COP 3502
Noah Schrock
Assignment 3
11/10/20

1. Did the results of your program confirm what you already knew or suspected about these sorting algorithms? How?

   More than less yes it did. The slow algorithms were of course slow and they made much more comparisons than the faster ones. Bubble had a lot of swaps, selection had less swaps since it uses a value to determine whether to swap or not, insertion tended to have a lot of copies in it.

   The faster algorithms finished very fast and had a lower comparison count. Quick sort used low and high for pivot values and finished faster than the other algorithms, merge used a midpoint and splits them into smaller arrays and then rejoins them sorted, and merge-insertion used the the initial comparisons of the merge sort but then inserts them into the list already sorted like the insertion algorithm.

2. What differences did you see between and among the slow (bubble, selection, insertion) and fast (quick, merge, merge-insertion) algorithms?

   Like I mentioned earlier the fast ones were much faster than the slow ones, obviously because they are much more advanced. The differences that I noticed are what I said for the first part.

   - Bubble sort had a lot of swaps
   - Selection sorts than bubble
   - Insertion had a lot of copies

   - Quick sort finished the fastest out of all of them
   - Merge used a midpoint and had a lot of block copies and less total copies
   - Merge insertion used two methods and had less block copies but a lot more copies than merge