Assignment 6

Alex Rigl

2018-12-14

1 Time Differences

Overall, the time differences were not as drastic as I expected, but that might be dude to the small data set. I was suprised to see that Bubble Sort was the fastest of the sorting methods, with insertion sort and selection sort around the same. Quick sort was the slowest, possibily due to its recursive properties. However, nothing was much faster than the other despite the complexity of the algorithms and what I have learned through the book and in class.

2 Tradeoffs

The tradeoffs rely on overall complexity and speed of the data. My data set was relatively unsorted, but if it were sorted we may want to rely on something like insertion sort which does better when data sets are already sorted. Going based purely on the data, for small data sets Bubble sort would be the best but I speculate that may fall off at larger ranges and dealing with memory allocation.

3 Programming language

Working with C++ really seems like it is incredibly efficient in what it does. However, I don't see other programming languages having a large impact on speed. From what I have heard when it comes to sorting algorithms most people either use C++, Java, or Python.

4 Shortcomings

The main problem with this testing is that there was discrepancies in the data. Although the numbers stayed in the same range, and the sorting algorithms in terms of speed stayed in the same order, differences in the data show that the method of testing is not entirely accurate. Also, because I tested with only around twenty data points, the time to complete each sorting is very similar to each other.