**Summary**

For Deliverable 4, I chose to complete Junit-based, property-based testing. I chose to test this over the combinatorial testing because we used Junit testing in Deliverable 2 and, now that I have a better understanding for Junit tests, I wanted to choose the assignment that can show this. I also like completing testing for properties rather than testing specific methods, which is another reason I chose this assignment.

I began working on this deliverable by creating the class script. I wrote a class that when the object is initialized, an array list containing a hundred arrays is created. Each array varies in length from one to one hundred. After I had written the constructor, I began creating Junit tests for three properties: the length of the unsorted array needs to equal the length of the sorted array, the last element in the sorted array must be larger than or equal to the first element in the sorted array, and all the elements in the sorted array are found in the unsorted array.

The biggest issue I faced when working on this deliverable was the Junit test producing fails after enough executions. Originally when I made this deliverable, I didn’t take into account that the length of each array is determined by a random variable, so if the random variable is 0, the array will remain in the array list uninitialized, which caused the unit test for checking if the last element is greater than or equal to the first element to fail. Luckily, this was easy enough to fix. I added a while loop after the length of the array was determined so that a new random number will keep being given until the value is not zero.

The main thing I learned from this deliverable is about the asserts. I was concerned that since I put an assert inside the for loop and the assert is determined as true, it would stop and consider the test as a pass and wouldn’t re-enter the for loop. I was easily able to check this and learned that the test will only stop the test if the test fails, otherwise, it will the assert as true, re-loop, and continue until it has reached the end of the test without any problems.