Министерство науки и высшего образования Российской Федерации

ФГБОУ ВО «Кубанский государственный технологический университет»

(ФГБОУ ВО «КубГТУ»)

Кафедра информационных технологий и кибербезопасности

Кафедра информационных систем и программирования

**ОТЧЕТ**

по лабораторной работе №3.

Тема работы: «Тестирование методом черного ящика».

Наименование дисциплины: «Тестирование и отладка программного обеспечения».

Подготовил:

Студент группы 23-КБ-ПР1

Юданов М. Д.

Проверил:

Волик А. Г.

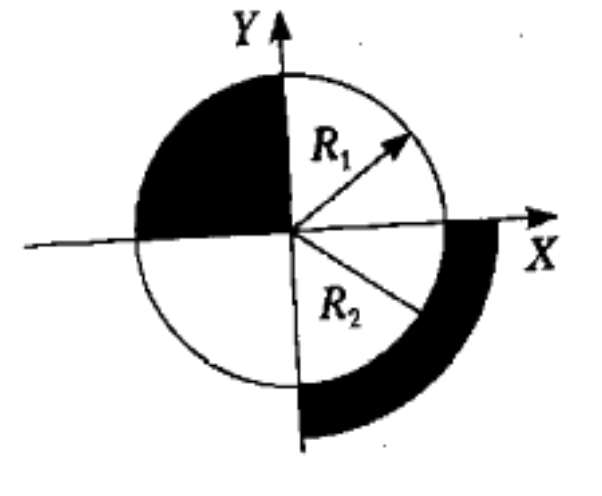
Краснодар

**2025**

**Цель работы**: изучить подход к тестированию методом черного ящика.

**Ход работы**:

1. Пишу код программы для класса, реализующего проверку принадлежности точки различным областям плоскости, соответствующих условию задачи.

****

**Код программы:**

import scala.math.\_

class CartesianPlane13(val R1: Double, val R2: Double) {

def isBelongToZone(x: Double, y: Double): Boolean = {

val distance = sqrt(x \* x + y \* y)

(x <= 0 && y >= 0 && distance <= R1) ||

(x >= 0 && y <= 0 && distance >= R1 && distance <= R2)

}

}

2. Определим классы эквивалентности и граничные значения для пересечений фигур.

Классы эквивалентности:

1. Точка в I четверти
2. Точка во II четверти внутри круга
3. Точка во II четверти за кругом
4. Точка в III четверти
5. Точка в IV четверти внутри круга
6. Точка в IV четверти внутри полоски
7. Точка в IV четверти за полоской

Граничные значения:

1. Начало координат
2. Точка между I и II четвертью (внутри круга)
3. Точка между I и II четвертью (на границе круга)
4. Точка между I и II четвертью (за кругом)
5. Точка во II четверти на границе круга
6. Точка между II и III четвертью (внутри круга)
7. Точка между II и III четвертью (на границе круга)
8. Точка между II и III четвертью (за кругом)
9. Точка между III и IV четвертью (внутри круга)
10. Точка между III и IV четвертью (на границе круга и полоски)
11. Точка между III и IV четвертью (внутри полоски)
12. Точка между III и IV четвертью (на верхней границе полоски)
13. Точка между III и IV четвертью (за полоской)
14. Точка в IV четверти на нижней границе полоски
15. Точка в IV четверти на верхней границе полоски
16. Точка между IV и I четвертью (внутри круга)
17. Точка между IV и I четвертью (на границе круга и полоски)
18. Точка между IV и I четвертью (внутри полоски)
19. Точка между IV и I четвертью (на верхней границе полоски)
20. Точка между IV и I четвертью (за полоской)

3. Составляю наборы тестовых данных для метода TestPoint в классе Area.

**Код тестов:**

import org.scalatest.funsuite.AnyFunSuite  
  
class CartesianPlane13Tests extends AnyFunSuite {  
  
 val *cartesianPlane* = new CartesianPlane13(5, 10)  
  
 val *testData*: Seq[(Double, Double, Boolean)] = Seq(  
 // Классы эквивалентности  
 (2, 1, false),  
 (-3, 3, true),  
 (-4, 4, false),  
 (-2, -2, false),  
 (2, -2, false),  
 (7, -3.5, true),  
 (9.5, -5, false),  
  
 // Граничные значения  
 (0, 0, true),  
 (0, 4.999, true), (0, 5, true), (0, 5.001, false),  
 (-3, 4, true),  
 (-4.999, 0, true), (-5, 0, true), (-5.001, 0, false),  
 (0, -4.999, false), (0, -5, true), (0, -7.5, true), (0, -10, true), (0, -10.001, false),  
 (3, -4, true), (8, -6, true),  
 (4.999, 0, false), (5, 0, true), (7.5, 0, true), (10, 0, true), (10.001, 0, false)  
 )  
  
 for (((x, y, expected), index) <- *testData*.zipWithIndex) {  
 test(f"Test #**$**{index + 1}: isBelongToZone(**$**x%.3f, **$**y%.3f) should be **$**expected") {  
 val result = *cartesianPlane*.isBelongToZone(x, y)  
 assert(result == expected, s"Expected **$**expected but got **$**result for (x=**$**x, y=**$**y)")  
 }  
 }  
}

4. Запускаю тесты и фиксирую результаты.

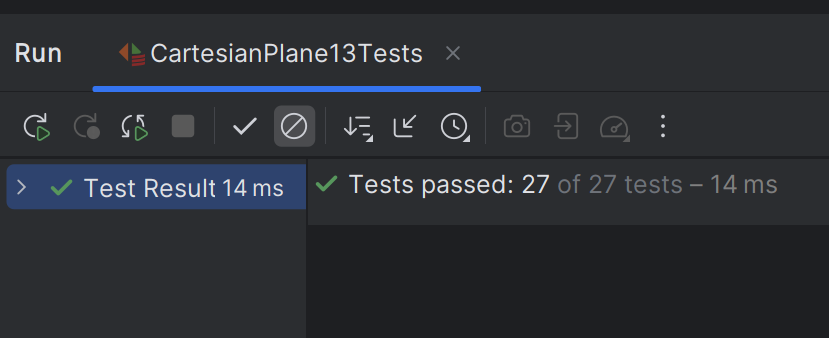


Рисунок 1 – Результаты тестирований.

5. Делаю вывод касательно проделанной работы.

**Вывод:** в ходе выполнения лабораторной работы я изучил подход к тестированию методом черного ящика. Также научился тестировать методы классов на основе тестового набора с использованием программных отладочных средств.