Brian Burton
02/18/17
CS 2420

Analysis Document

Shell, Merge, Java, and Quick Sort Sorted Analysis

Shell, Merge, Java, and Quick Sort Reverse Analysis

Shell, Merge, Java, and Quick Sort Random Analysis
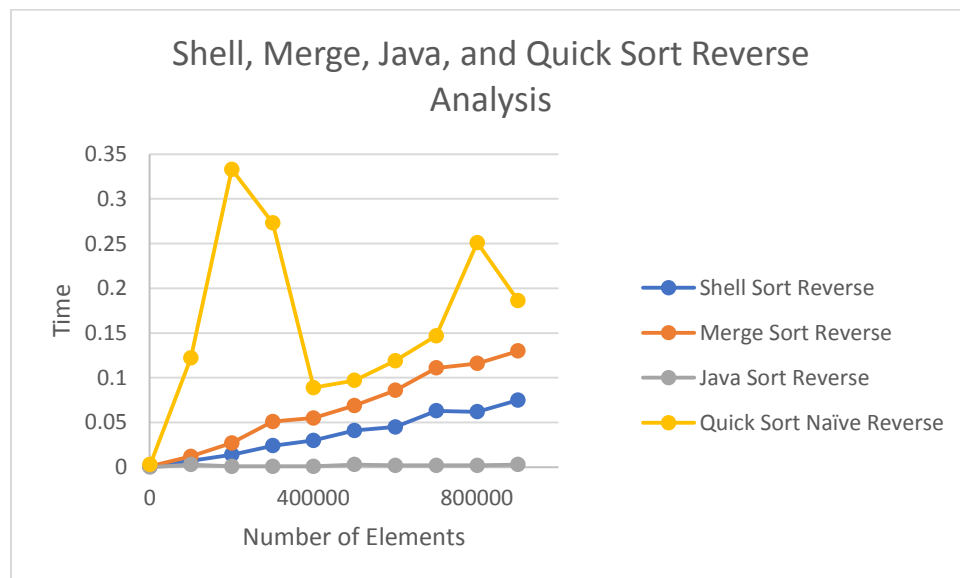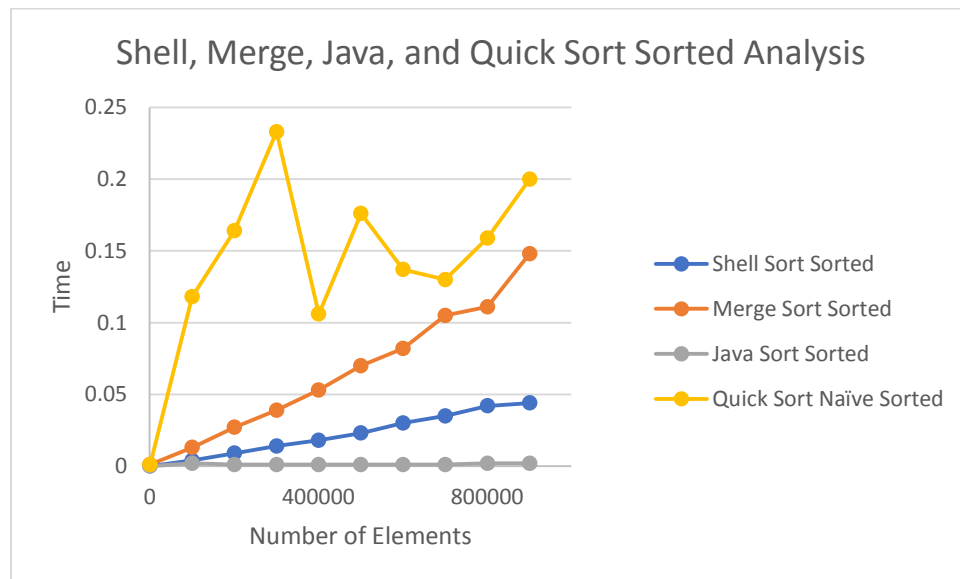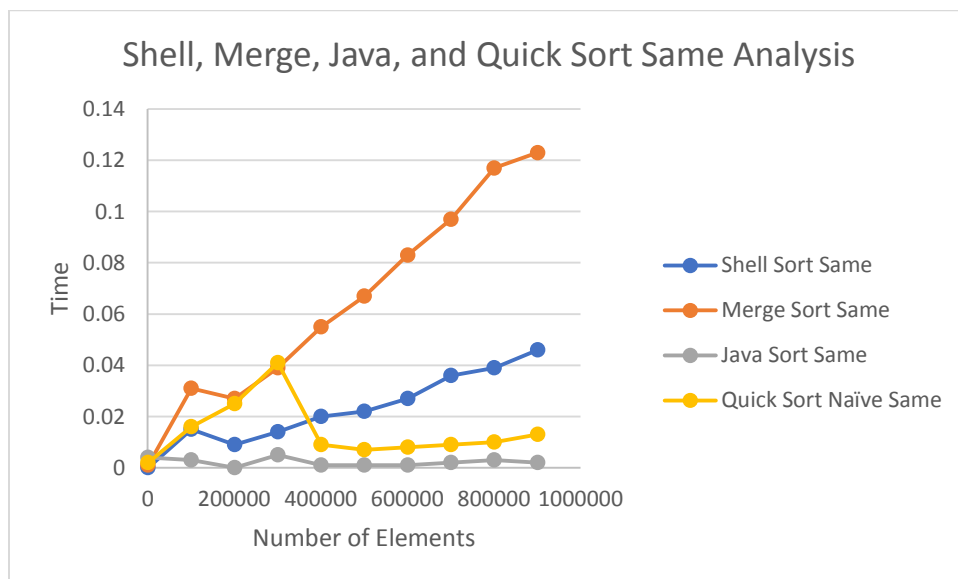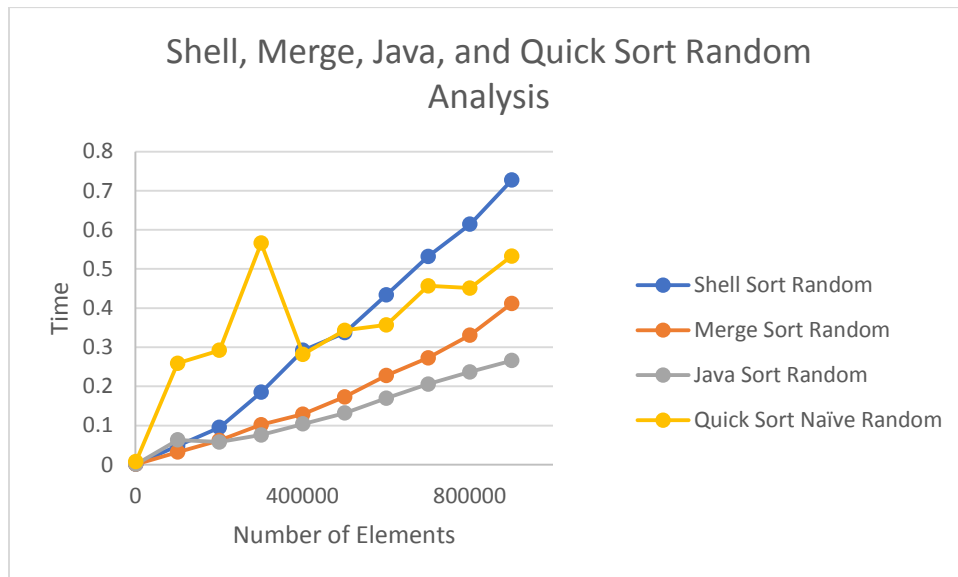

Shell, Merge, Java, and Quick Sort Same Analysis

The two flaws in the assumption that the best partition is the middle element or the average value of all the elements in the array are as follows. First of all, we do not know what is in the array. There could be a lot of duplicate values, large numbers, or small numbers. Any of these scenarios can effect how much time the algorithm will take to complete it's task.

The flaw for averaging the elements is the fact that there could be a lot of zeros, duplicate values, large numbers, or small numbers in the array. The reasoning for this is very similar to what has already been said. While these ideas are not perfect, I believe that they are perfect for the average case array.

Fisher-Yates shuffle uses the termination of zero to i in stead of zero to N because i is a better way to communicate that you are using the index's, not the elements in the array. It also helps communicate the simple fact that it is a shuffle algorithm and not a sorting algorithm.

What I learned from this assignment is how to be patient, debug my code, and how to better analysis how the code is either helping my algorithm speed up or slow down. It was amazing how just one line of code could have so much of a difference in the performance. Understanding how the algorithm worked was also an important thing that I learned in this assignment. It was also very interesting to see how each algorithm had it's ups and downs. None of them are perfect but they can all perform well in their unique circumstances.