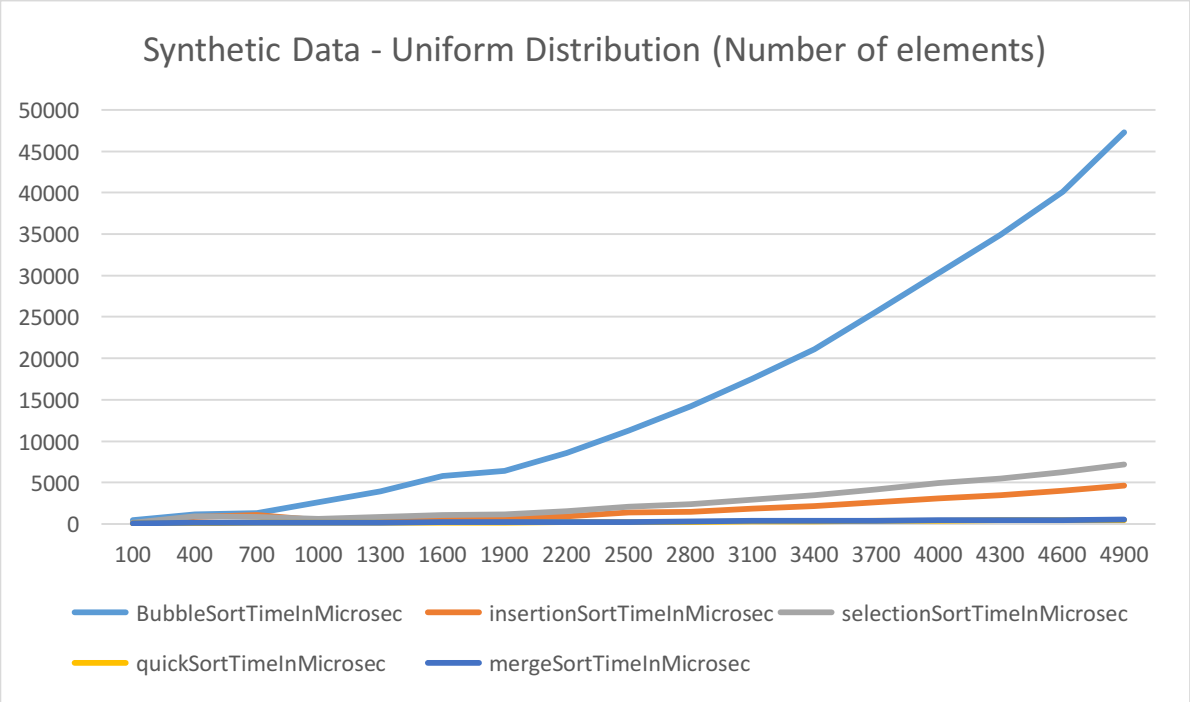
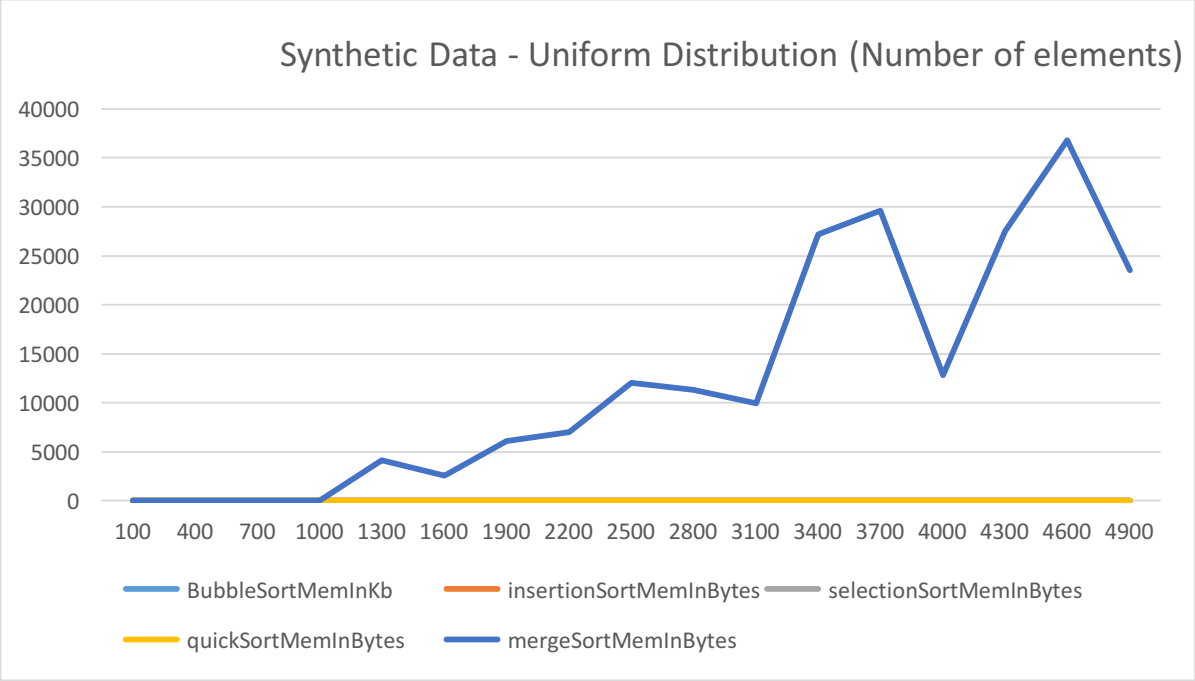
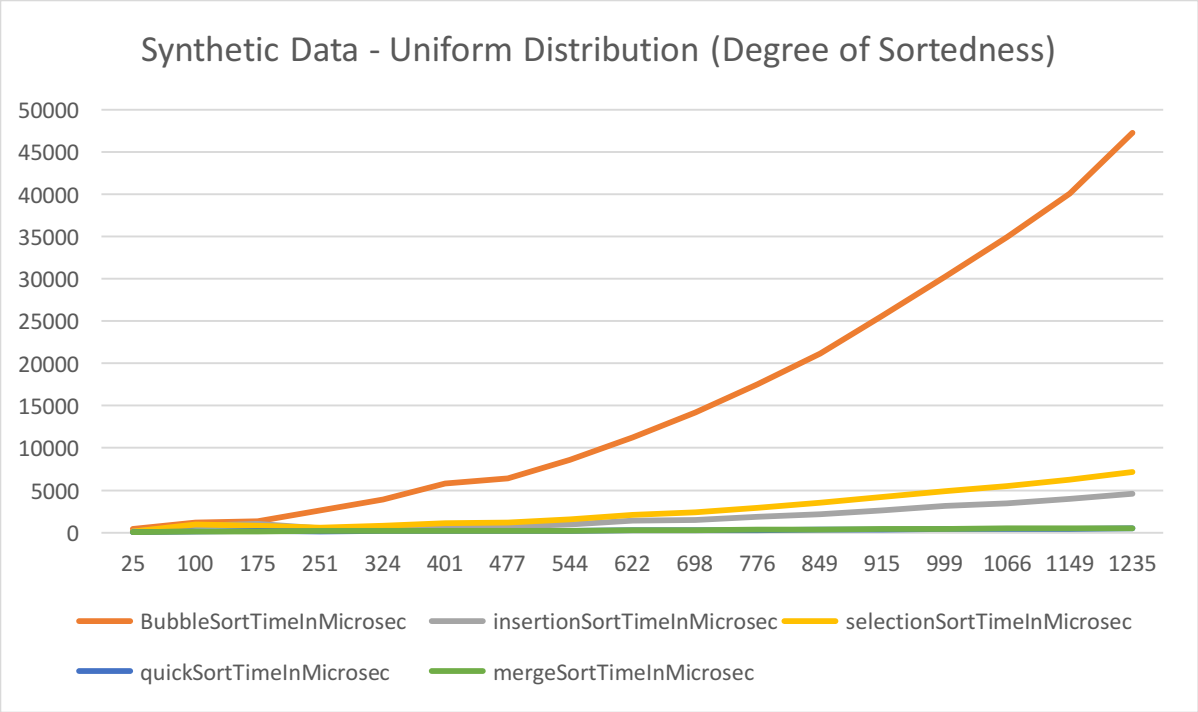
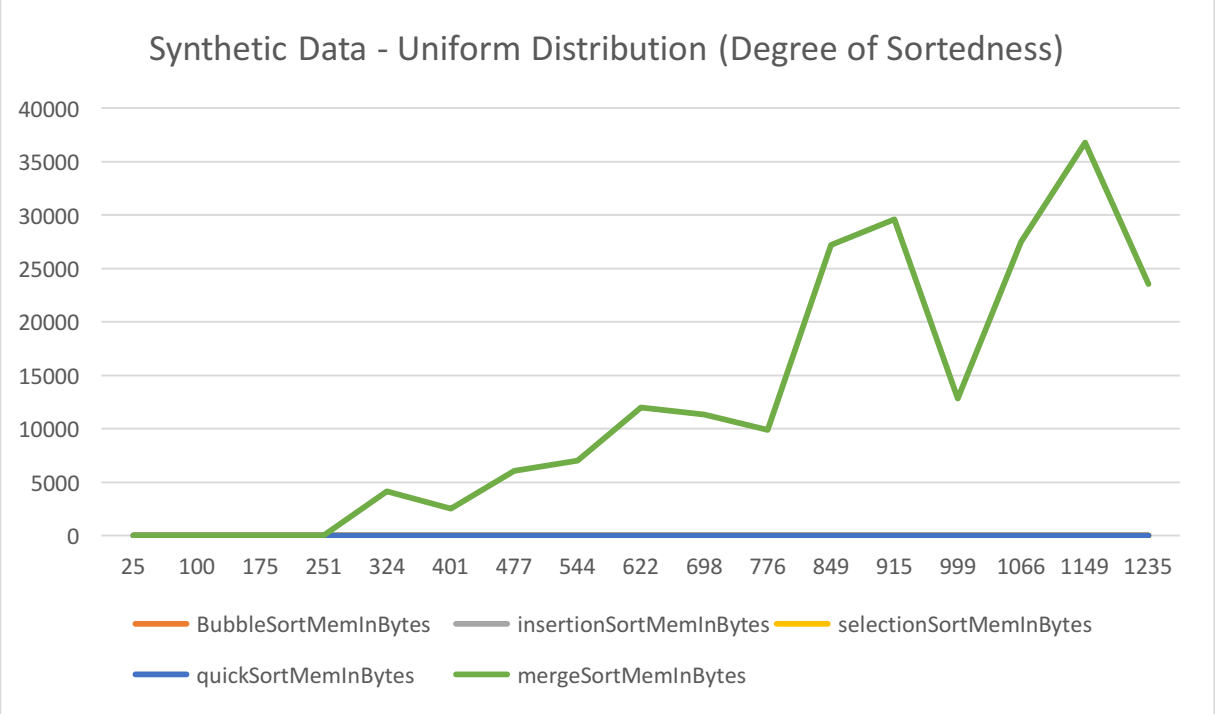
SORTING – SIMULATION

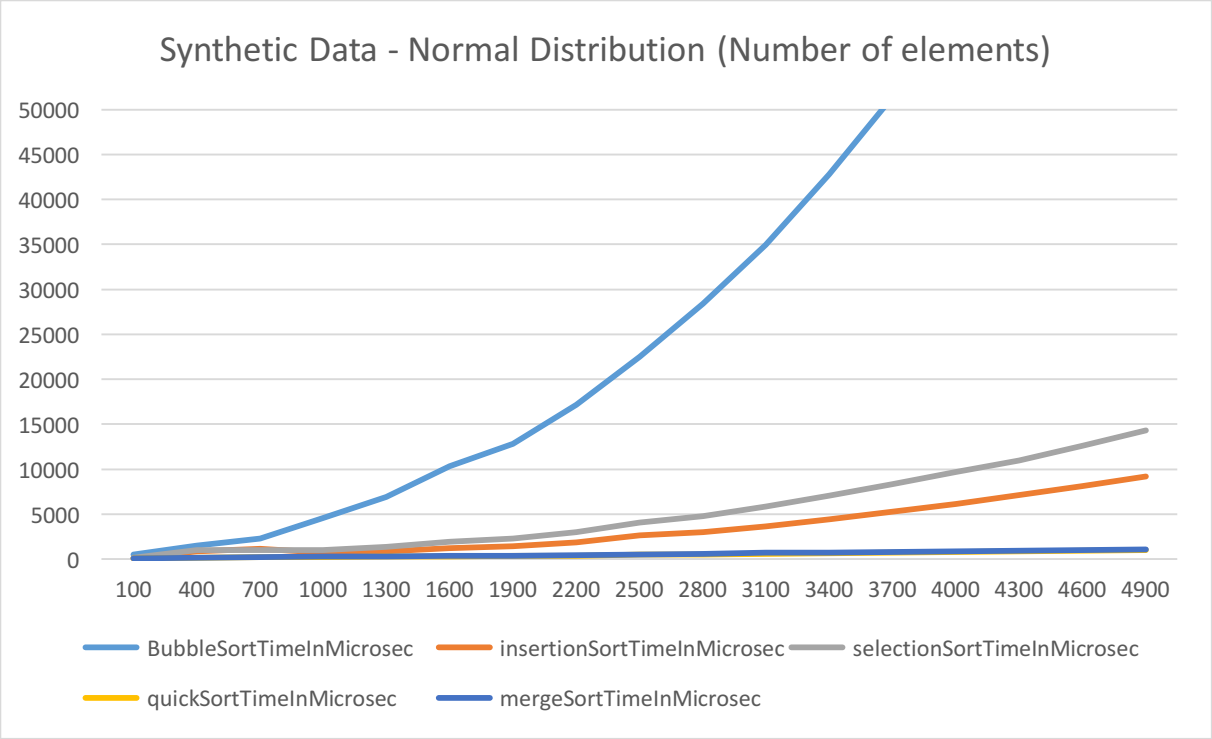
Iswarya Sridharan

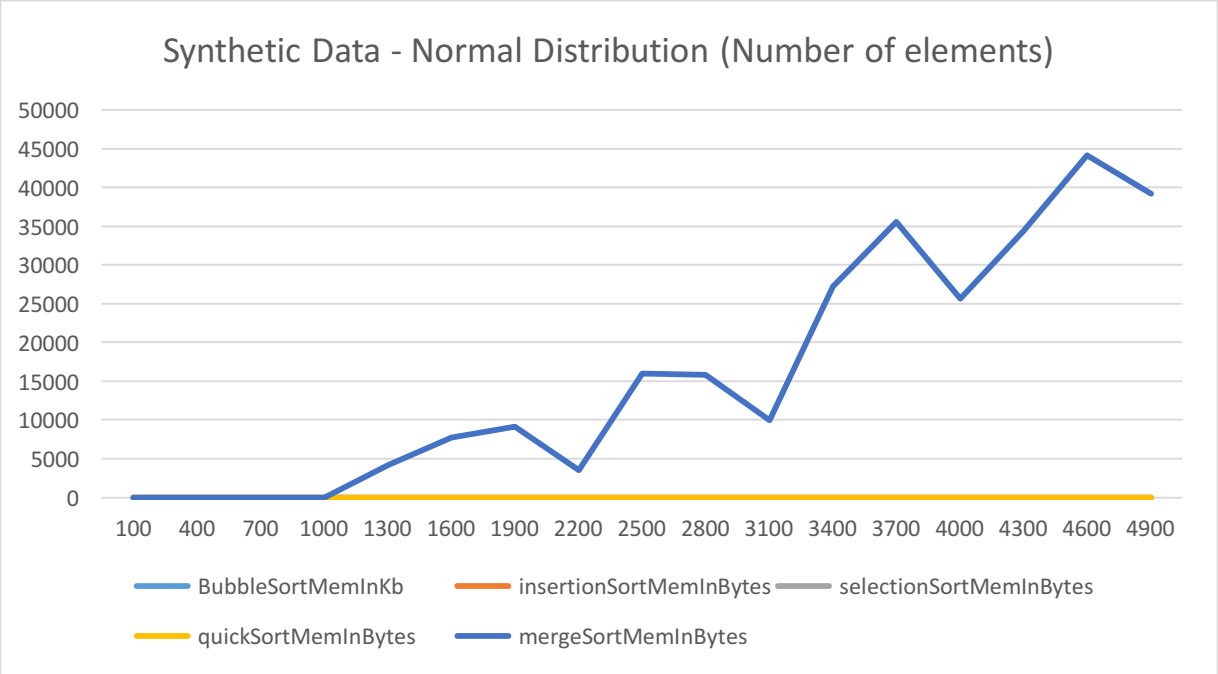


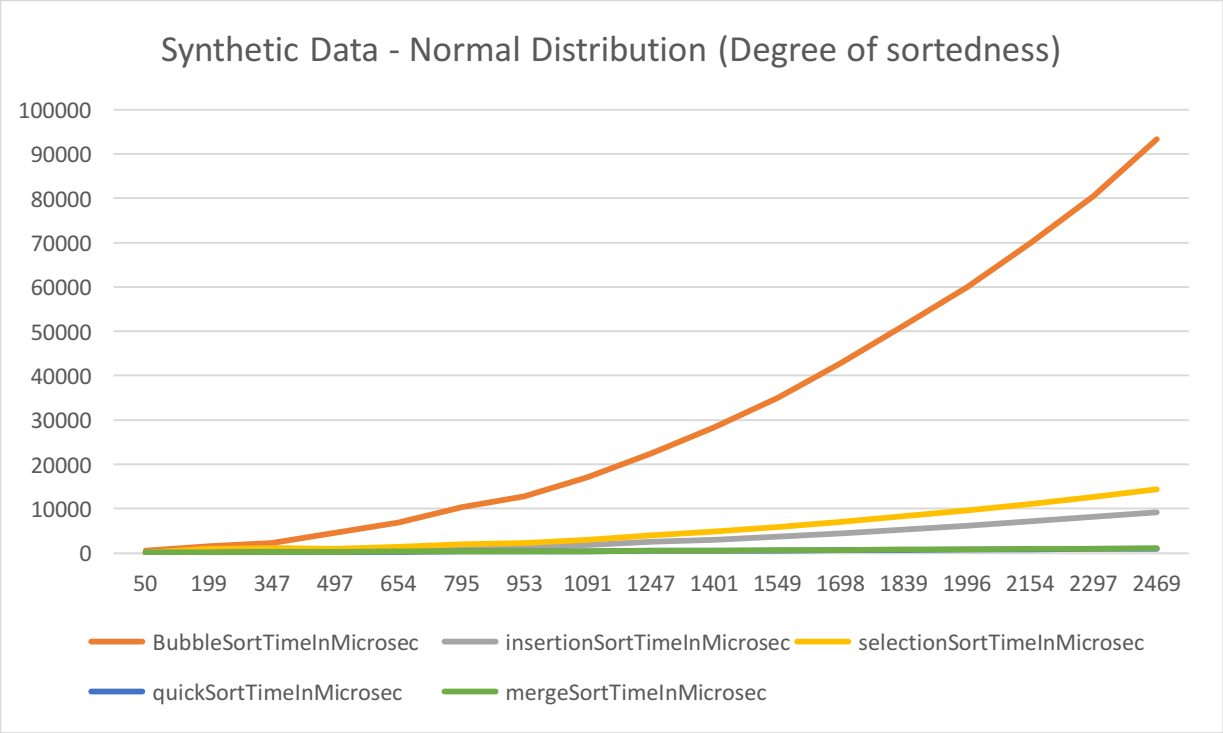


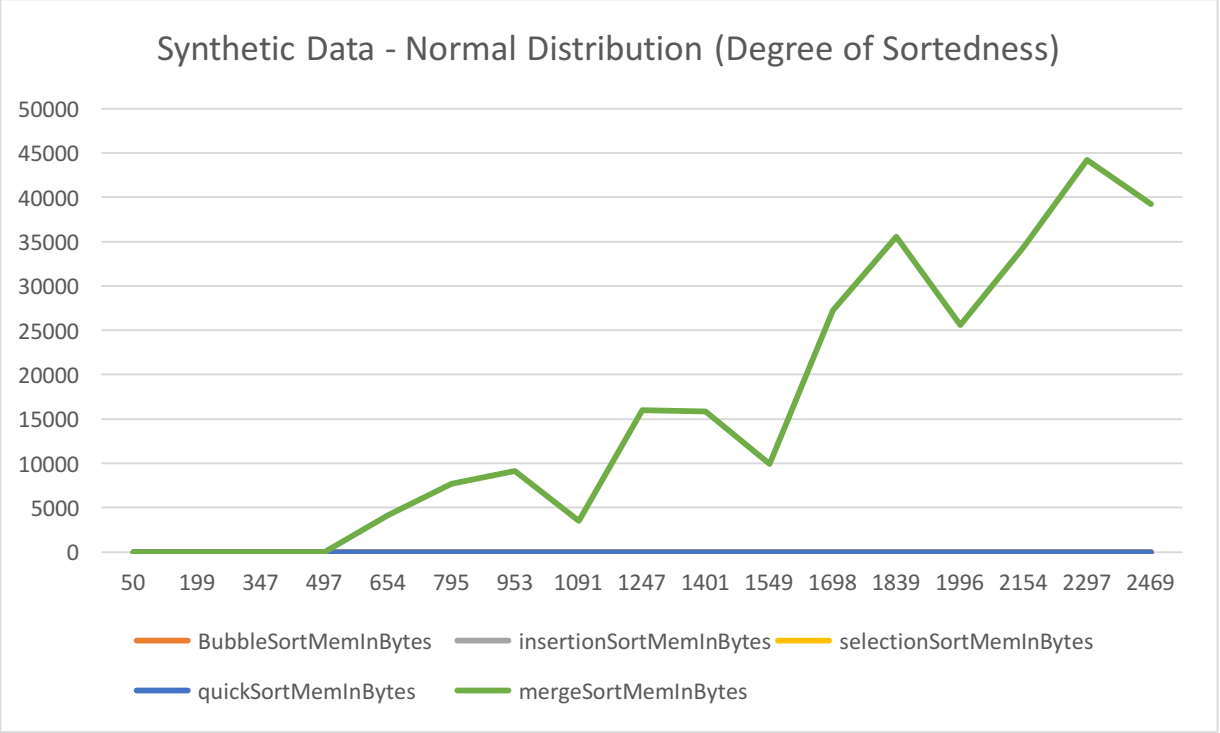

Synthetic Data - Uniform Distribution (Degree of Sortedness)
50000
45000
40000
35000
30000
25000
20000
15000
10000
5000
0
25 100 175 251 324 401 477 544 622 698 776 849 915 999 1066 1149 1235
BubbleSortTimeInMicrosec
insertionSortTimeInMicrosec
selectionSortTimeInMicrosec
quickSortTimeInMicrosec
mergeSortTimeInMicrosec

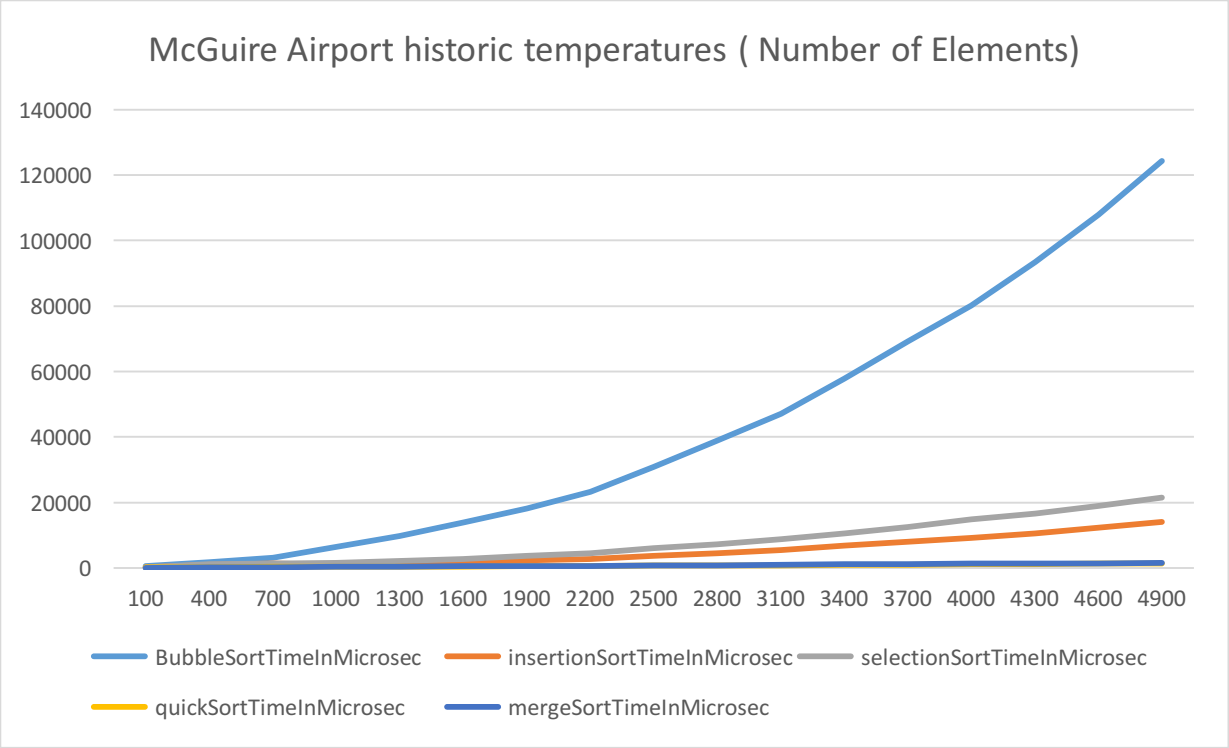

Synthetic Data - Uniform Distribution (Degree of Sortedness)
40000
35000
30000
25000
20000
15000
10000
5000
0
25 100 175 251 324 401 477 544 622 698 776 849 915 999 1066 1149 1235
BubbleSortMemInBytes
insertionSortMemInBytes
selectionSortMemInBytes
quickSortMemInBytes
mergeSortMemInBytes

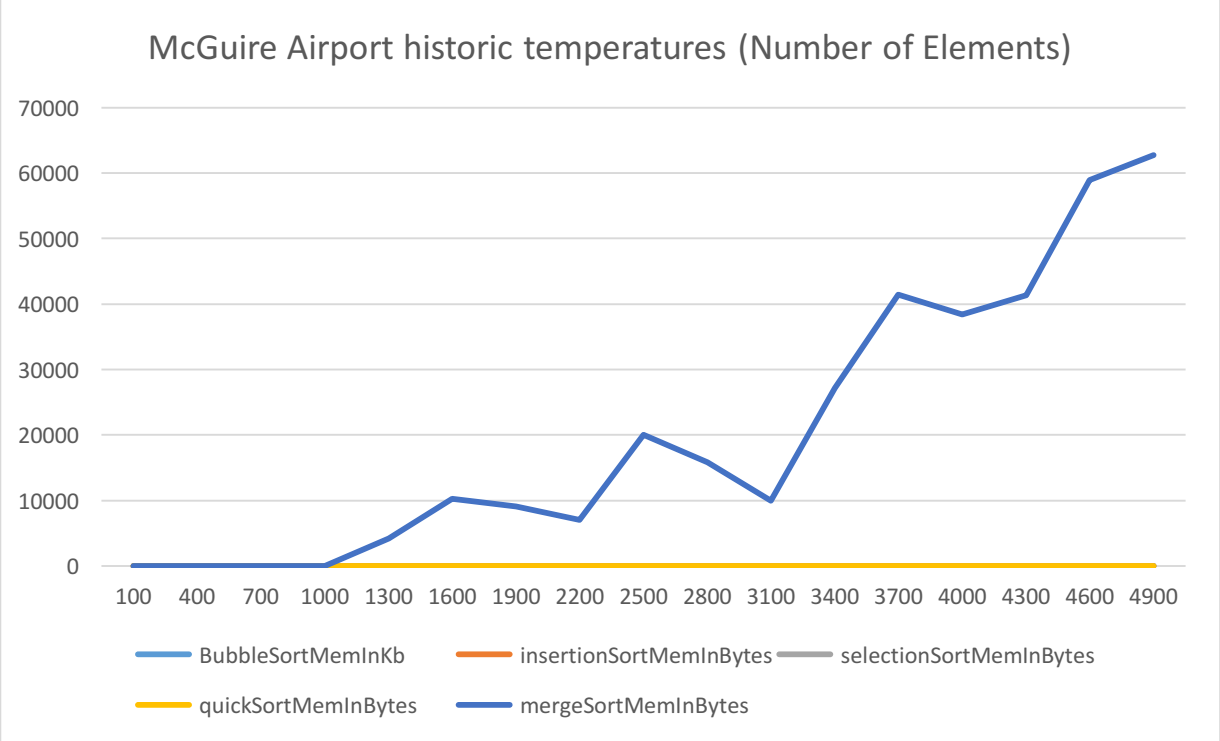

Synthetic Data - Normal Distribution (Number of elements)
50000
45000
40000
35000
30000
25000
20000
15000
10000
5000
0
100 400 700 1000 1300 1600 1900 2200 2500 2800 3100 3400 3700 4000 4300 4600 4900
BubbleSortTimeInMicrosec
insertionSortTimeInMicrosec
selectionSortTimeInMicrosec
quickSortTimeInMicrosec
mergeSortTimeInMicrosec

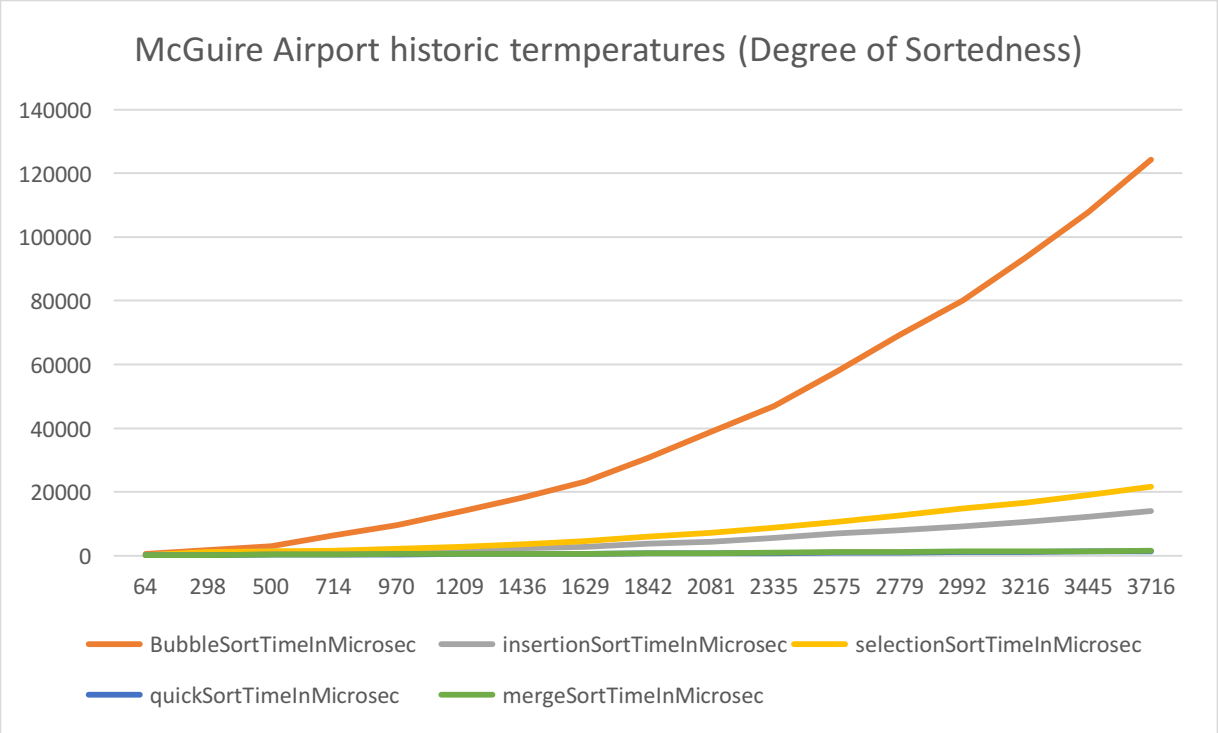

Synthetic Data - Normal Distribution (Number of elements)
50000
45000
40000
35000
30000
25000
20000
15000
10000
5000
0
100 400 700 1000 1300 1600 1900 2200 2500 2800 3100 3400 3700 4000 4300 4600 4900
BubbleSortMemInKb
insertionSortMemInBytes
selectionSortMemInBytes
quickSortMemInBytes
mergeSortMemInBytes

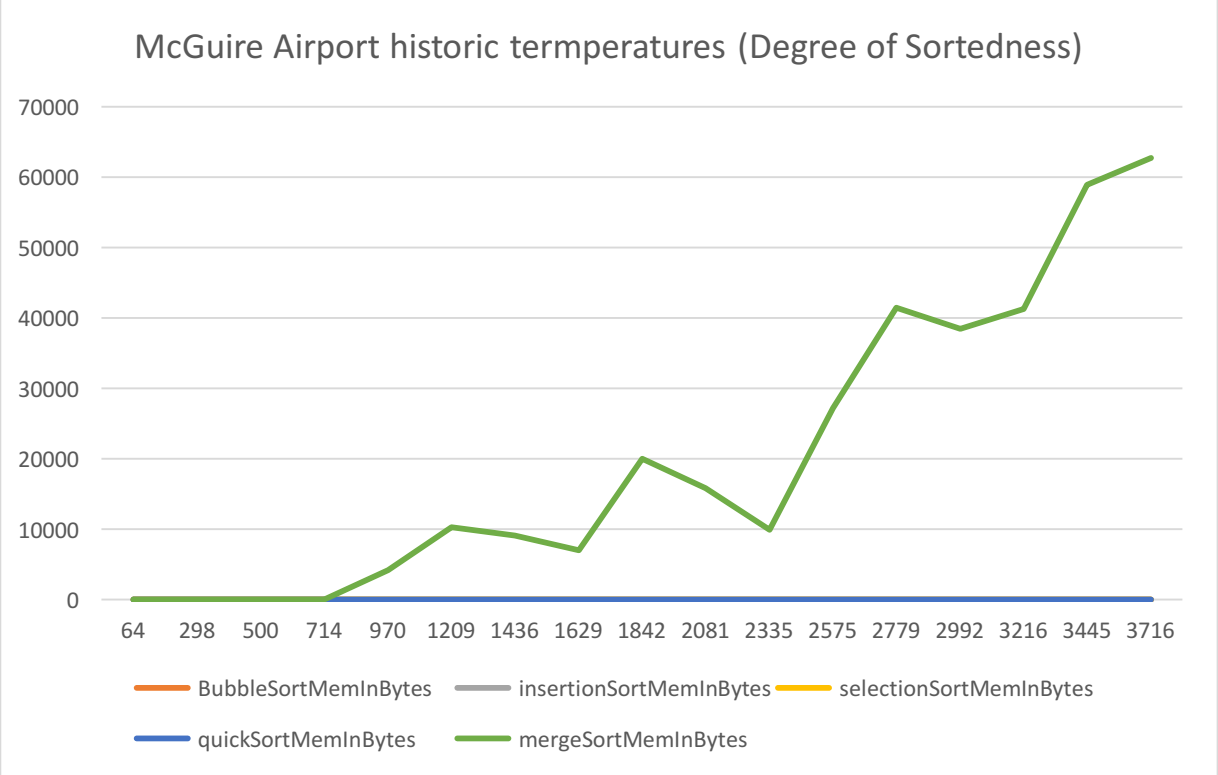

Synthetic Data - Normal Distribution (Degree of sortedness)
100000
90000
80000
70000
60000
50000
40000
30000
20000
10000
0
50 199 347 497 654 795 953 1091 1247 1401 1549 1698 1839 1996 2154 2297 2469
BubbleSortTimeInMicrosec
insertionSortTimeInMicrosec
selectionSortTimeInMicrosec
quickSortTimeInMicrosec
mergeSortTimeInMicrosec

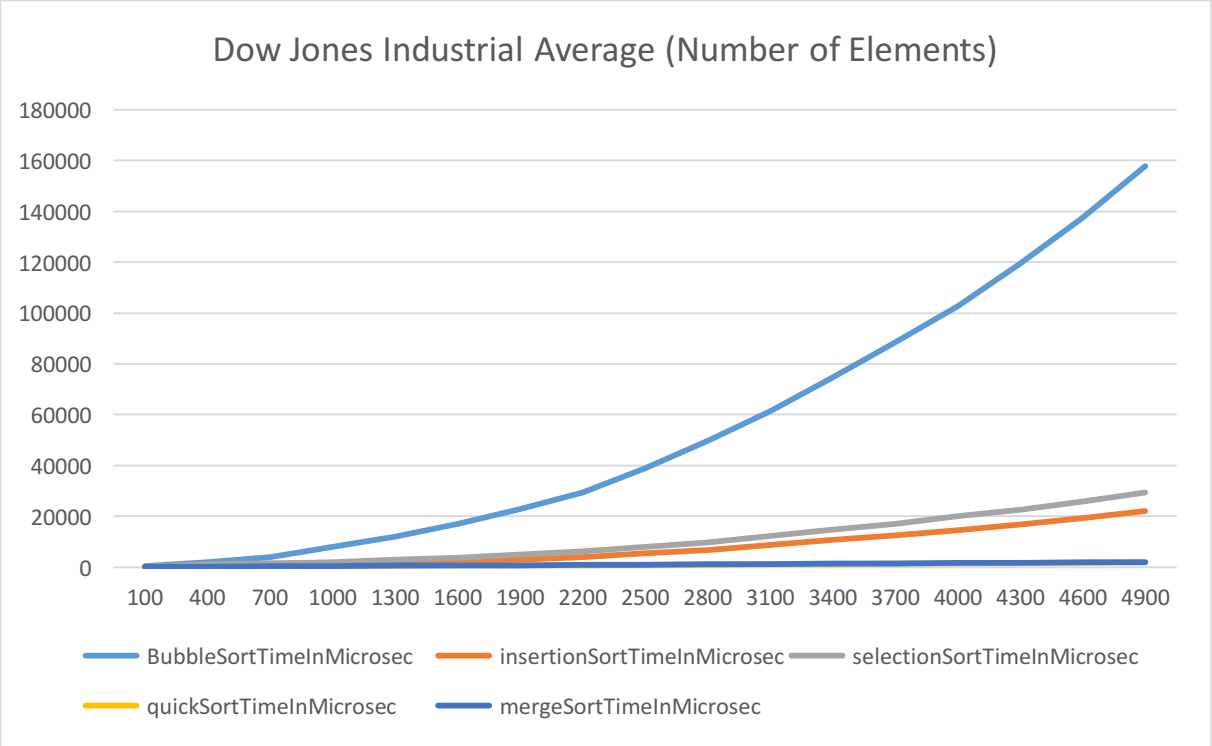

Synthetic Data - Normal Distribution (Degree of Sortedness)
50000
45000
40000
35000
30000
25000
20000
15000
10000
5000
0
50 199 347 497 654 795 953 1091 1247 1401 1549 1698 1839 1996 2154 2297 2469
BubbleSortMemInBytes
insertionSortMemInBytes
selectionSortMemInBytes
quickSortMemInBytes
mergeSortMemInBytes

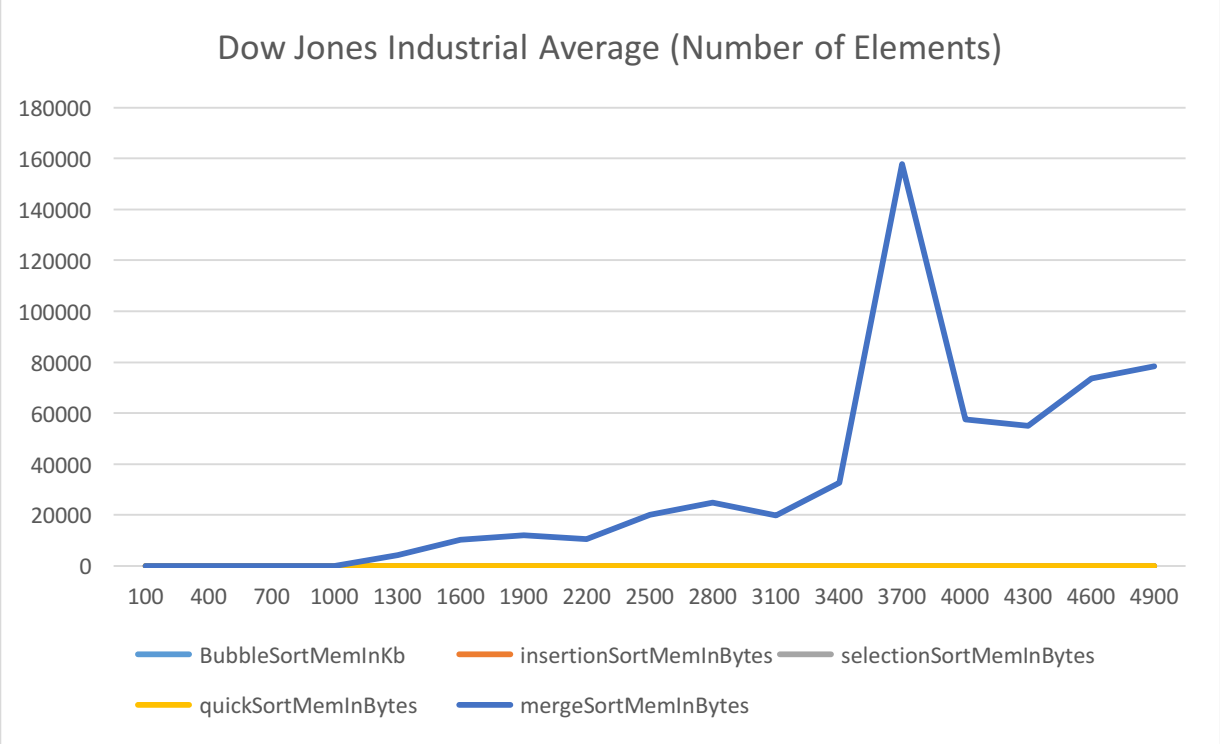

McGuire Airport historic temperatures (Number of Elements)
140000
120000
100000
80000
60000
40000
20000
0
100 400 700 1000 1300 1600 1900 2200 2500 2800 3100 3400 3700 4000 4300 4600 4900
BubbleSortTimeInMicrosec
insertionSortTimeInMicrosec
selectionSortTimeInMicrosec
quickSortTimeInMicrosec
mergeSortTimeInMicrosec

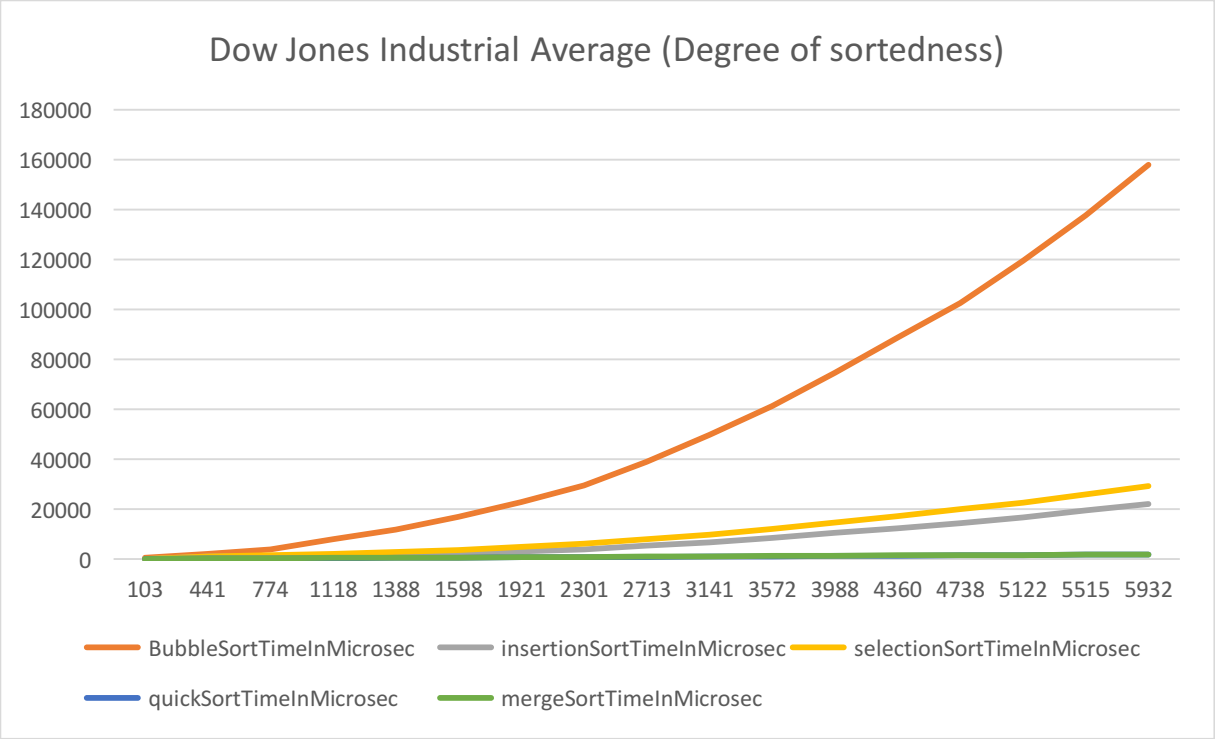

McGuire Airport historic temperatures (Number of Elements)
70000
60000
50000
40000
30000
20000
10000
0
100 400 700 1000 1300 1600 1900 2200 2500 2800 3100 3400 3700 4000 4300 4600 4900
BubbleSortMemInKb
insertionSortMemInBytes
selectionSortMemInBytes
quickSortMemInBytes
mergeSortMemInBytes

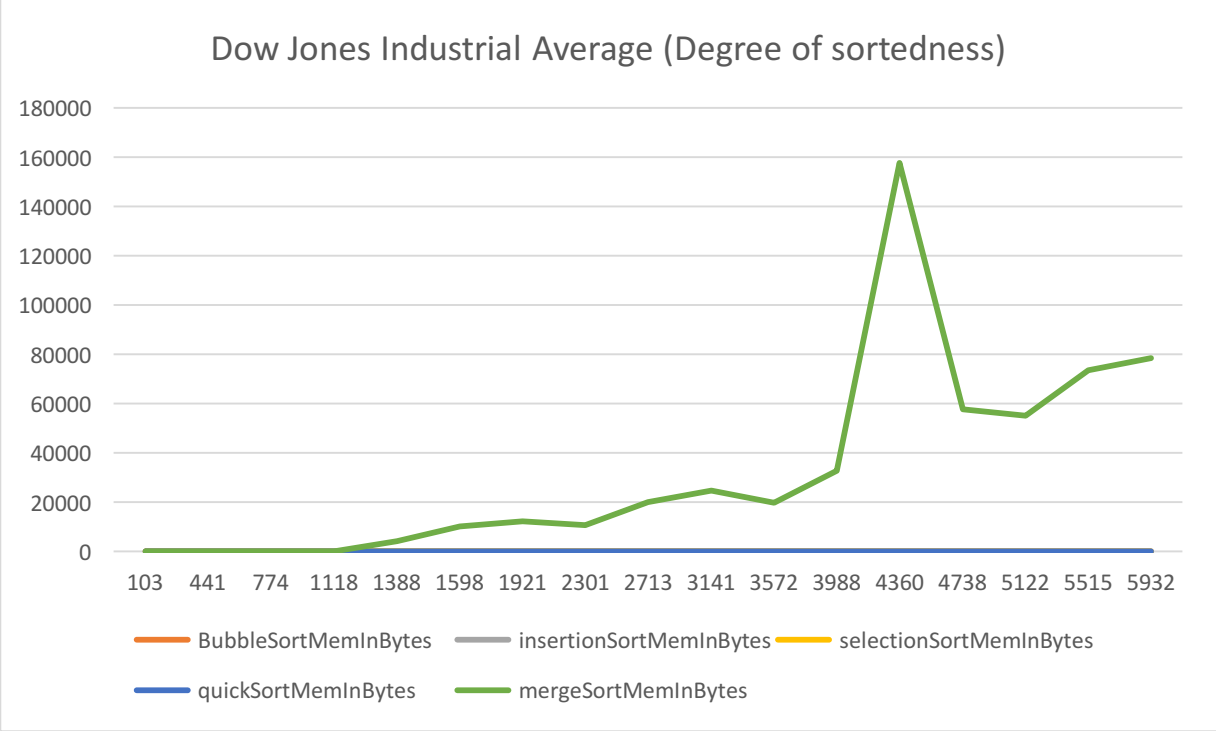

McGuire Airport historic termperatures (Degree of Sortedness)
140000
120000
100000
80000
60000
40000
20000
0
64 298 500 714 970 1209 1436 1629 1842 2081 2335 2575 2779 2992 3216 3445 3716
BubbleSortTimeInMicrosec
insertionSortTimeInMicrosec
selectionSortTimeInMicrosec
quickSortTimeInMicrosec
mergeSortTimeInMicrosec


McGuire Airport historic termperatures (Degree of Sortedness)
70000
60000
50000
40000
30000
20000
10000
0
64 298 500 714 970 1209 1436 1629 1842 2081 2335 2575 2779 2992 3216 3445 3716
BubbleSortMemInBytes
insertionSortMemInBytes
selectionSortMemInBytes
quickSortMemInBytes
mergeSortMemInBytes


Dow Jones Industrial Average (Number of Elements)
180000
160000
140000
120000
100000
80000
60000
40000
20000
0
100 400 700 1000 1300 1600 1900 2200 2500 2800 3100 3400 3700 4000 4300 4600 4900
BubbleSortTimeInMicrosec
insertionSortTimeInMicrosec
selectionSortTimeInMicrosec
quickSortTimeInMicrosec
mergeSortTimeInMicrosec


Dow Jones Industrial Average (Number of Elements)
180000
160000
140000
120000
100000
80000
60000
40000
20000
0
100 400 700 1000 1300 1600 1900 2200 2500 2800 3100 3400 3700 4000 4300 4600 4900
BubbleSortMemInKb
insertionSortMemInBytes
selectionSortMemInBytes
quickSortMemInBytes
mergeSortMemInBytes


Dow Jones Industrial Average (Degree of sortedness)
180000
160000
140000
120000
100000
80000
60000
40000
20000
0
103 441 774 1118 1388 1598 1921 2301 2713 3141 3572 3988 4360 4738 5122 5515 5932
BubbleSortTimeInMicrosec
insertionSortTimeInMicrosec
selectionSortTimeInMicrosec
quickSortTimeInMicrosec
mergeSortTimeInMicrosec


Dow Jones Industrial Average (Degree of sortedness)
180000
160000
140000
120000
100000
80000
60000
40000
20000
0
103 441 774 1118 1388 1598 1921 2301 2713 3141 3572 3988 4360 4738 5122 5515 5932
BubbleSortMemInBytes
insertionSortMemInBytes
selectionSortMemInBytes
quickSortMemInBytes
mergeSortMemInBytes

(1) Insertion Sort :

The algorithm checks a number with the values in its previous index and places the number in the right position. This has the worst case of $O(n^2)$ running time. It can be used for a smaller data set or where the array is sorted mostly.

(2) Selection Sort :

The algorithm searches for the lowest number and places it in the starting of the array and increases its index. This has the worst case running time of $O(n^2)$. This can be used for a smaller data sets.

(3) Bubble Sort :

This algorithm compares the concurrent two indexes n times and swaps if necessary. It also has the worst case of $O(n^2)$. This can be used for a smaller data sets.

(4) Merge Sort :

It uses divide and conquer approach. It splits the data into two subsets recursively and sorts them and merges them. It has a running time of O(n logn) and can be useful to be used in places where the data set is large.

(5) Quick Sort :

This algorithm selects a pivot and compares the values to it and swaps accordingly. It is best to use for average sized data sets. When the data set size is large, the performance of it degrades. The average running time of it is O(n logn) and the worst case running time is $O(n^2)$.

Data Sets used:

1) Synthetic data for uniform distribution
2) Synthetic data for normal distribution
3) McGuire Airport Historic temperature
4) Dow Jones Industrial Average

Method of Sortedness Used:

Inversion