Adam Darr

CS 1632 – DELIVERABLE 4

PROPERTY-BASED TESTING

https://github.com/adamdarr/deliverable-4

**Summary**

For this assignment, I chose to use property-based testing instead of combinatorial testing. The reason being is not because I prefer property-based testing over combinatorial testing, but because I wanted to explore what it would be like to test a built-in java method like the Arrays.sort() method. Through doing this assignment, I was able to learn more about the intricacies of property-based testing.

My first step in the process of doing property-based tests was looking over the course slides and learning more about the properties that an array sorting method should have. These properties include the output array having the same length as the input array, the values in the output arrays are always decreasing and never increasing, every element in the original array is in the output, no element not in the original array is in the output array, running the algorithm again should not change the output array, and running the algorithm twice should always produce the same resulting output array. For my tests, I narrowed the properties down to three: (1) the output array is always the same size as the array that is passed in, (2) every element in the input array is in the output array, and (3) values in the output array are always increasing or equal. I split these properties into three separate test cases and used the JUnit @Before tag to initialize random length and value arrays before each case.

Personally, I did not run into many problems in this assignment. In fact, the hardest part was writing the method to set up the random length and value arrays. To do this, I used a two dimensional array to store 100 individual arrays. Then I randomized the length and values with Java’s built in random utilities. I chose to allow the values stored in the array to be anywhere from 0 to 2^31 – 1 since I wanted to test a large variety of possible values and ensure that the properties held even with high or low values.

Overall, I was able to learn a lot from this assignment. Testing for properties forced me to think in a different way than I had previously thought of unit tests. This allowed me to gain a solid understanding of how sorting methods should work in terms of their properties, not just their inputs and their outputs. I believe that thinking from the perspective of the properties of a sort method allowed me to create something more robust than what is typical of the JUnit tests I have written in the past.

**Screenshot of Executed JUnit Tests**

