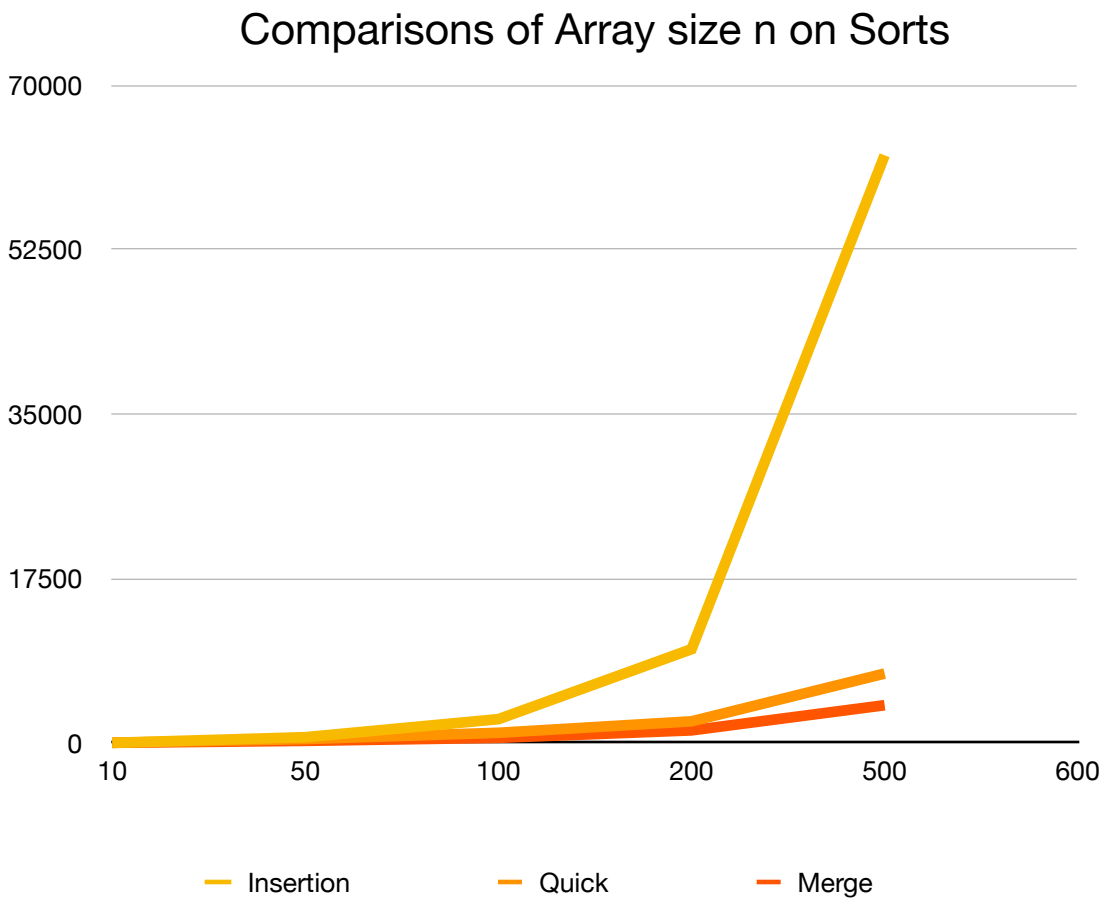
Comparisons

n	Insertion	Quick	Merge
10	26	31	28
50	626	427	237
100	2546	1094	557
200	9988	2312	1351
500	62507	7394	4026

We are not comparing values in the array for radix sort! This is why we don't have a radix column.



Discussion

Overall, this was a very difficult first assignment for me. After many hours, and 3 slip days used - I feel as if my algorithms and SortsDriver are completely correct. Radix sort definitely took the most time implementing, as I used bytes to compare. This is a counting sort algorithm and I referred to Sedgewick's work. I really enjoyed the pressure that this project gave me, and I seemed to work through it after a lot of research.

This was the first time I have ever used gradle to compile and run my programs, I think this adds immense ease-of-use. If I had more time, I would implement another enhancement- but I feel as if I put sufficient work to understand the 'big idea' of these sorting algorithms. *My enhancement* was the *countSort* method which utilizes max/mins and is proportionate to the range of the array. It is not comparison based, so technically beats out insertion & quickSort! Press "c" when prompted to access the secret sort.

This assignment took me over thirty hours in total.

Acknowledgements

Much research was done for this assignment. Whew! Stack Overflow had many questions and answers that assisted me with developing this program.

Professor Java - http://professorjava.weebly.com/index.html

Sedgewick's Algorithms - https://algs4.cs.princeton.edu/51radix/

Joseph Tully - helper methods for use as-is, I manipulated and changed a couple of the methods.

Geeks for Geeks - https://www.geeksforgeeks.org/counting-sort/