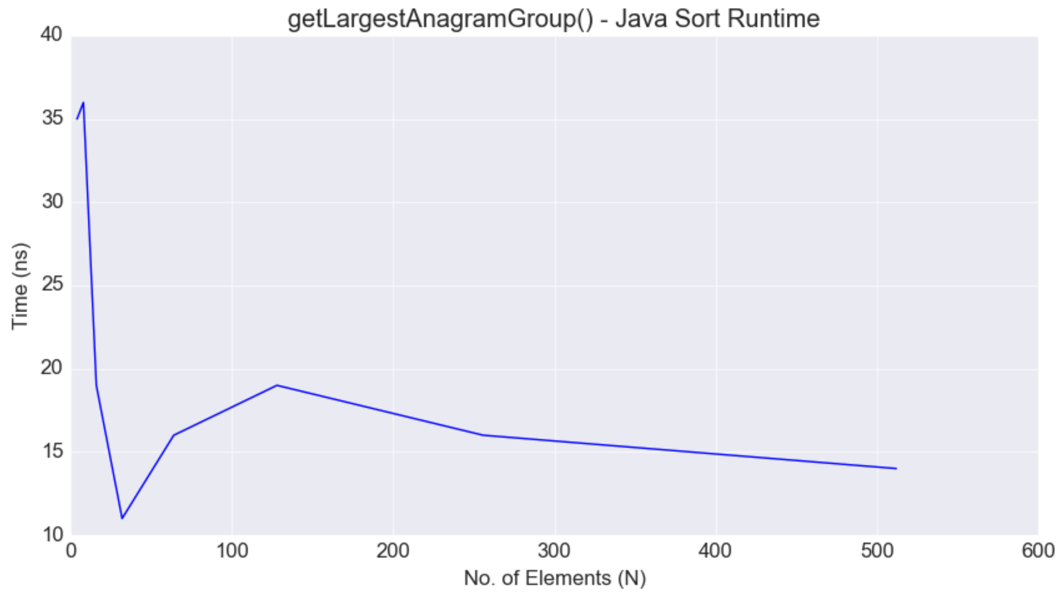
1. Rahul Ramkumar was my partner, he submitted our source code
2. I learned a lot from my partner, especially about running timing tests. I learned to run them more efficiently and effect ways to code timing tests. I think he learned more about running tests on our code from me.
3. My programming partner was great, he was very on top of the due date and started the assignment early. He was well versed in java and overall very enjoyable to work with.
4. The Big O Behavior of areAnagrams should I think be the same as that of an insertion sort, since that is the primary part of what areAnagrams is doing: performing an insertion sort. Therefore I believe it is ON^2 with N being the length of the word.



We tried our best with the timing and this is the closest we could get to O(N^2). Not sure if it's a timing error or if the parameters of time need to be widened but we're at least getting some sort of growth. So our graphs did not exactly match our predicted.

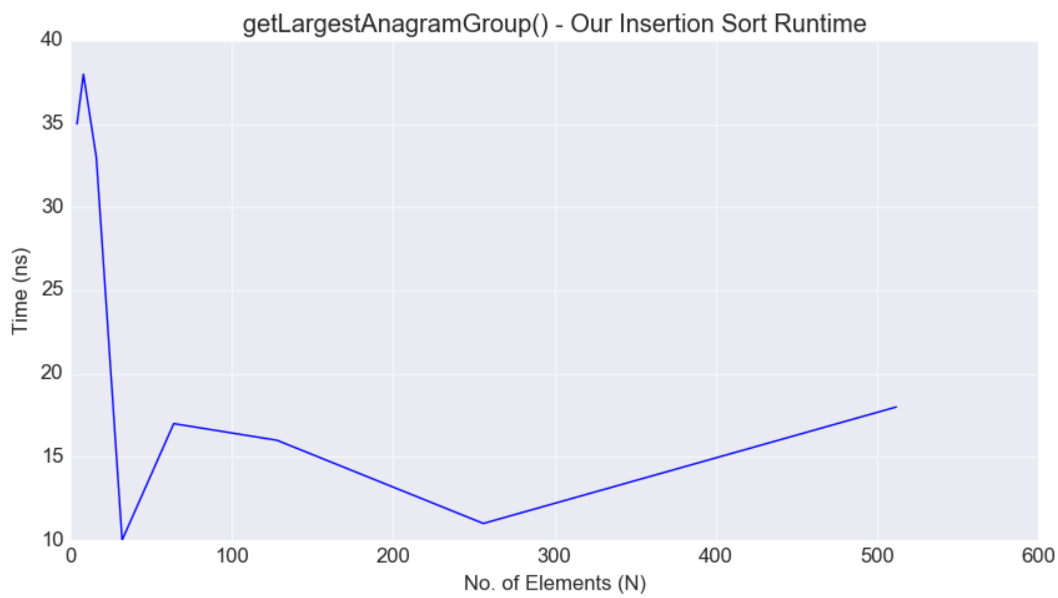
5. The getLargestAnagramGroup method should have the Big O behavior of a O(N^3) method since it has to do an insertion sort for the items but also search and find the largest single group of Anagrams to return, which is O(n), so when you put them together you get O(N^3).

Lingxi Zhong
U0770136



The growth rate of the plotted running times here do not match our predicted algorithm, likely some kind of timing error that we couldn't figure out

6. The run time performance of getLargestAnagramGroup if we use Java's sort method instead should be more efficient than ours since the built in Java method uses a Dual Pivot Quicksort, which already sounds fancier, but on top of the fancy name offers an apparent O(n log(n)) performance, which is more efficient than our O(N^2) insertion sort.



This graph is a little closer to our predicted then our #5 graph but still a little off. There seems to be a lot of noise in our timing tests that we'll need to figure out for future tests.

Lingxi Zhong
U0770136

7. I spent 8 hours on this assignment