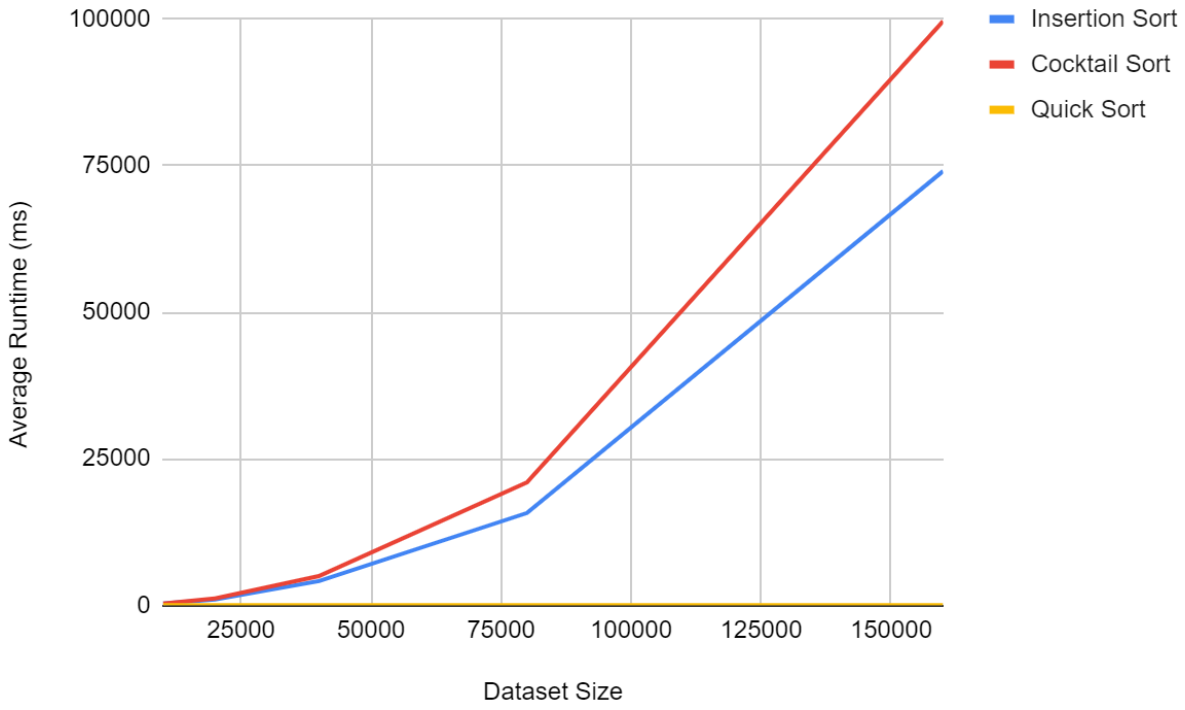
(1) Testing Insertion Sort vs. Cocktail Sort vs. QuickSort

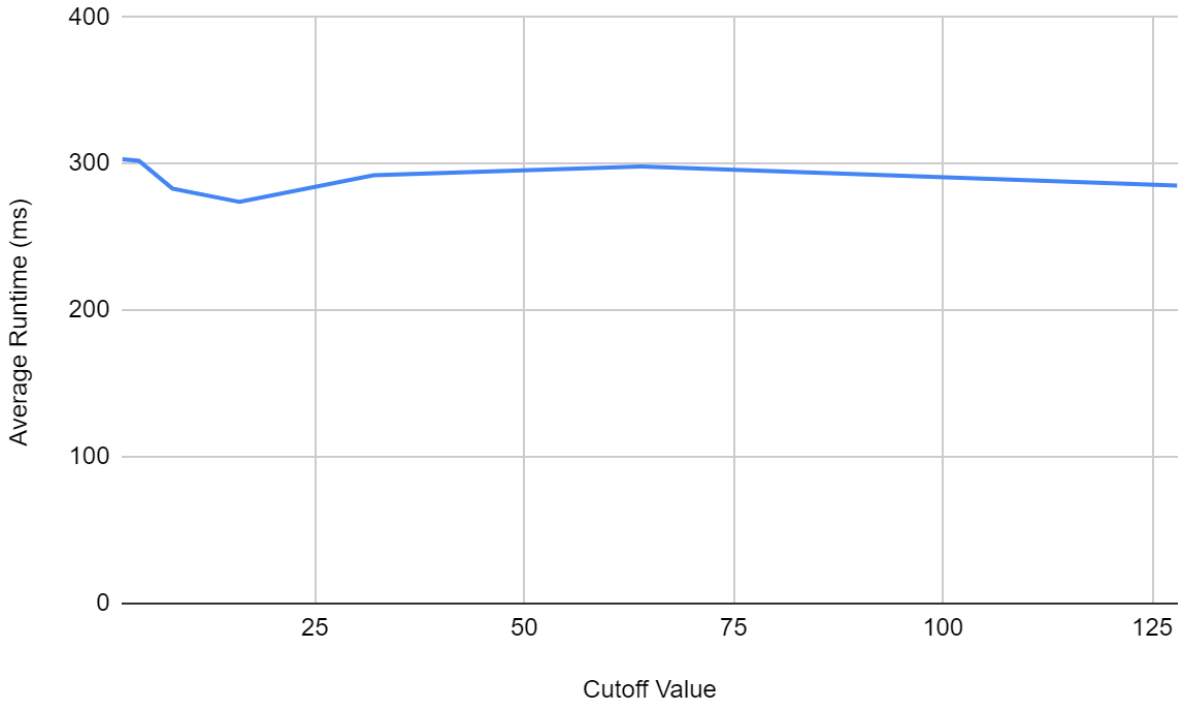
Dataset Size	Insertion Sort Average Time (ms)	Cocktail Sort Average Time (ms)	QuickSort Average Time (ms)
10000	295	312	5
20000	1006	1151	5
40000	4136	4988	8
80000	15751	20963	25
160000	74063	99648	48



Based on the plot, in terms of time complexity, it appears that Quicksort is the most efficient out of the algorithms compared. After Quick Sort, Insertion Sort is the most efficient and finally, Cocktail Sort is the least efficient. If we break down the logic behind the sort algorithms, it makes sense why they are the most and least efficient algorithms. Quick Sort uses partitioning and recursive calls to compare the elements to each other and sort them at an individual level, then combines them back together to form a sorted list. On the other hand, Cocktail Sort goes forward and backward comparing each pair and swapping if necessary until no swaps need to be made, which requires more operations compared to breaking down the list into simpler comparisons.

(2) Testing Modified QuickSort Cutoff Values

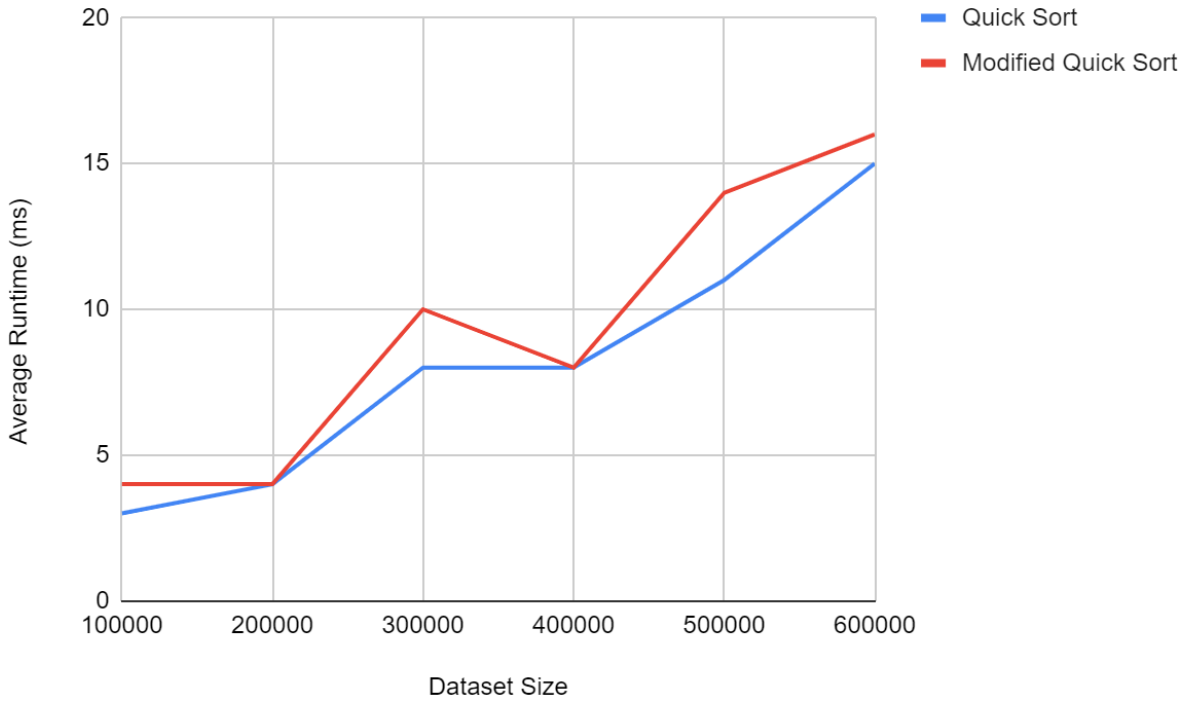
Dataset Size	Cutoff	Average Time (ms)
1,000,000	2	303
1,000,000	4	302
1,000,000	8	283
1,000,000	16	274
1,000,000	32	292
1,000,000	64	298
1,000,000	128	285



The cutoff at 16 gave me the fastest performance at 274 ms in comparison to the other cutoffs.

(3) Testing Traditional QuickSort vs. Modified QuickSort

Dataset Size	Cutoff Value	Quicksort Average Time (ms)	Modified QuicksortAverage Time (ms)
100000	16	3	4
200000	16	4	4
300000	16	8	10
400000	16	8	8
500000	16	11	14
600000	16	15	16



Based on the plot above, the Quick Sort is more efficient in terms of time complexity in comparison to the Modified Quick Sort. While their algorithm is nearly identical. Depending on the cutoff value, modified Quicksort utilizes Insertion Sort, which is neither the most or least efficient. Due to that modification, it could be one of the reasons why the Quick Sort algorithm is still more efficient than the Modified Quick Sort. However, this data was dependent on the cutoff value and perhaps that contributed to the above conclusion.