Program 1: Insertion-Sort and Merge-Sort
CSCI 430, Fall 2017

John E. Buckley III

September 17, 2017

Implementation:

The implementation of these algorithms were very similar to the pseudocode from the book and class notes. The main difference being: knowing what imports needed for your methods of implementing it, and how you would implement the randomly generated numbers, the ordered numbers, and reverse numbers, and how you would take n as a command line parameter.

Testing Expectations:

I expected the ordered data for insertion sort to be linear and the reverse ordered data to quadratic because, as discussed in class, for the best case scenario we got $\Theta(n)$ and for the worst case we got $\Theta(n^2)$. Since the randomly generated numbers are not quit in sorted or reverse order I assumed this line would land between these two cases.

Testing Observations and Analysis:

My assumptions of what each case would look like on a graph were correct, however all my times were slightly higher than those shown in the sample spreadsheet, but the results were the same.

Take-Away:

Recursion is my friend. Although it may look like more work on the face of it, doesn't actually means it is. Merge-Sort looked as if it would take longer as it took longer to implement compared to Insertion-Sort, however the results were that Merge-Sort is much more effective.