# CSS 342 B: Program 4 Lab Report

---

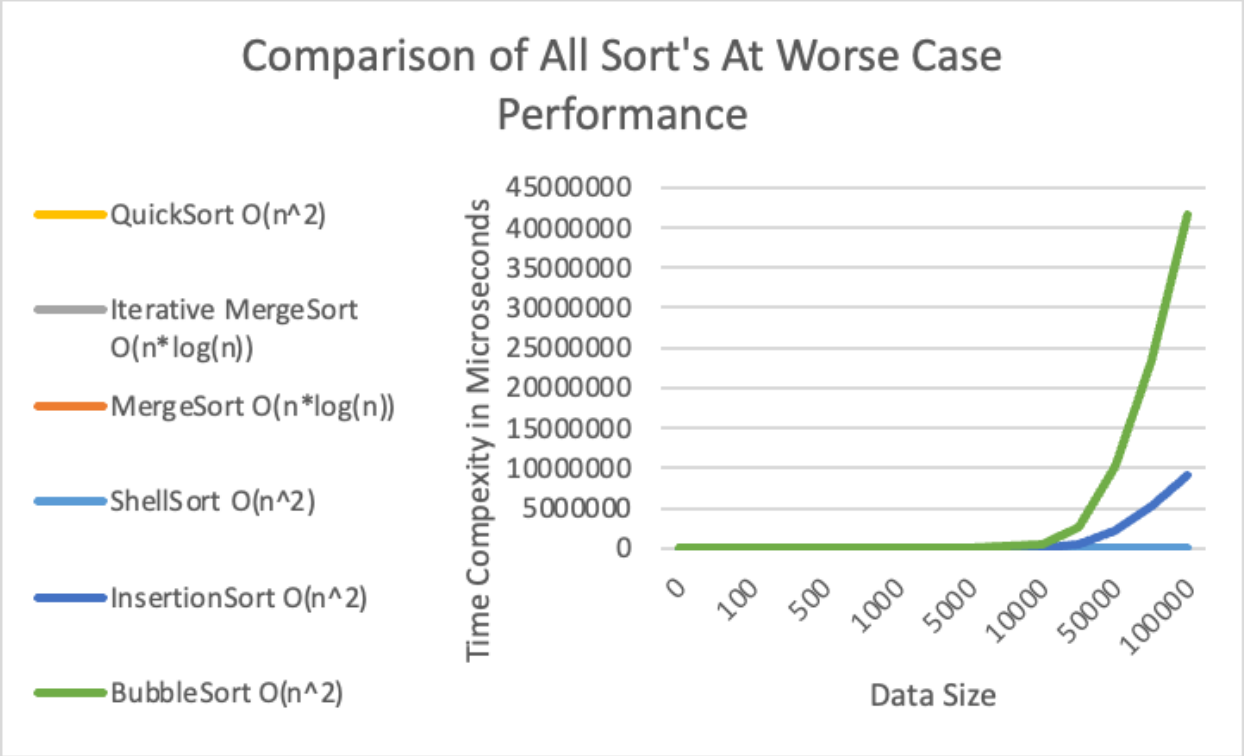
Prepared By: Hannah Pham
Date Prepared: Nov 22, 2022

## Analysis

This is an overview comparison of Insertion Sort, Merge Sort, Iterative Merge Sort, Quick Sort, Shell Sort, and Bubble Sort in their worst-case scenarios or Big-O notation. All sorts have similar time complexity when the data size is small. But as the data size increases, Bubble Sort and Insertion Sort take more time to perform. As can be seen in Figure 1, when the data size is above 10,000, Bubble Sort grows quadratically, and Insertion Sort, which is a little faster than Bubble Sort, also has a similar upward curve.

Thus, it is best to not use Bubble Sort or Insertion Sort to sort big amounts of data.
On the contrary, Quick Sort, Iterative Merge Sort, Merge Sort, and Shell Sort takes the least amount of time to perform big data size. It is recommended to use Quick Sort for the best performance.

**Figure 1**
*Comparison of All Sort's Performance*



# CSS 342 B: Program 4 Lab Report

## Analysis

This is a comparison between Merge Sort and Iterative Merge Sort. Both Merge Sort and Iterative Merge Sort use Divide and Conquer mechanism to sort a data set.

Merge Sort is a recursive algorithm and uses functions to call stack. And because of this, the program uses more space for operation.

As can be seen in Figure 2, the two sorts grow similarly in time complexity as the data size grows, although Iterative Merge Sort performs slightly faster.
Iterative Merge Sort is slightly faster than Merge Sort because it doesn't call recursive functions, and thus uses less space in the program.

**Figure 2**
*Comparison Between Merge Sort and Iterative Merge Sort*