IS2545 - DELIVERABLE 4 - PROPERTY-BASED TESTING

Name: Xin Wen

Git: <https://github.com/Seloen/Test1111/tree/master/Deliverable_4>

**Summary:**

In this deliverable, the “billify()” method is written to map the “squareIt()” function through an array accepted as parameter of “billify()” method. That array is randomly generated by RandomGenerator() function. I simply used a for loop to complete the mapping process and then return a new array whose last element represents sum of all squared elements.

I came up with 5 properties to test in for this program, they are:

* *Output array size should be exactly one bigger than input array size.*

Since the last element of output array is the sum of all other elements, size of output should always be one larger than input.

* *Elements of input array should be respectively mapped the square to output array.* *(Except for the last element in the output array)*

This program is designed to square elements, so the squareIt() function has to work well.

If this test passed, then we can ensure two things: billify method works and the squareIt function is correct.

* *The last element of output array is the sum of all previous elements.*

The code to calculate sum of all other elements should always work.

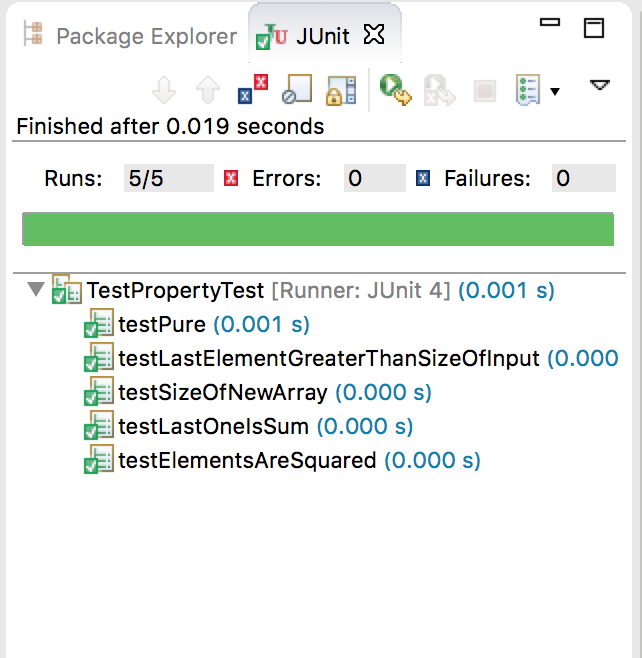
* *Running it twice on same array should always has same output.*

Purity of the program has to be ensured.

* *Last element of output array should always equals to or greater than size of input array. (Considering of the edge case when every element is 1)*

One of the edge case is that all elements of the randomly generated array are 1, which causes the result of mapping square function is still an array of 1s. Therefore the minimum value of the sum is the input array length.

The only one failure I encountered was I used “assertEquals” to assert two arrays during the test of purity. Then



I realized that I should use “assertArrayEquals” instead.