CS2420 Assignment04 Analysis

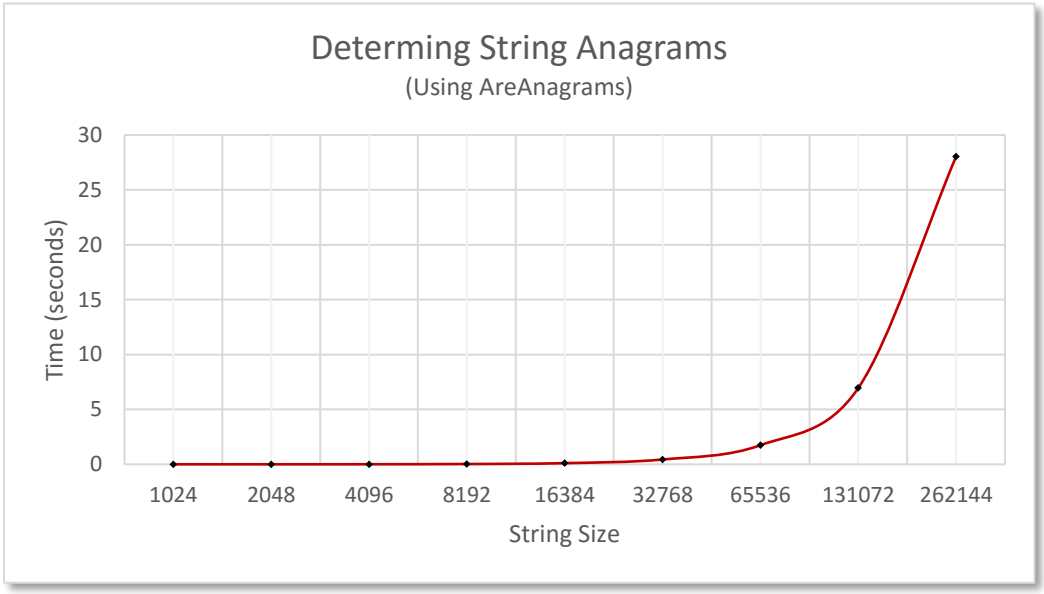
ANDREW WORLEY / U0651238
University of Utah

1. Nicholas Kerr was my partner in this assignment, and I submitted the source code.

2. Nicholas kept well organized notes and made use of pseudocode more than myself. This taught me to slow down and start on paper before diving into a problem. Nicholas had some questions regarding the Comparable interface and use of a comparator. I was able to help him obtain a clearer understanding.

3. Nicholas was willing to spend the time to write and understand the methods in this assignment. However, I plan to change partners. I felt his understanding of the coursework required review that lengthened the assignment completion time above what my schedule can currently support.

4. The areAnagrams method makes two calls to the sort method (one for each string passed in). For the following, N will be the size of a string. The sort method has a complexity of $O(N^2)$ for an average/worst case. Therefore, we predict that areAnagrams will have a complexity of $(f(N)=(1+1)N^2)$ which results $O(N^2)$.

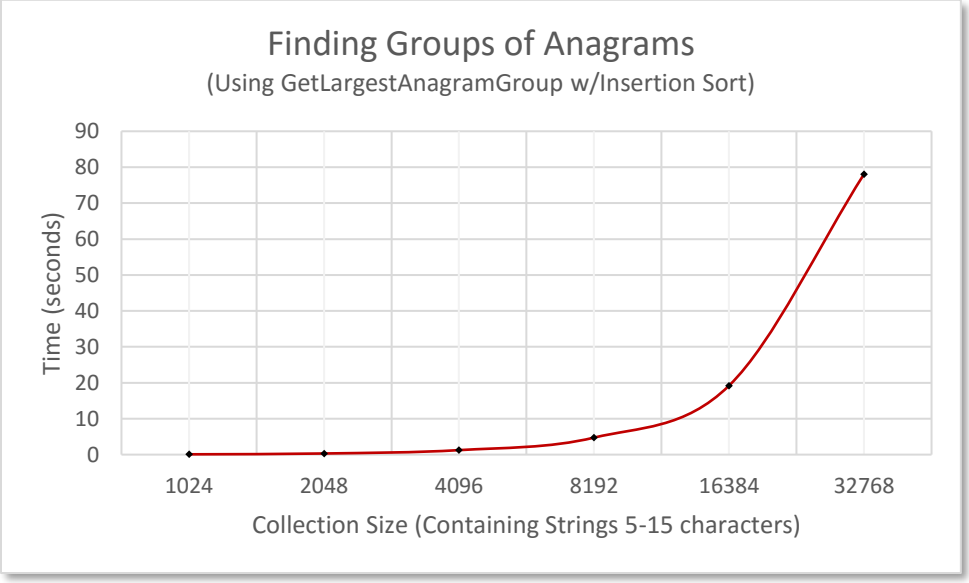


Upon analyzing the graph, using insertion sort for real world words will be effective, and areAnagrams matches our predicted behavior.

Note: the word for the chemical “Methionylthreonylthreonylglutaminylarginyl...isoleucine” is 189,819 characters long but such word would be out of scope for the specifications to create an anagram utility used in a puzzle game.

CS2420 Assignment04 Analysis

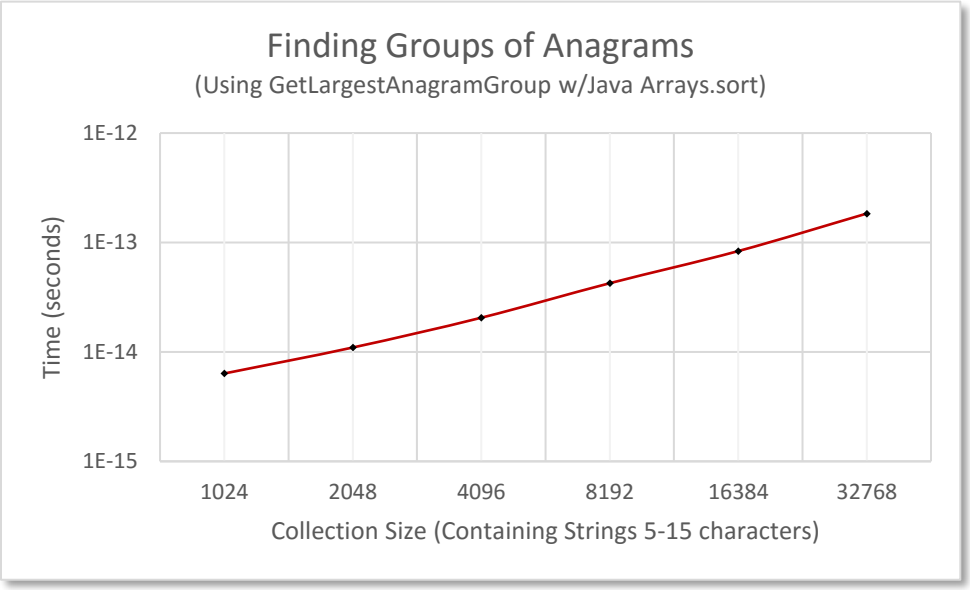
5. For analyzing the getLargestAnagramGroup method N is the size of the string collection in which getLargestAnagramGroup will sort and check for anagram groups. Each string in a test collection is a random string with a size between 5 and 15. All test collections are stored in a list prior to testing so that each iteration per test sorts the same data. We predict that getLargestAnagramGroup will have a complexity of $O(N^2)$. This is due to the asymptotic behavior of the function, and the string sizes to be sorted are so small that their affect is negligible. Also, while finding groups of anagrams the collection is only iterated once.



As predicted the getLargestAnagramGroup using insertion sort models $O(N^2)$ complexity.

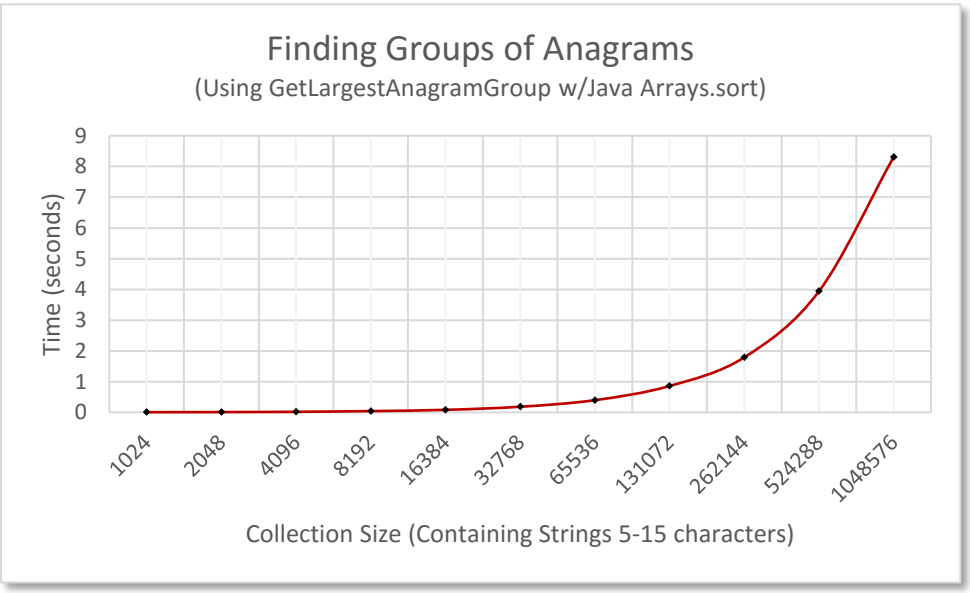
CS2420 Assignment04 Analysis

6. When comparing the getLargestAnagramGroup using Java's Arrays.sort the same test criteria was established as used previously for timing using insertion sort.



Using Java's Arrays.sort method would be preferred to insertion sort for the anagram utility. There are approx. 1,025,109 in the English language and this implementation can sort those items in 8 seconds (see 6.aside.1). The documentation states the Arrays.sort is a modified TimSort which has an average case complexity of O(NlogN), with N being the size of a collection. The above graph represents O(NlogN).

6.Aside.1



7. 10-15 hours were spent on this assignment.