Mahsha Tizhoush
mtizhoush@csu.fullerton.edu
CPSC 335

Project 2 Report

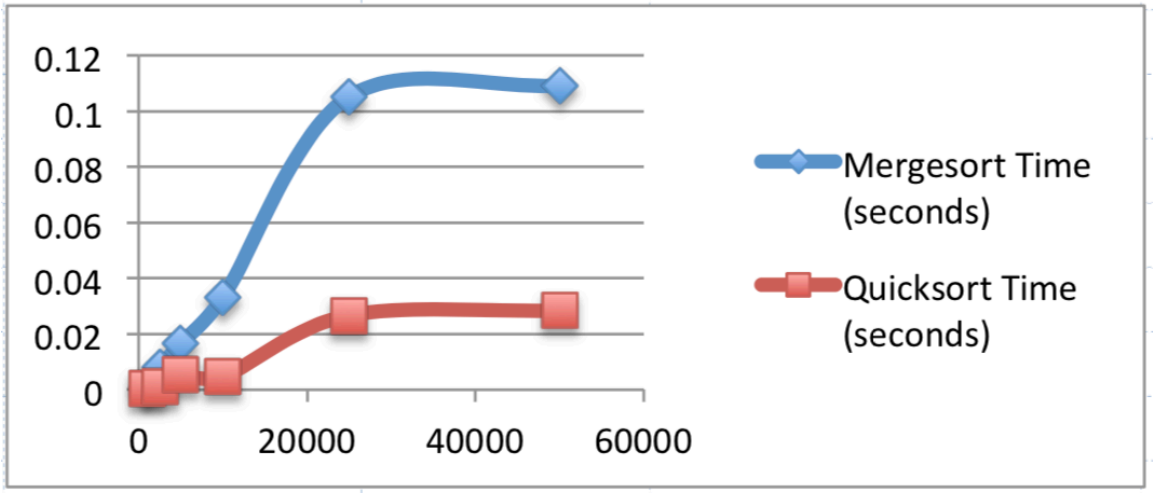
Hypothesis:

- This experiment will test the following:
  1. Randomization can be used to generate data for testing an algorithm and determining performance
  2. Two algorithms of the same efficient class can have different average running times and different ranges of performance

Analysis of Data:

Both algorithms tested with different instances of n:

Size (n)	Mergesort Time	Quicksort Time (seconds)
1000	0.00111842	0.000362563
2500	0.00810762	0.00102816
5000	0.0165582	0.00526829
10000	0.0331502	0.00477769
25000	0.105219	0.0267083
50000	0.109198	0.0284237



Both algorithms tested with the same size of n, ran multiple times:

Quicksort			Mergesort	
Size (n)	Time (seconds)		Size (n)	Time (seconds)
50000	0.0549262		50000	0.215421
50000	0.0797435		50000	0.246335
50000	0.0708299		50000	0.214219
50000	0.0682709		50000	0.203474
50000	0.0742501		50000	0.236928
50000	0.0787088		50000	0.220979
50000	0.0737112		50000	0.226667
50000	0.0856983		50000	0.212246
50000	0.0748971		50000	0.238118
50000	0.0709645		50000	0.224962
Average:	0.07320005		Average:	0.2239349
Standard Deviation:	0.008181677		Standard Devia	0.013379778

Conclusion:

From observing the data I acquired, I can conclude that the hypothesis is true. The average time for mergesort with a fixed size of n was about 0.22 seconds, while for quicksort it was about 0.07 seconds. Even with various sizes of n, quicksort still outperformed over mergesort. This concludes that even though both these algorithms have an efficiency of O(nlogn), quicksort is slightly more efficient and quicker than mergesort.