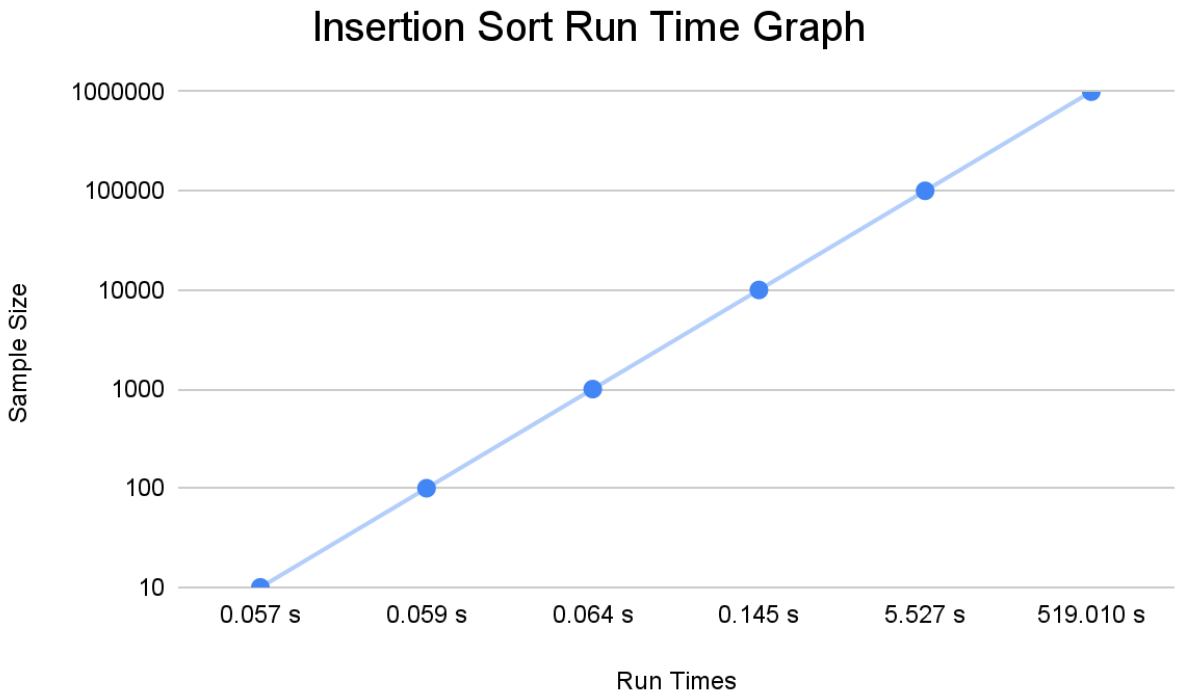
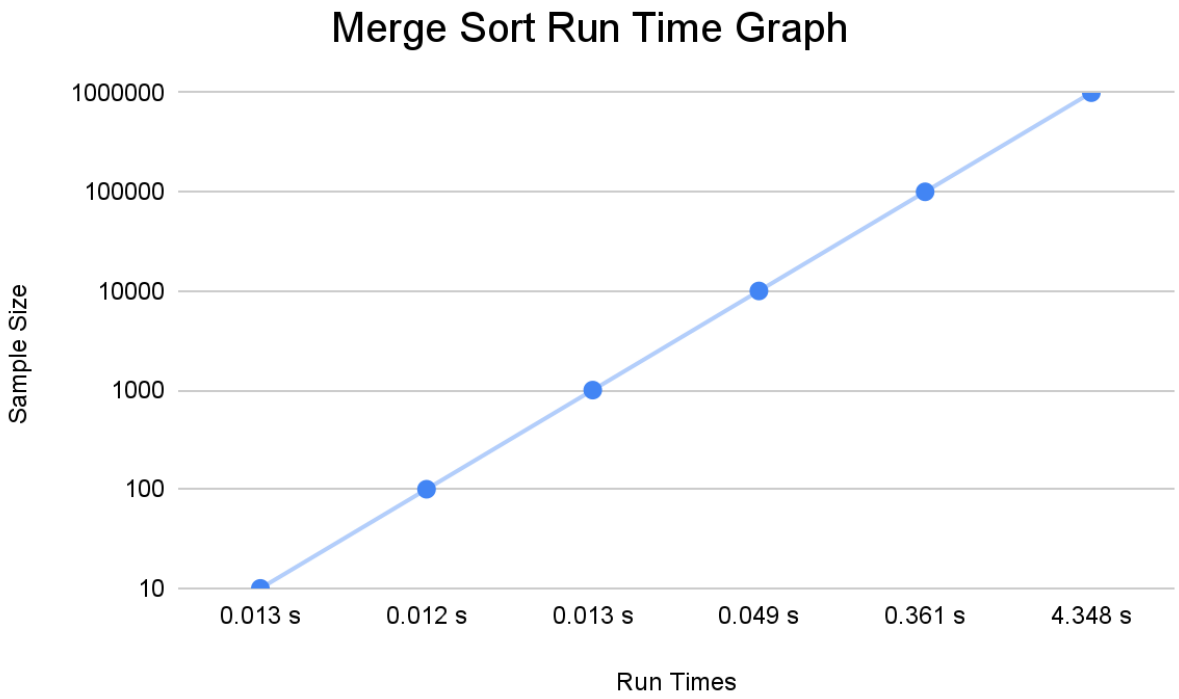
Tro Hovasapian
Coen 12L Fall 2022
Thursday Lab 2:15
Lab Project 5 - Report
12/02/2022

	Θ(n Log n) algorithm Merge Sort	Θ(n^2) algorithm Insertion Sort
10	0.013 s	0.057 s
100	0.012 s	0.059 s
1000	0.013 s	0.064 s
10000	0.049 s	0.145 s
100000	0.361 s	5.527 s
1000000	4.348 s	519.010 s

When looking at time complexity, in theory, merge sort would be faster. Since we know the order of time complexity is O(1), O(log n), O(n), (n log n), O(n^2), and so on, we know which one would be faster. As merge sort is O(n log n), and insertion sort is O(n^2), we know which one is faster, but also that the difference in time increases as the sample size increases. Through multiple trials, we can see that the run time of insertion sort grows exponentially with bigger samples. While mergesort's runtime does grow as well, the increases in time are not comparable to insertion sort. With a sample size of 1 million, the run time of merge sort was 4.4 seconds, roughly, and the run time for insertion sort was over 8 minutes. We can conclude that the difference between runtimes increases with larger gaps as n increases.


Insertion Sort Run Time Graph
1000000
100000
10000
1000
100
10
Sample Size
0.057 s
0.059 s
0.064 s
0.145 s
5.527 s
519.010 s
Run Times


Merge Sort Run Time Graph
1000000
100000
10000
1000
100
10
Sample Size
0.013 s
0.012 s
0.013 s
0.049 s
0.361 s
4.348 s
Run Times