1)

Insertion Sort

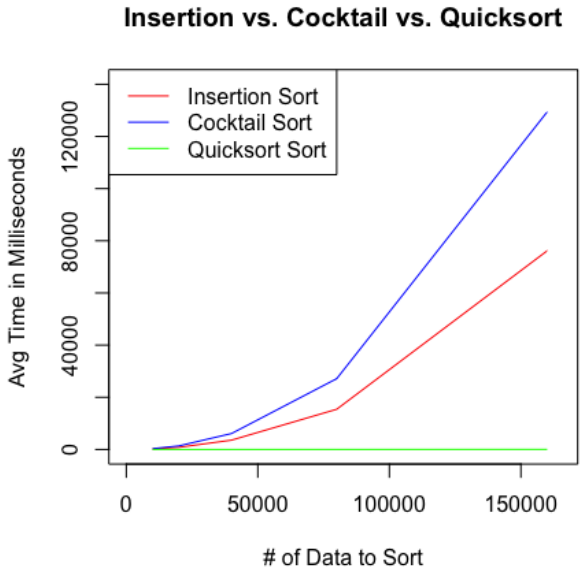
# of Data to Sort	Average Time Taken to Sort (Milliseconds)
10000	224
20000	820
40000	3567
80000	15419
160000	76174

Cocktail Sort

# of Data to Sort	Average Time Taken to Sort (Milliseconds)
10000	304
20000	1368
40000	6104
80000	27153
160000	129406

Quick Sort

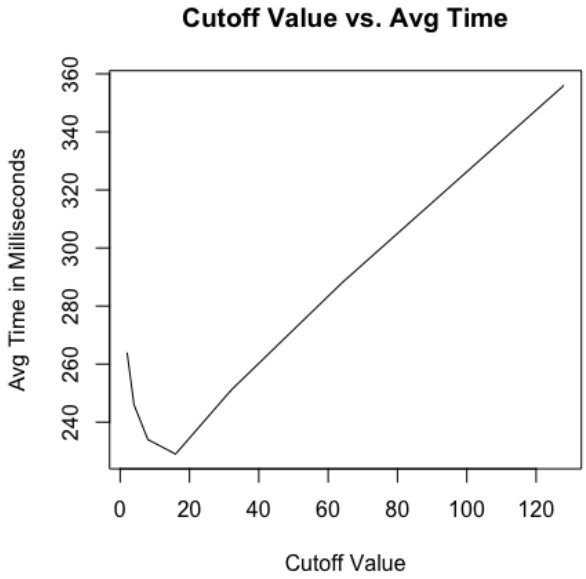
# of Data to Sort	Average Time Taken to Sort (Milliseconds)
10000	3
20000	3
40000	5
80000	14
160000	30



It appears that cocktail sort had the worst average time complexity increasing in an exponential-like fashion. Insertion sort appears to have the second worst time complexity with shallower exponential growth. Quicksort had by far the best time complexity as the number of data grew, the average time remained stable and even appears to decrease slightly.

2)

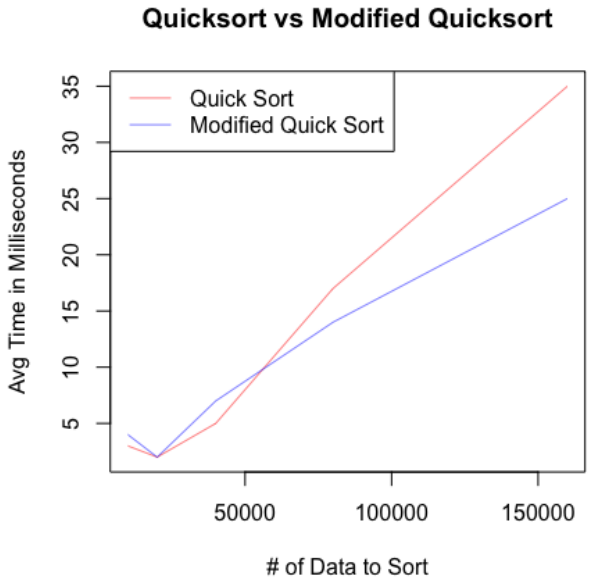
Cutoff Value	Avg Time in Milliseconds
2	264
4	246
8	234
16	229
32	251
64	288
128	356



The cutoff value of 16 appears to give me the best average run time as it is the global minimum of this curve. The runtime seems to decrease exponentially until cutoff 16 and then increases at a seemingly linear rate after that.

3)

# of Data to Sort	Quick Sort Avg Time in Milliseconds	Modified Quick Sort Avg Time in Milliseconds
10000	3	4
20000	2	2
40000	5	7
80000	17	14
160000	35	25



It appears that modified quicksort performs slightly slower in comparison to quick sort for values under ~50000, however after that point modified quicksort scales much more favorably in comparison to normal quicksort. Modified quicksort seems to be faster with greater amount of data in comparison to normal quicksort.