Logan Jett

Rene German

CPSC 350-01

19 December 2020

Sorting Algorithms

I really enjoyed this assignment. It was extremely interesting to see the difference of these five sorting methods in real time. When initially completing this assignment, I was using a small file of simply 10 items to be sorted, which every algorithm did with ease essentially instantaneously. Once everything was working properly, I began working with larger amounts of data, and that is when I could really begin to notice the differences between each. My thoughts on each here:

1. BubbleSort: Very easy to implement and understand, but easily the slowest. In any scenario with small amounts of data, this should hold up fine and show no real performance differences.
2. InsertionSort: Similar in ease and definitely a little quicker. I found this and SelectionSort to be most similar.
3. SelectionSort: As mentioned above, but slightly more difficult to implement. I wouldn't pick this in neither small data amounts nor extremely large data amounts unless recursion was not possible.
4. QuickSort: This for me was the most difficult to implement with the partition index portion, but I can see why it might be used. This was about as quick as the previous two when using a file of about 100,000 numbers, however the data was reverse sorted so this could have had an effect on the total runtime, which was the biggest shortcoming.
5. MergeSort: This was the most fun to build, and while I felt this was slightly easier, I also understood this algorithm the most going into the assignment. At 100,000 values, the sorting time was still nearly instant. I would say that MergeSort is my favorite!