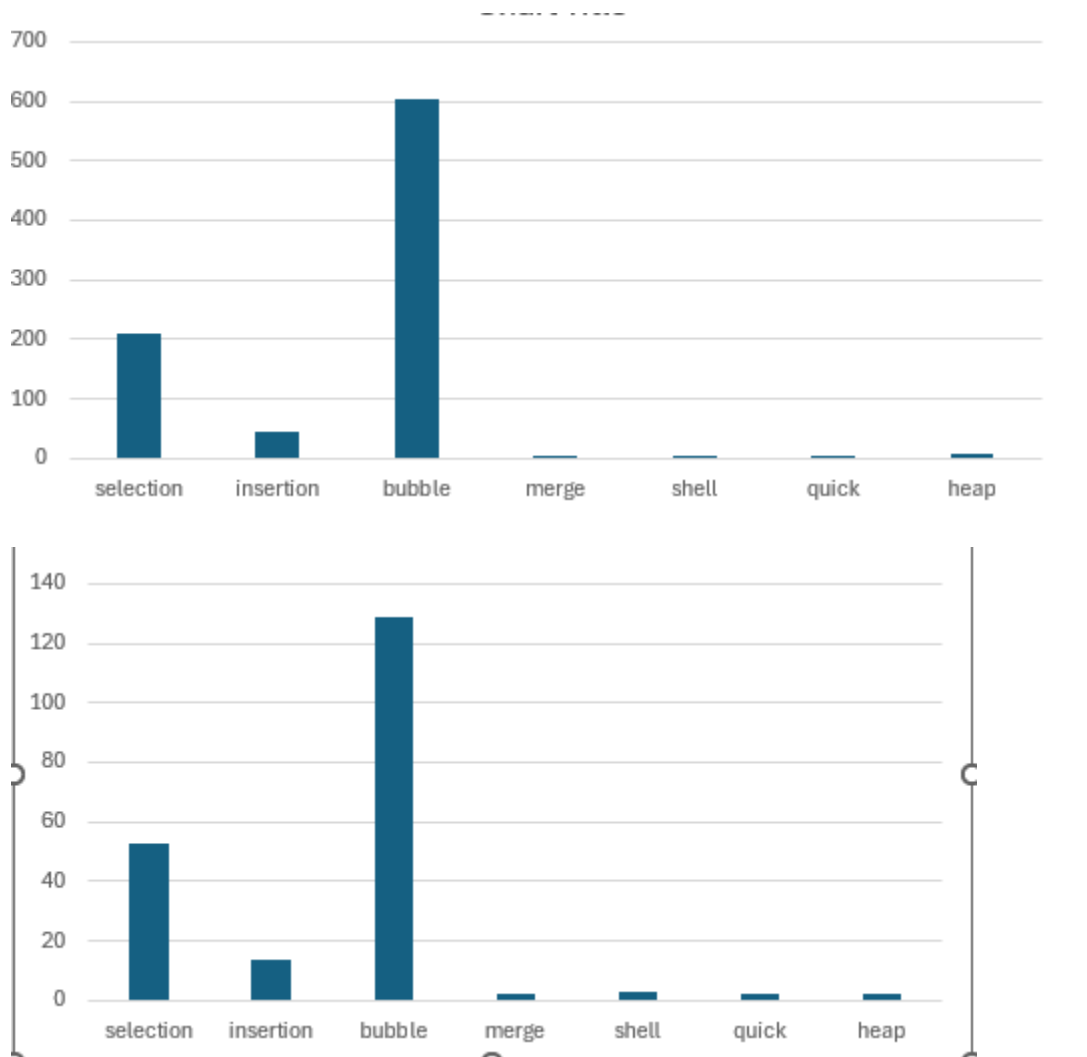Emanuel Espada
12/8/24
Professor Byron Hoy, Data Structures and Algorithms I

      In the code we were tasked to make seven different algorithms were put to the test in order to see which of the seven could sort an array of data the fastest. For my testing I used an array of ten thousand and another array of twenty thousands elements full of numbers ranging from 0 to 9,999.

      The graphs show the findings of the testing in units of milliseconds. From the graph we can see right off of the bat that bubble and selection were the worst when it came to sorting the arrays quickly with bubble taking 604 ms on the trial with 20,000 elements and 129 ms with the trial involving 10,000 elements. Selection was the second worst option when it came to time. On the other hand quicksort was the fastest but a very small amount just beating merge sort. In the

first trial with 10,000 elements quicksort took only 1.88 ms and on the second trial it took 3.4ms. We can see that as the amount of elements that need to be sorted rises so does the time the algorithms take to sort them. I believe that quicksort was able to perform so well was because of its in place sorting which allows it to only require a small amount of additional storage while it is sorting.