

Eli Annoni
Prof. German
CPSC 350-01
December 19, 2020

Assignment 6: Sorting Write-Up

Testing these 5 sorting algorithms was a huge learning moment for me. I feel that we rarely get to perform empirical analysis on algorithms in class due to the amount of time it takes to implement such algorithms. So being able to finally witness just how much of a difference there is between $O(n^2)$ and $O(n \log n)$ by empirical analysis was very eye-opening to me.

With my program, I tested 3 different data set sizes: 3,000, 100,000, and 1,000,000. For 3,000, I was expecting very fast performances from all algorithms, which was true. Each algorithm performed roughly the same, and all in less than a second. For 100,000 I was surprised to see that Selection was 12 seconds, Insertion 10, Bubble 42, and Quick and Merge still < 1 . What I definitely did not expect was the result for the 1,000,000 element data set. Out of the first 3 algorithms I was only able to test Selection Sort, because it took a whole 19 *minutes* to execute. And Quick and Merge Sort were still < 1 second.

This absolutely blows my mind to see the enormous difference in performance times. It really puts into perspective the importance of selecting efficient algorithms if large amounts of data are being handled, as runtime difference could be as drastic as minutes, even hours.

However, the downsides to empirical analysis would be the implementation time, as even simple sorting algorithms can be tedious to implement. Also, even the testing time could take a while, since I wasn't even able to get to Insertion or Bubble Sort, which would have taken way longer.

In conclusion, empirical analysis can be extremely helpful for programmers to understand the sheer importance of what their decisions mean, yet it is not a reliable and sustainable method of algorithm selection.