Drew Rife

Dr. Armstrong

Algorithms Sorting Project

April 02, 2017

**Algorithms Project Report**

In this project, the goal was to use QuickSort, MergeSort, and HeapSort. Overall I found this project pretty challenging, especially quicksort. It took me awhile to understand the 3-Way partitioning algorithm and it basically took the majority of my time doing this project. I don’t know why but the other implementations I felt were rather simple to implement and did not require me to spend a lot of hours trying to get it to work. I made a bash script to generate the number files and then used java to do the sorting implementations. I have provided JUnit tests to test prove that my sorts work. The program when finished with all the sorts, prints the data to a results.CSV file located within Sorter/Results/results.csv where it includes the File, the Sorting Algorithm and the time duration it took to complete the sort.

Looking at the sorting statics, it seems that most of all had good averages and standard deviations. I found that merge sort was my fastest algorithm, working far faster than the other two algorithms. I have no idea why it would be faster, as I expected QuickSort to be faster. Merge sort also had the most consistent averages and standard deviations. Most standard deviations were pretty good ranging from [0-5). It seems I must not have programmed the heapsort very well as it was significantly slower than the other algorithms. It would consistently take over 100ms to sort while the other algorithms would take 10 ms at most.