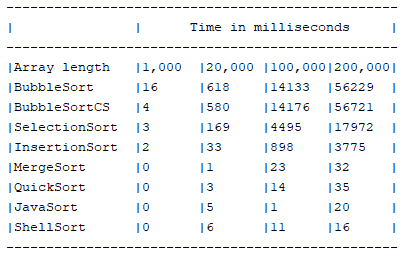
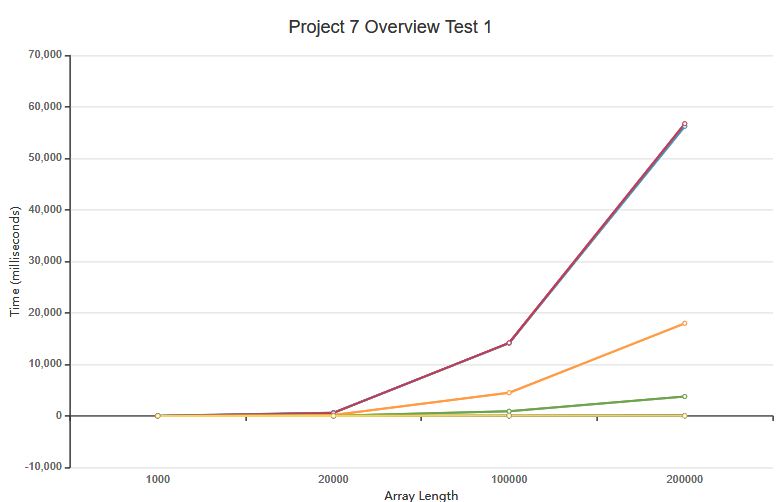
Name: Logan Lafauci

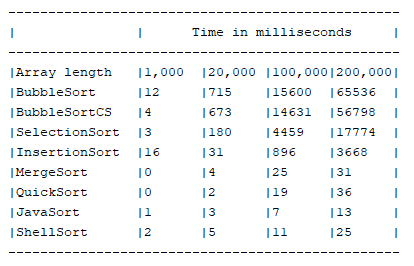
LSU ID: llafau1

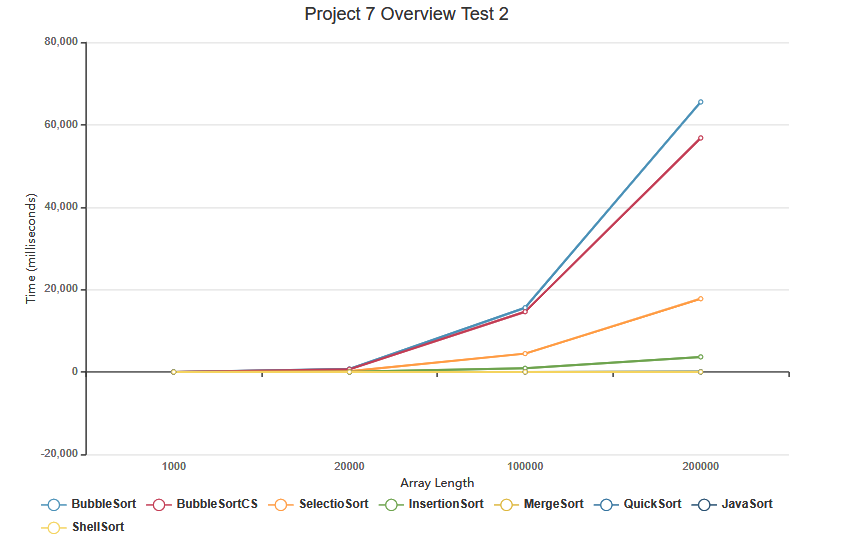
Test 1:





Test 2:





The complexity in the sorting algorithm I choose is O(n^2) at its worst and O(nlogn) at its best. Now the reason why my algorithm runs as fast as it does is due to the fact that it divides multiple parts of the array, sorts it and then moves on to another divided part. It is following similar principles as the merge and quicksort which is the reason why it’s time is comparable to those certain sorts.

One thing I would like to mention is that BubbleSortCS has the potential to save a lot of time but it also has the potential to not save any time at all given the nature of how an array is set up and that is evident in the above diagrams. BubbleSortCS is the most notable but you can see similar differences in the faster running programs as well when you compare test 1 and test 2 to each other which is interesting because some sorting algorithms will be faster and some will be slower given the random numbers there are.