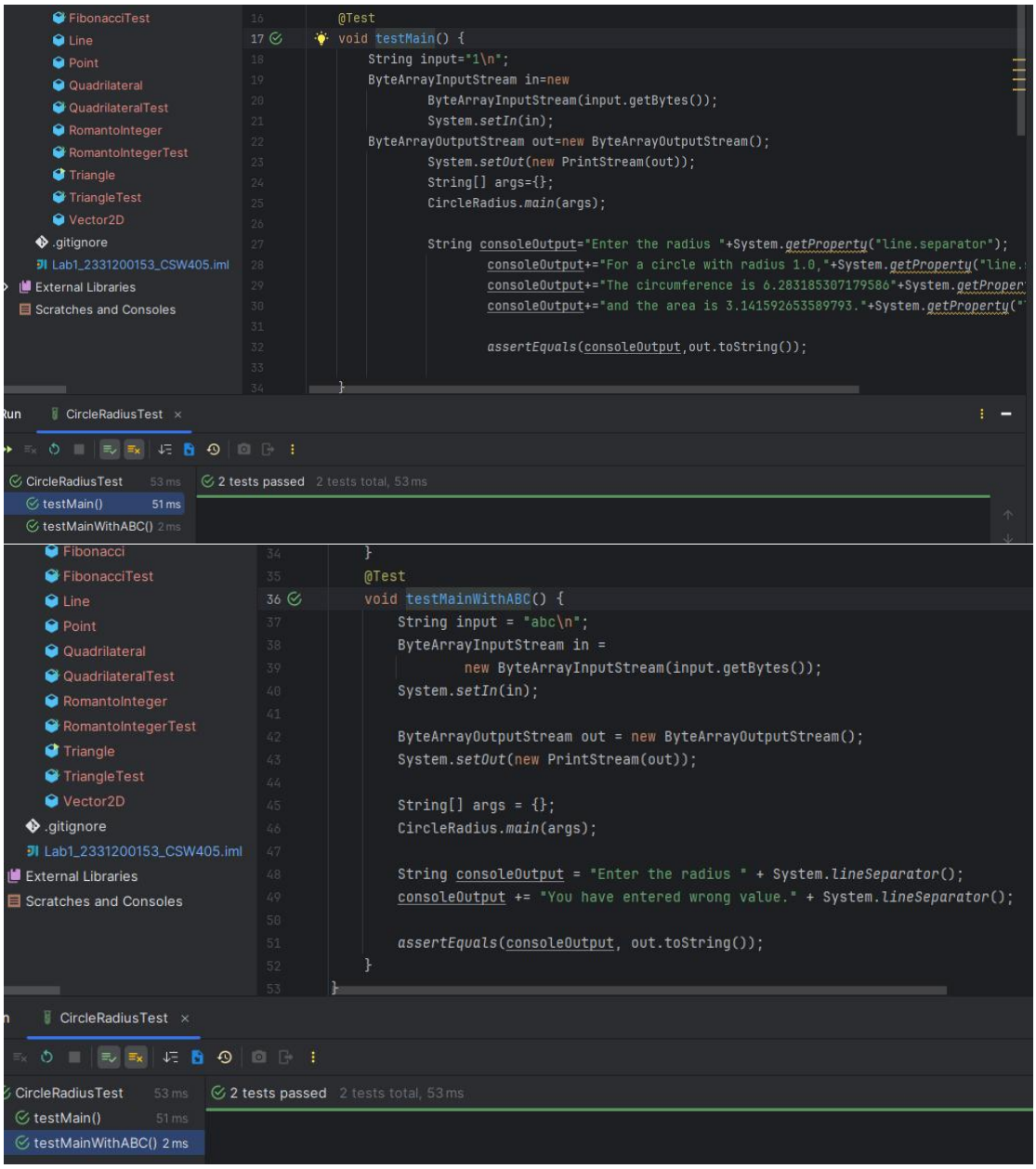
Phạm Trần Gia Hưng MSSV:2331200153

1. Use the attached files “Circle-Radius.java” and “Circle-Radius-Test.java”. Add the code in the main function to accommodate the last test case that is when test data is "abc"

Bug:

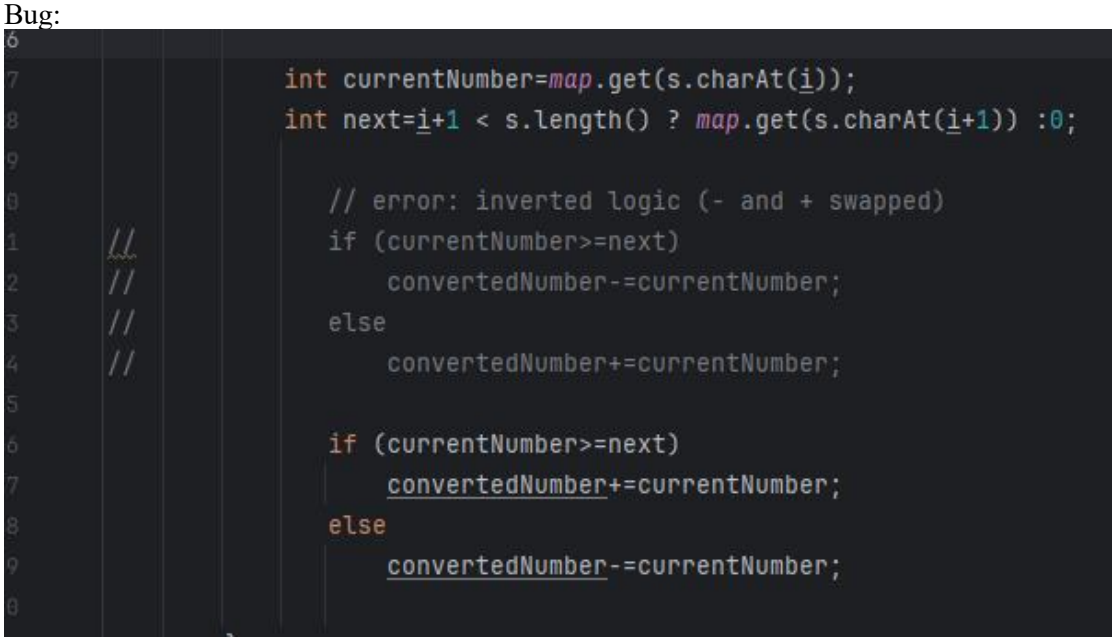
```
        } else
            System.out.println("You have entered wrong value.");
        // bug fix: add exception handling for input like "abc"
        // the code didnt have input reader
    } catch (Exception e) {
        System.out.println("You have entered wrong value.");
    }
```

Test:

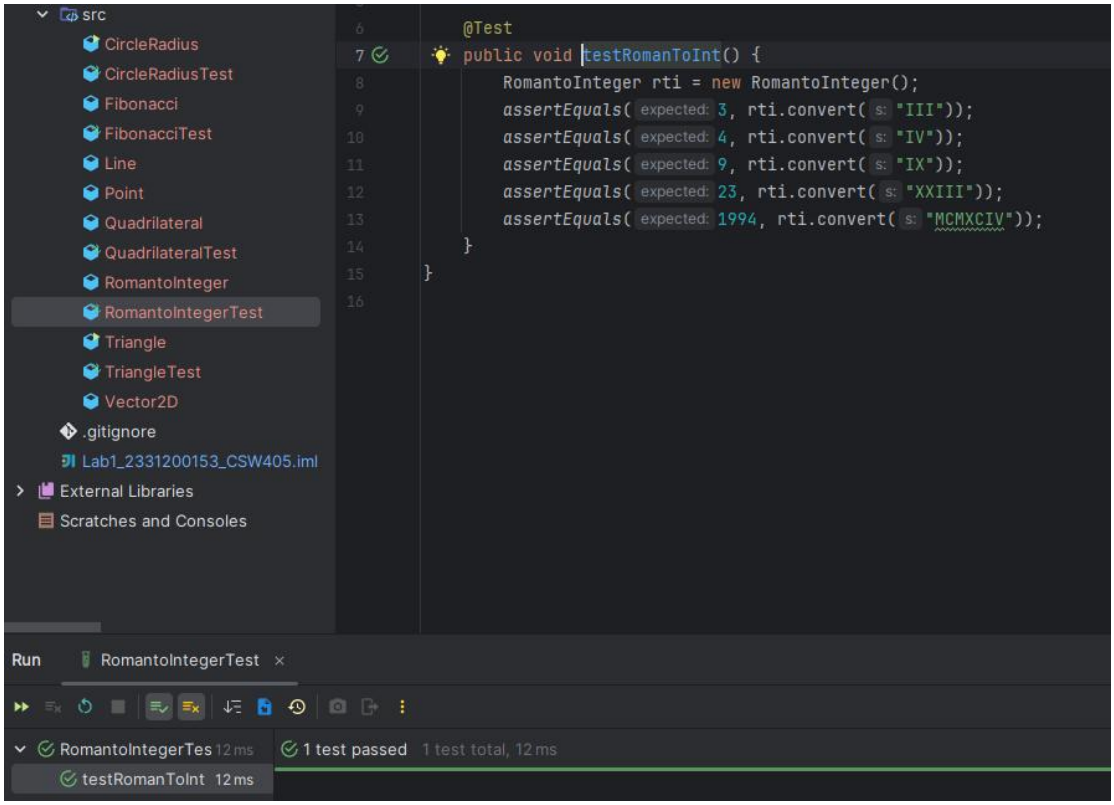


2. Consider the file “RomantoInteger.java”. This program implements a program that converts a Roman numeral that is in form of a String, to an integer. The Roman numeral system is based on seven essential numerals: I, V, X, L, C, D, and M (1, 5, 10, 50, 100, 500, and 1,000, respectively). So, rules are
a. We can’t add more than three of the same Roman numeral together
b. If a numeral comes after a numeral that is larger or equal in value, then it must be added to the numeral before it.
c. Smaller numeral placed before a larger numeral indicates subtraction of that smaller numeral from the larger one.
d. We can’t subtract more than one value from a Roman numeral. Example of test cases for this program:

Bug:

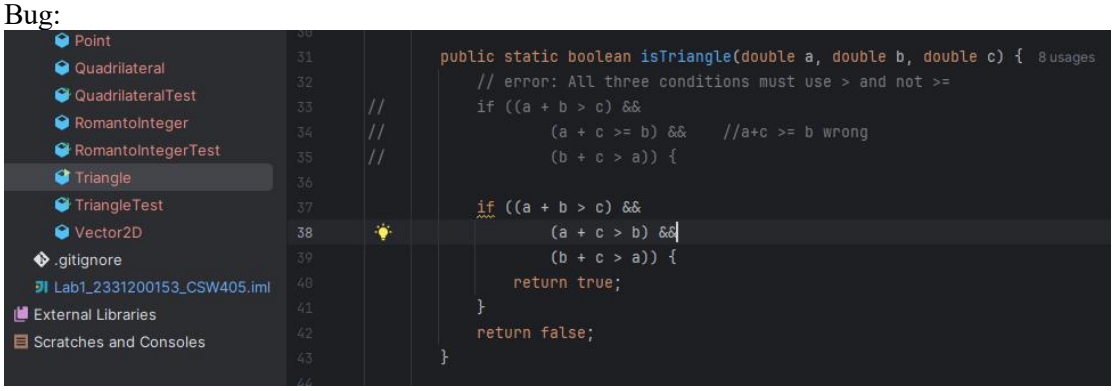


Test:

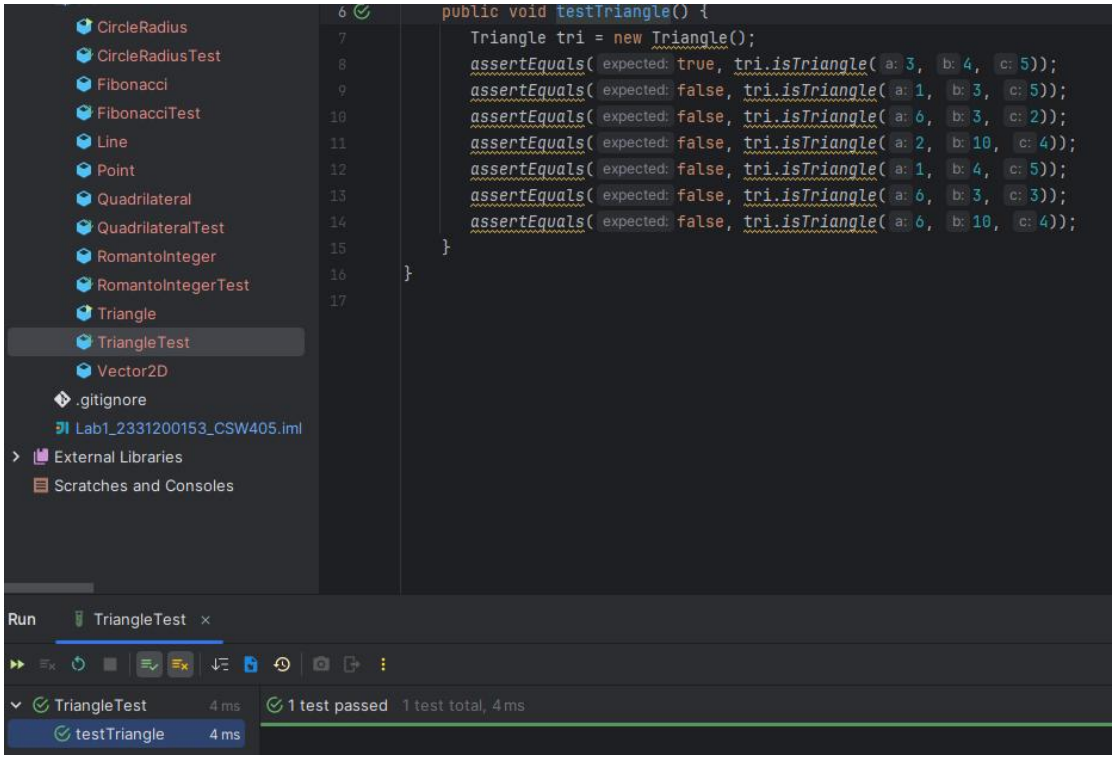


3. Consider Triangle.java file. Test the main method and isTriangle() method

Bug:

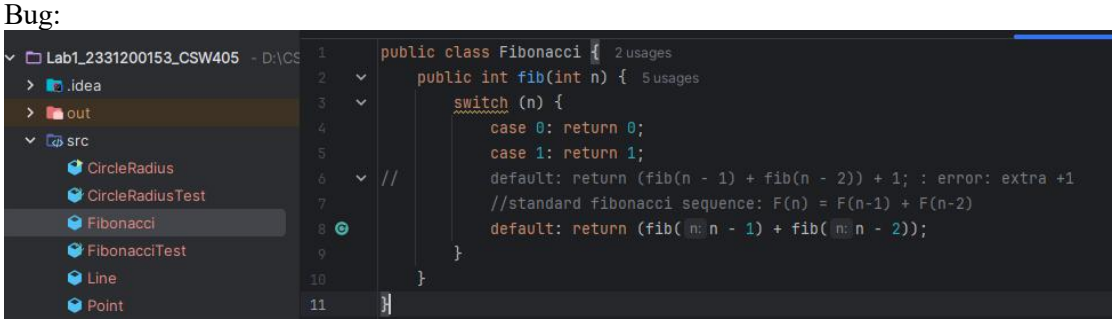


Test:

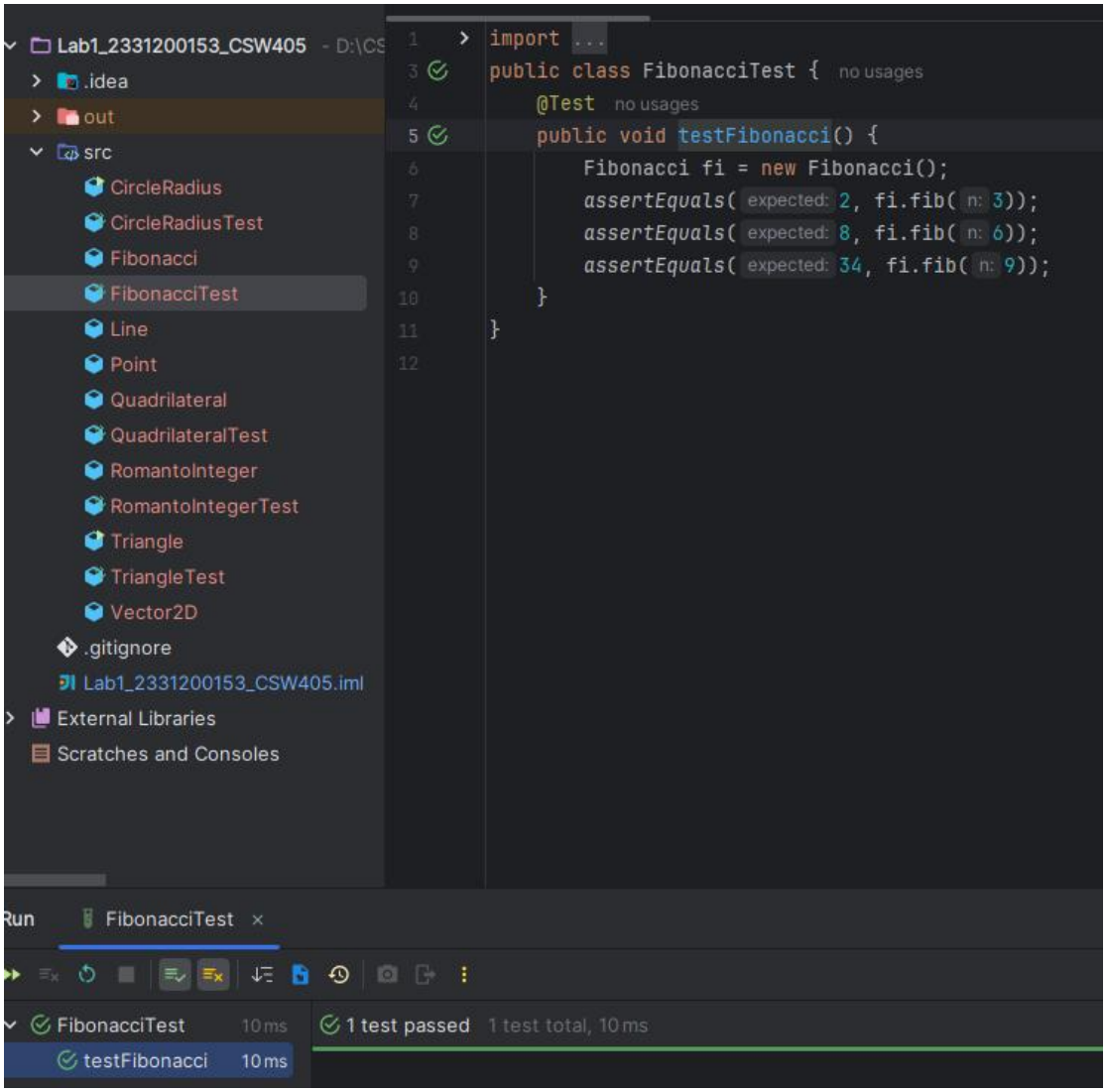


4. Look at the Fibonacci class. This class is an attempt at implementing the recursive method fib, which should generate the nth Fibonacci number. Create test cases for this class and run those test cases using Junit

Bug:



Test:



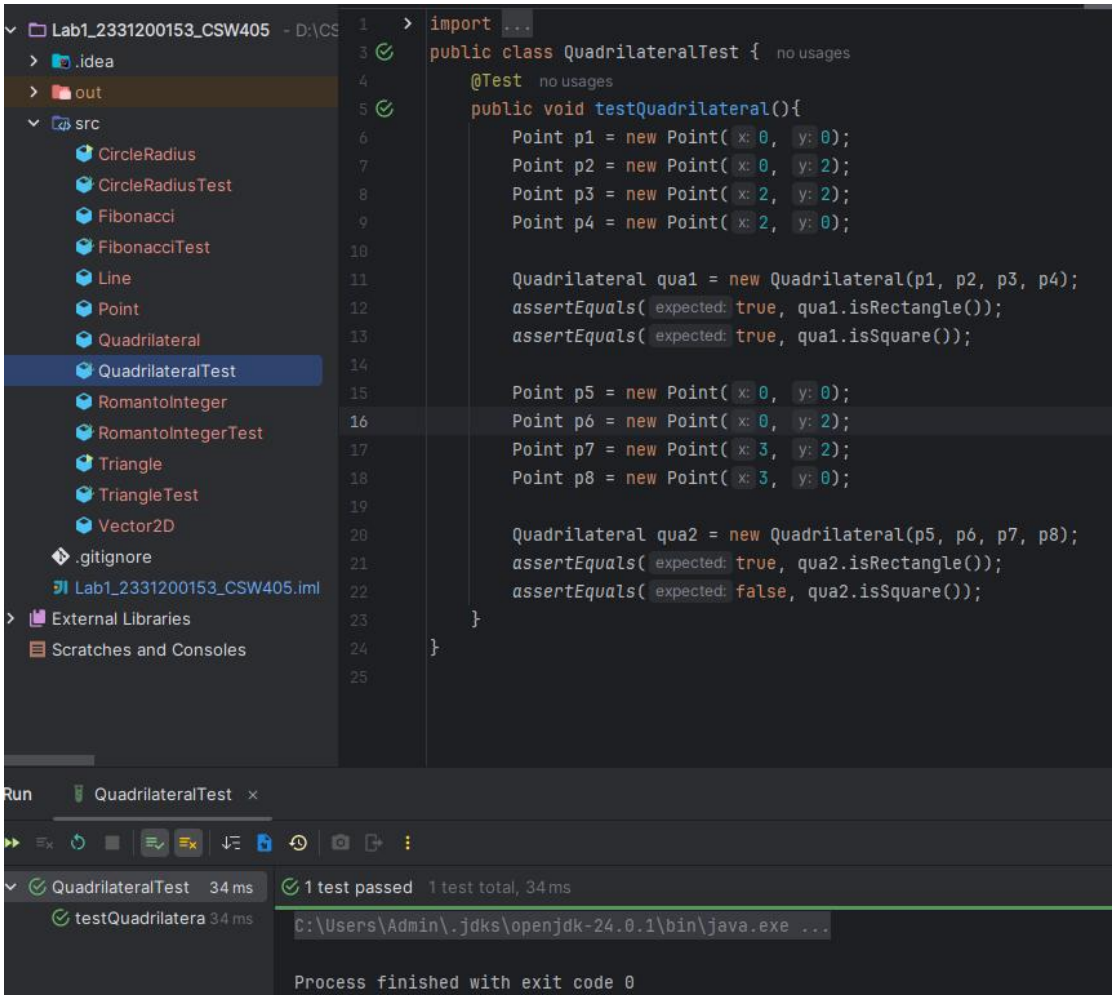
5. The Quadrilateral class denotes a polygon with four sides. It has two methods, isRectangle() and isSquare(). Furthermore, it also uses the classes Point, Line and Vector2D. To find if the polygon is a rectangle, vectors and dot products are used to determine if every corner forms a right angle. To find if the polygon is a square, isRectangle() is used and check if the lengths of all sides are equal

Bug:

```
    public Boolean isSquare() {  2 usages
//      return (isRectangle() &&
//              l1.isSameLengthAs(l3)); -- // error: only checks opposite sides
        return (isRectangle() &&
                l1.isSameLengthAs(l2) &&
                l2.isSameLengthAs(l3) &&
                l3.isSameLengthAs(l4));
    }

}
```

Test:


Lab1_2331200153_CSW405 - D:\CS
.idea
out
src
CircleRadius
CircleRadiusTest
Fibonacci
FibonacciTest
Line
Point
Quadrilateral
QuadrilateralTest
RomantoInteger
RomantoIntegerTest
Triangle
TriangleTest
Vector2D
.gitignore
Lab1_2331200153_CSW405.iml
External Libraries
Scratches and Consoles
import ...
public class QuadrilateralTest { no usages
@Test no usages
public void testQuadrilateral(){
Point p1 = new Point(x: 0, y: 0);
Point p2 = new Point(x: 0, y: 2);
Point p3 = new Point(x: 2, y: 2);
Point p4 = new Point(x: 2, y: 0);
Quadrilateral qua1 = new Quadrilateral(p1, p2, p3, p4);
assertEquals(expected: true, qua1.isRectangle());
assertEquals(expected: true, qua1.isSquare());
Point p5 = new Point(x: 0, y: 0);
Point p6 = new Point(x: 0, y: 2);
Point p7 = new Point(x: 3, y: 2);
Point p8 = new Point(x: 3, y: 0);
Quadrilateral qua2 = new Quadrilateral(p5, p6, p7, p8);
assertEquals(expected: true, qua2.isRectangle());
assertEquals(expected: false, qua2.isSquare());
}
}
Run
QuadrilateralTest
QuadrilateralTest 34 ms
testQuadrilatera 34 ms
1 test passed 1 test total, 34 ms
C:\Users\Admin\.jdks\openjdk-24.0.1\bin\java.exe ...
Process finished with exit code 0