```
//Testing Project 2 - Harley Phung

import org.junit.*;
import static org.junit.Assert.*;
import java.util.NoSuchElementException;
public class HW2tester {



    /**
     * Test the average calculation of the single array
     */
    @Test
    public void testSingleAverage() {
        double[] input1 = {1.1, 1.2, 1.3, 1.4};
        double[] input2 = {1.25, -1.50, 1.75, -2.0, 2.25, 2.50, 2.75};
        double[] input3 = {};
        double[] input4 = {2.5};
        //Try catch block for NoSuchElementException
        boolean thrown = false;
        try {
            HW2.average(input3); //test 0
        } catch (NoSuchElementException e) {
            thrown = true;
        }
        assertEquals(1.25, 1.25, HW2.average(input1));   //Test many that only have
positive value
        assertEquals(1.00, 1.00, HW2.average(input2));   //Test many that have
negative value
        assertTrue(thrown); //test 0
        assertEquals(2.5,2.5,HW2.average(input4)); //Test one
    }

    /**
     * Test the average calculation of the 2 dimension array
     */
    @Test
    public void testDoubleAverage() {
        double[][] input2d1 = new double[][] {{1.1, 1.2, 1.3}, {1.4, 1.5}};
        double[][] input2d2 = new double[][] {{1.1, -1.2}, {1.4, 1.6, 2.1}};
        double[][] input2d3 = new double[][] {{-1}, {1}};
        double[][] input2d4 = new double[][] {};
        //Try catch block for NoSuchElementException
        boolean thrown = false;
        try {
            HW2.average(input2d4); //test 0
        } catch (NoSuchElementException e) {
            thrown = true;
        }
        assertEquals(1.3, 1.3, HW2.average(input2d1));//test many
        assertEquals(1.0, 1.0, HW2.average(input2d2));//test many with negative
input
        assertEquals(0.0, 0.0, HW2.average(input2d3));// test one each row of the
array
        assertTrue(thrown);
    }

    /**
     * Test the number of words counted
     * */
    @Test
    public void testCountWords() {
        String firstString = new String(" ");
        String secondString = new String("one");
        String thirdString = new String("  one"  );
        String fourthString = new String("One fish, two fish, red fish , blue
fish !");
        String fifthString = new String("   One fish, two fish,    red fish , blue
fish    !  ");
        assertEquals(0, 0, HW2.countWords(firstString)); //Test no words
        assertEquals(1, 1, HW2.countWords(secondString)); // Test one word without
whitespaces
        assertEquals(1, 1, HW2.countWords(thirdString)); // Test one word with
whitespaces
        assertEquals(10, 10, HW2.countWords(fourthString)); //Test many words in a
String without whitespaces
        assertEquals(10, 10, HW2.countWords(fifthString)); //Test many words in a
String with many whitespaces in between characters
    }

    /**
     *Test if the String return to specific position.
     */
    @Test
    public void testTruncate() {
        String firstString1 = new String ("  ");
        String secondString1 = new String("This   ");
        String thirdString1 = new String("     excessively long");
        String fourthString1 = new String("Is this homework fun?");
        String fifthString1 = new String("        Hi       ?");
        assertEquals("Test no words with longer desired length", "  ",
HW2.truncate(firstString1, 10)); //Test no words with longer desired length
        assertEquals("Test one word with whiteSpaces", "This",
HW2.truncate(secondString1, 1)); //Test one word with whiteSpaces
        assertEquals("Test many words but short desired length", "
excessively", HW2.truncate(thirdString1, 17)); // test many words but short desired
length
        assertEquals("Test many words with normal whitespace", "Is this",
HW2.truncate(fourthString1, 10)); // test many words with normal whitespace
        assertEquals("Test many words with normal whitespace", "Is this homework",
HW2.truncate(fourthString1, 17)); //test many words with normal whitespace
        assertEquals("One word with test many whiteSpaces", "        Hi",
HW2.truncate(fifthString1,1)); // test one word with many whiteSpaces
    }

    /**
     * Test if the String return with evenly added whitespaces
     */
    @Test
    public void testPadString() {
        String firstString2 = new String("  ");
        String secondString2 = new String("This  ");
        String thirdString2 = new String("This is really fun!");
        String fourthString2 = new String("This   is really fun!");
        String fifthString2 = new String("This is   really fun!");
        String sixthStirng2 = new String("This is really fun!   ");
        assertEquals("Test 0", "  ", HW2.padString(firstString2,4)); //Test 0
        assertEquals("Test 1 with smaller desired length", "This  ",
HW2.padString(secondString2, 1)); //Test 1 with smaller desired length
        assertEquals("Test 1 with smaller input length", "This  ",
HW2.padString(secondString2, 10)); //Test 1 with smaller input length
        assertEquals("Test many with normal string whitespace", "This is really
fun!", HW2.padString(thirdString2, 2)); //Test many with normal string whitespace
        assertEquals("Test many with smaller string length", "This  is  really
fun!", HW2.padString(thirdString2, 22)); //Test many with smaller string length
        assertEquals("Test many with unequal added whitespace", "This  is  really
fun!", HW2.padString(thirdString2, 23)); //Test many with unequal added whitespace
        assertEquals("Test many with more whitespace at first", "This  is  really
fun!", HW2.padString(thirdString2, 22)); //Test many with more whitespace at first
    }
}
```