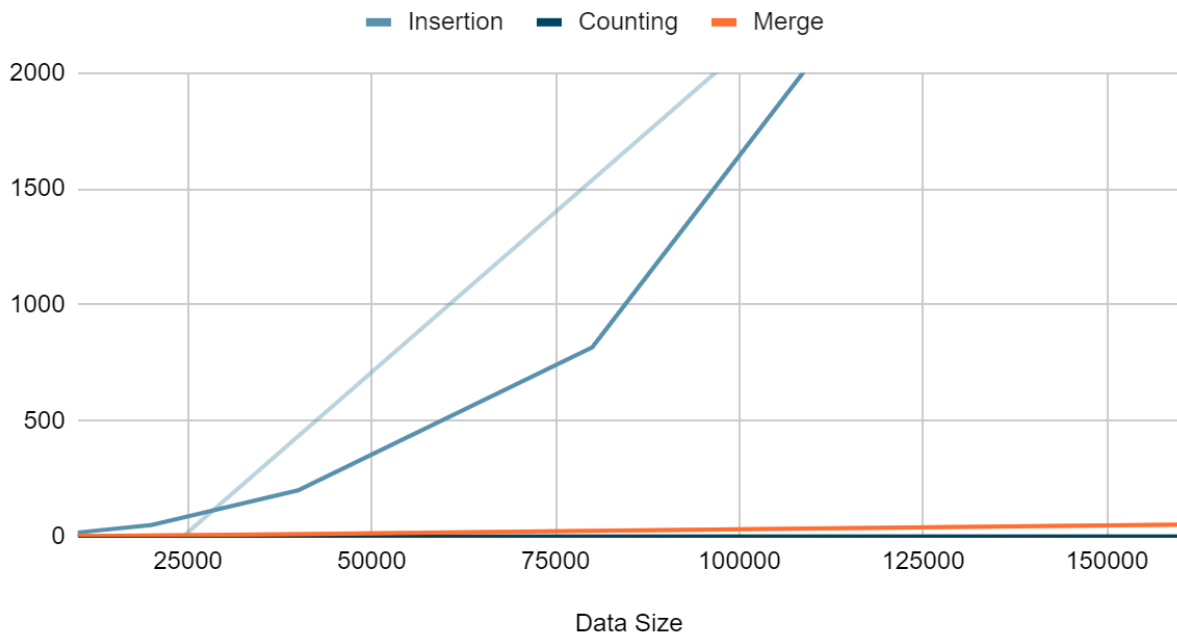Task 1

Insertion, Counting and Merge
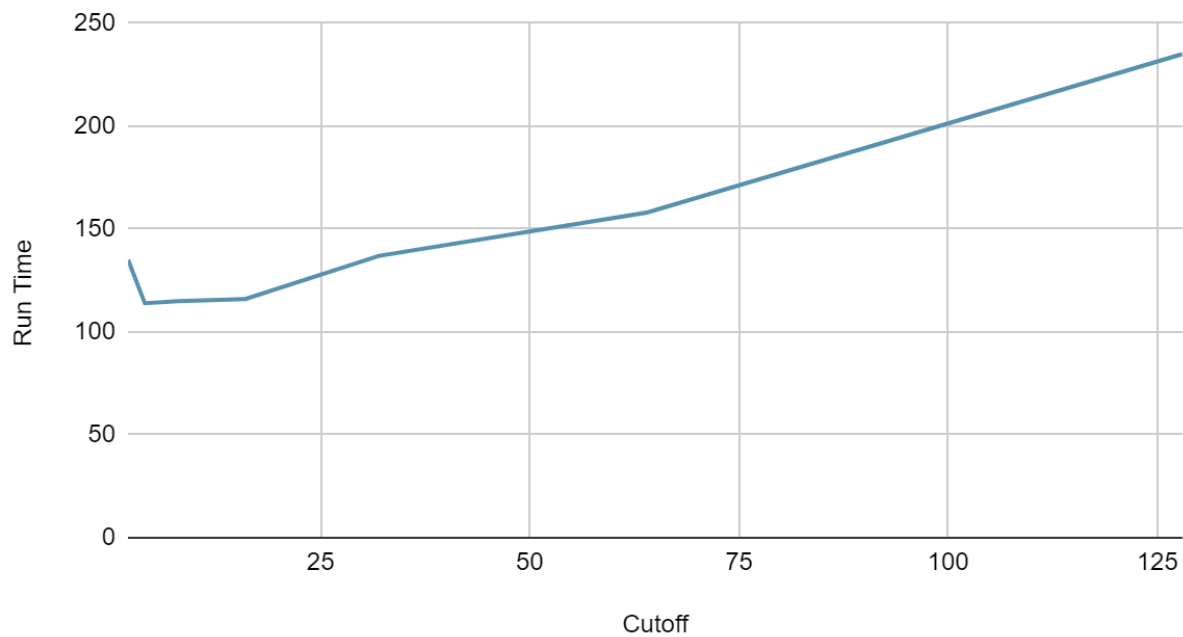
— Insertion   — Counting   — Merge

Data Size

Conclusion:

The Insersion sort grow so much faster than the other two sorting method, since the complexity of Insersion sort is $O(n^2)$ and the complexity of counting $O(n)$ and merge sort is $O(nlog(n))$.
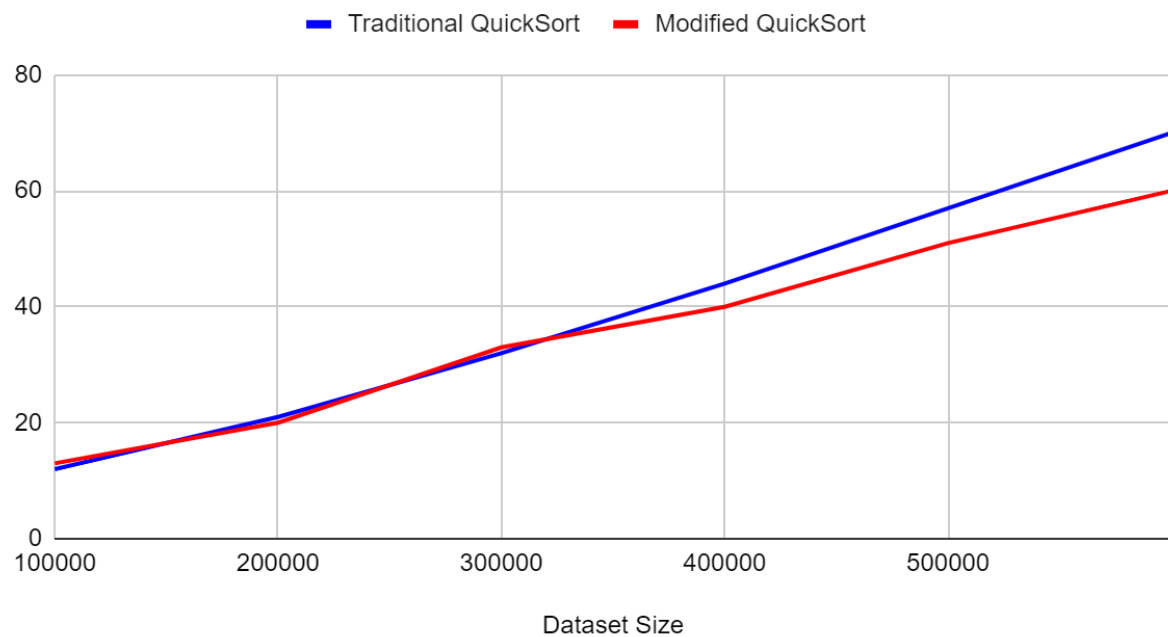
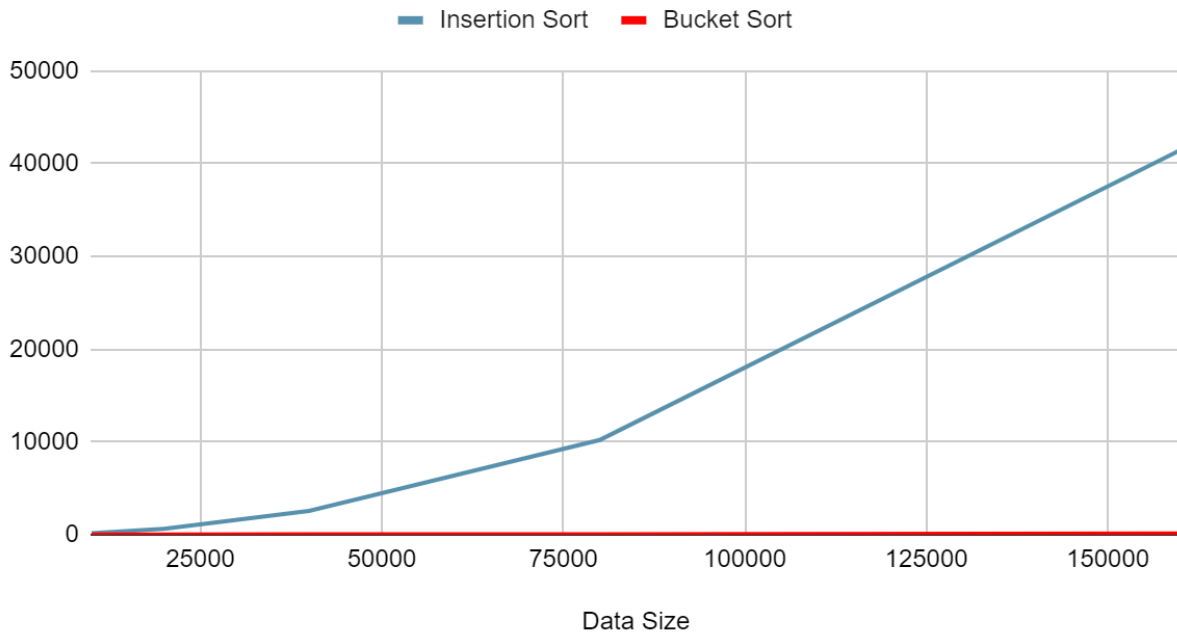Task 2:

Run Time vs. Cutoff

The best cutoff is at 4.
Task 3:

## Traditional QuickSort and Modified QuickSort

Task 4:

Insertion Sort and Bucket Sort
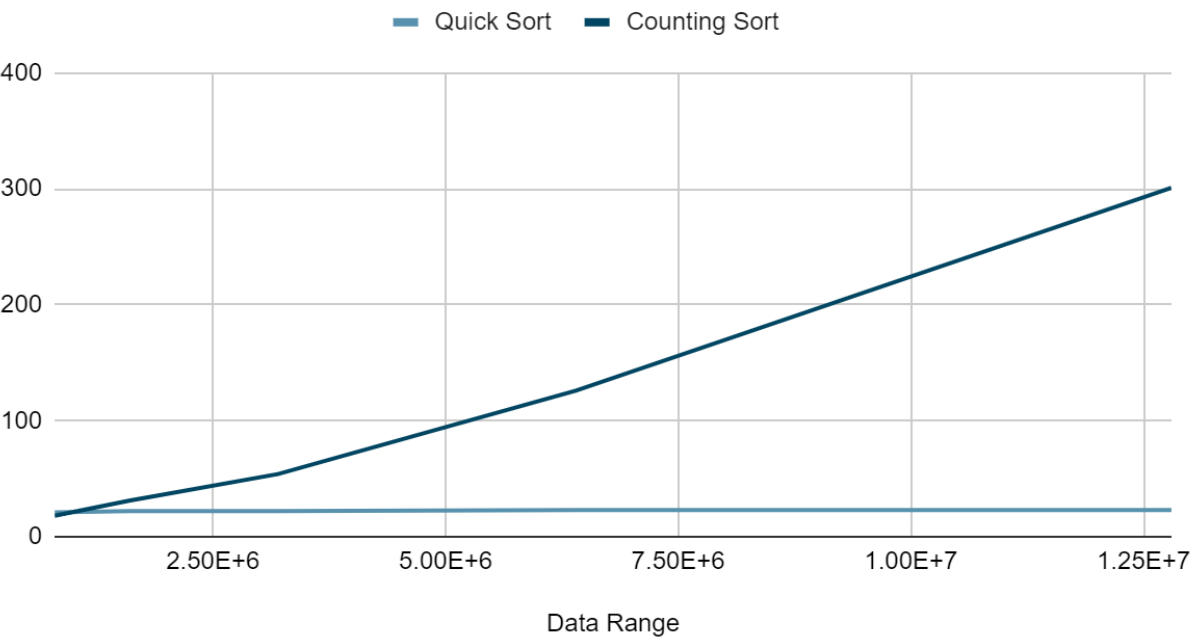
— Insertion Sort   — Bucket Sort

Data Size

Since the complexity of insertion sort is O(n^2) which is so much greater than bucket sort. When n increases, the blue line increments so much more than the red line.

Task 5:

Size 50000, Quick Sort and Counting Sort

— Quick Sort — Counting Sort

Data Range
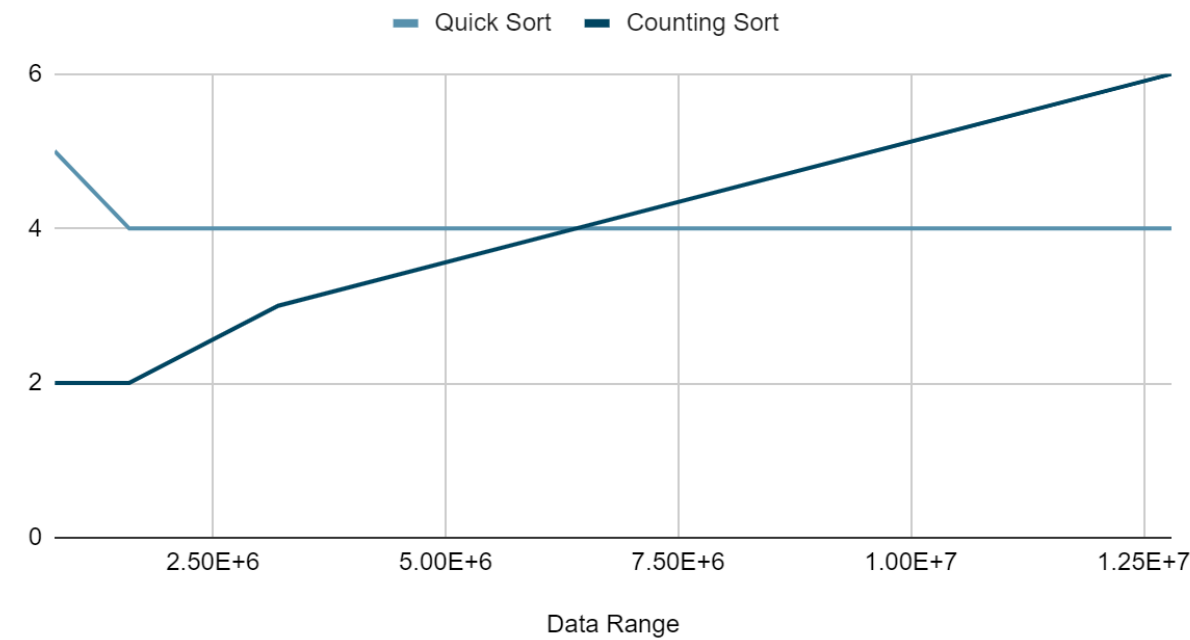
Size 200000, Quick Sort and Counting Sort

— Quick Sort — Counting Sort

Data Range

The range of data is completely correlated to the running time of quick sort, since it treats every element as a byte. However, the running time of counting sort increases dramatically as the range of data increases. It is because the counting sort needs to create a new array with length of the maximum value of array.