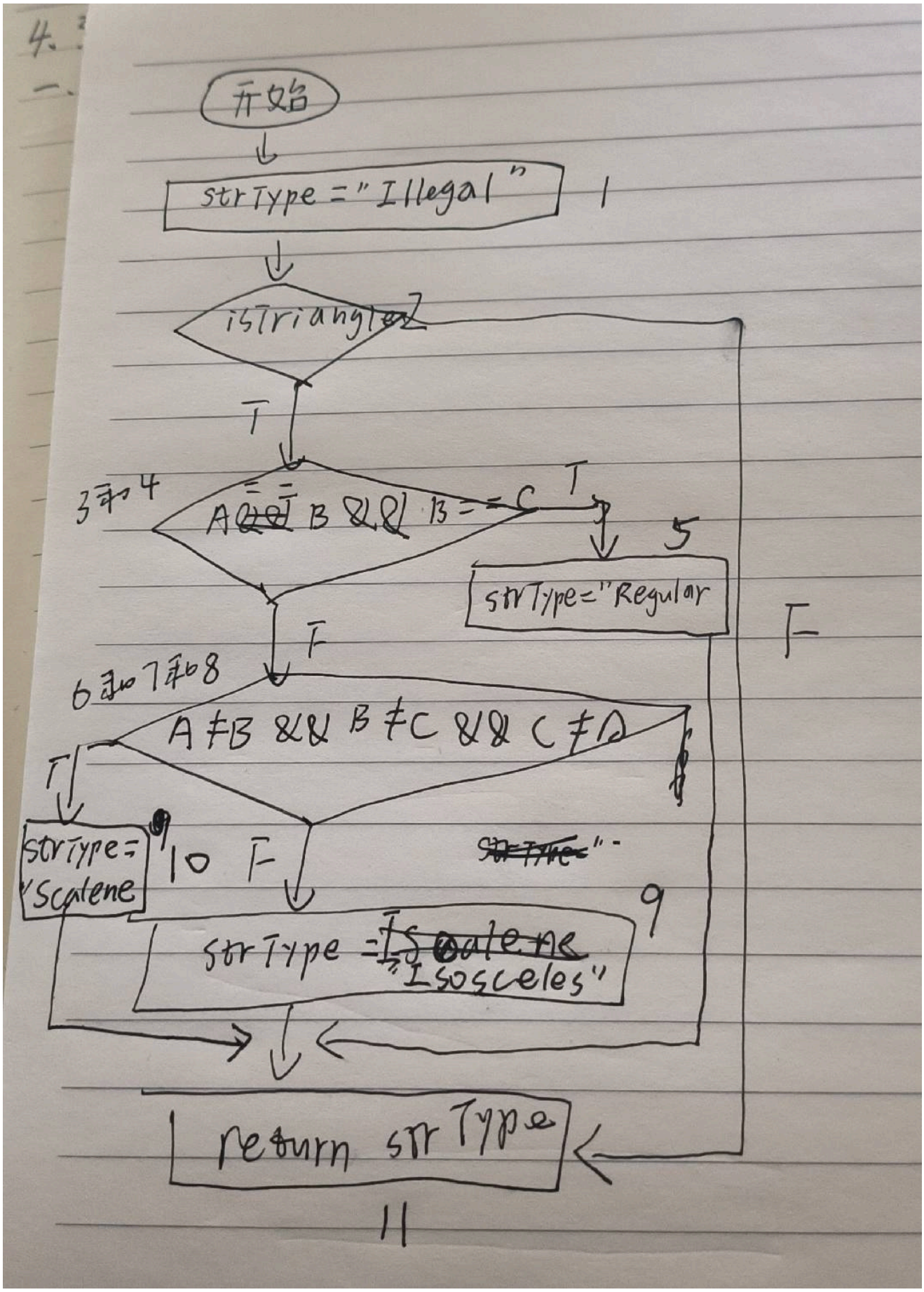
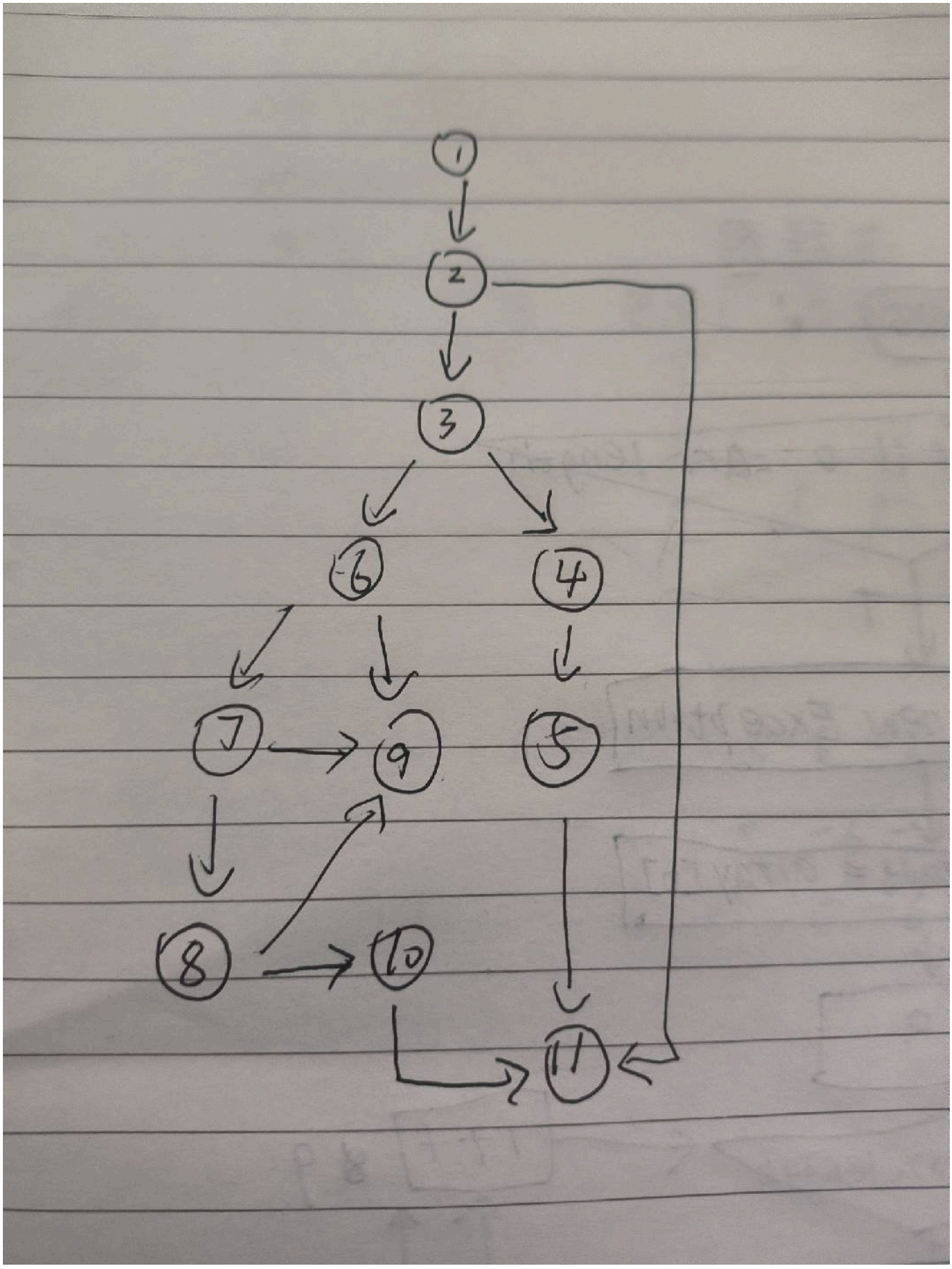
Triangle - getType()

流程图


开始
strType = "Illegal"
1
isTriangle
T
3 和 4
A == B && B == C
T
5
strType = "Regular
F
F
6 和 7 和 8
A ≠ B && B ≠ C && C ≠ A
T
strType = "Scalene
10
F
9
strType = "Isosceles"
return strType
11

控制流图



MCDC 覆盖

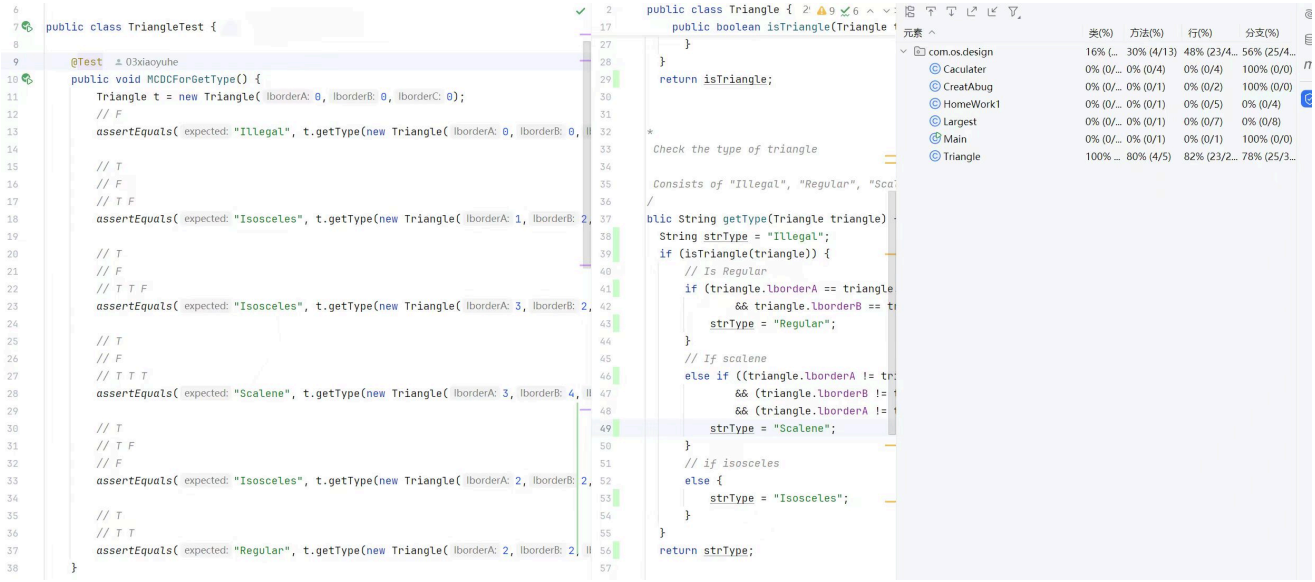
用例	输入	条件 1	组合条件 2	组合条件3	结果
1	{0,0,0}	F	- -	- - -	Illegal
2	{1,2,2}	T	F -	T F -	Isosceles
3	{2,2,3}	T	F -	T T F	Isosceles
4	{3,4,5}	T	F -	T T T	Scalene
5	{2,2,1}	T	T F	F - -	Isosceles
6	{2,2,2}	T	T T	- - -	Regular

基本路径及其测试用例

基本路径	是否可行	测试用例	结果
1 2 11	可行	1 2 3	Illegal
1 2 3 4 5 11	可行	2 2 2	Regular
1 2 3 6 7 9 11	可行	2 3 3	Isosceles
1 2 3 6 7 8 9 11	可行	2 3 2	Isosceles
1 2 3 6 7 8 10 11	可行	2 3 4	Scalene
1 2 3 4 6 9 11	可行	2 2 3	Isosceles
1 2 3 6 9 11	不可行		

编程截图

MCDC

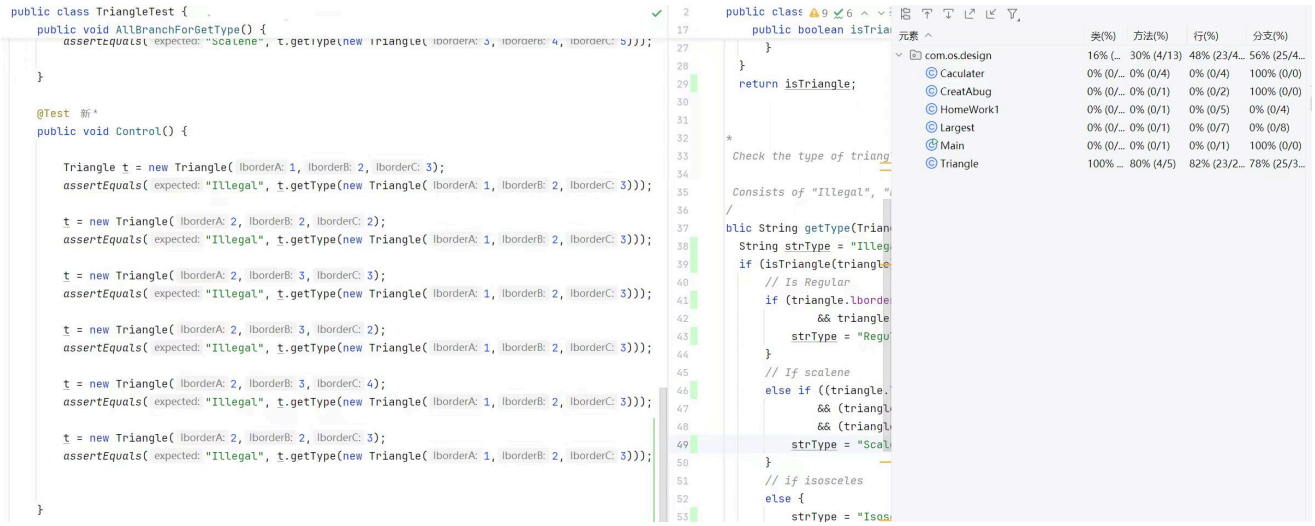


```
public class TriangleTest {

    @Test  03xiaoyuhe
    public void MCDCForGetType() {
        Triangle t = new Triangle( lborderA: 0, lborderB: 0, lborderC: 0);
        // F
        assertEquals( expected: "Illegal", t.getType(new Triangle( lborderA: 0, lborderB: 0,

        // T
        // F
        // T F
        assertEquals( expected: "Isosceles", t.getType(new Triangle( lborderA: 1, lborderB: 2,

        // T
        // F
        // T T F
        assertEquals( expected: "Isosceles", t.getType(new Triangle( lborderA: 3, lborderB: 2,

        // T
        // F
        // T T T
        assertEquals( expected: "Scalene", t.getType(new Triangle( lborderA: 3, lborderB: 4,

        // T
        // T F
        // F
        assertEquals( expected: "Isosceles", t.getType(new Triangle( lborderA: 2, lborderB: 2,

        // T
        // T T
        assertEquals( expected: "Regular", t.getType(new Triangle( lborderA: 2, lborderB: 2,
    }
```

```
public class Triangle {  9  6
    public boolean isTriangle(Triangle
        }
    }
    return isTriangle;


*
 Check the type of triangle

 Consists of "Illegal", "Regular", "Sca
/
blic String getType(Triangle triangle)
    String strType = "Illegal";
    if (isTriangle(triangle)) {
        // Is Regular
        if (triangle.lborderA == triangle
                && triangle.lborderB == t
            strType = "Regular";
        }
        // If scalene
        else if ((triangle.lborderA != tr
                && (triangle.lborderB !=
                && (triangle.lborderA !=
            strType = "Scalene";
        }
        // if isosceles
        else {
            strType = "Isosceles";
        }
    }
    return strType;
```

元素	类(%)	方法(%)	行(%)	分支(%)
com.os.design	16% (...	30% (4/13)	48% (23/4...	56% (25/4...
Caculater	0% (0/...	0% (0/4)	0% (0/4)	100% (0/0)
CreatAbug	0% (0/...	0% (0/1)	0% (0/2)	100% (0/0)
HomeWork1	0% (0/...	0% (0/1)	0% (0/5)	0% (0/4)
Largest	0% (0/...	0% (0/1)	0% (0/7)	0% (0/8)
Main	0% (0/...	0% (0/1)	0% (0/1)	100% (0/0)
Triangle	100% ...	80% (4/5)	82% (23/2...	78% (25/3...

基本路径测试



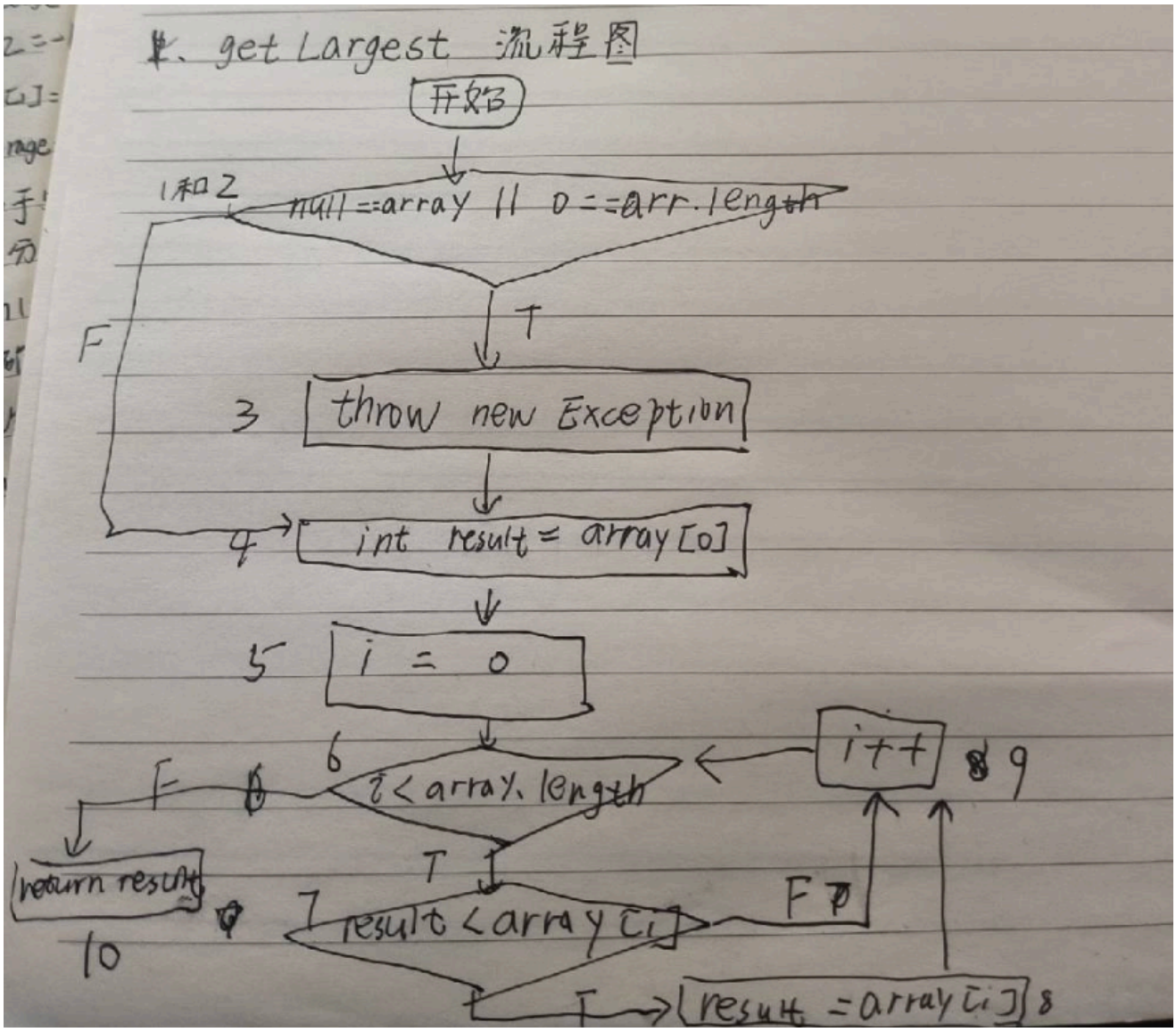
```
public class TriangleTest {
    public void AllBranchForGetType() {
        assertEquals( expected: "Scalene", t.getType(new Triangle( lborderA: 3, lborderB: 4, lborderC: 5)));

    }

    @Test  新*
    public void Control() {

        Triangle t = new Triangle( lborderA: 1, lborderB: 2, lborderC: 3);
        assertEquals( expected: "Illegal", t.getType(new Triangle( lborderA: 1, lborderB: 2, lborderC: 3)));

        t = new Triangle( lborderA: 2, lborderB: 2, lborderC: 2);
        assertEquals( expected: "Illegal", t.getType(new Triangle( lborderA: 1, lborderB: 2, lborderC: 3)));

        t = new Triangle( lborderA: 2, lborderB: 3, lborderC: 3);
        assertEquals( expected: "Illegal", t.getType(new Triangle( lborderA: 1, lborderB: 2, lborderC: 3)));

        t = new Triangle( lborderA: 2, lborderB: 3, lborderC: 2);
        assertEquals( expected: "Illegal", t.getType(new Triangle( lborderA: 1, lborderB: 2, lborderC: 3)));

        t = new Triangle( lborderA: 2, lborderB: 3, lborderC: 4);
        assertEquals( expected: "Illegal", t.getType(new Triangle( lborderA: 1, lborderB: 2, lborderC: 3)));

        t = new Triangle( lborderA: 2, lborderB: 2, lborderC: 3);
        assertEquals( expected: "Illegal", t.getType(new Triangle( lborderA: 1, lborderB: 2, lborderC: 3)));


    }
```

```
public class  9  6
    public boolean isTria
        }
    }
    return isTriangle;


*
 Check the type of triang

 Consists of "Illegal", "
/
blic String getType(Trian
    String strType = "Illeg
    if (isTriangle(triangle
        // Is Regular
        if (triangle.lborde
                && triangle
            strType = "Regu
        }
        // If scalene
        else if ((triangle.
                && (triangl
                && (triangl
            strType = "Scal
        }
        // if isosceles
        else {
            strType = "Isos
```

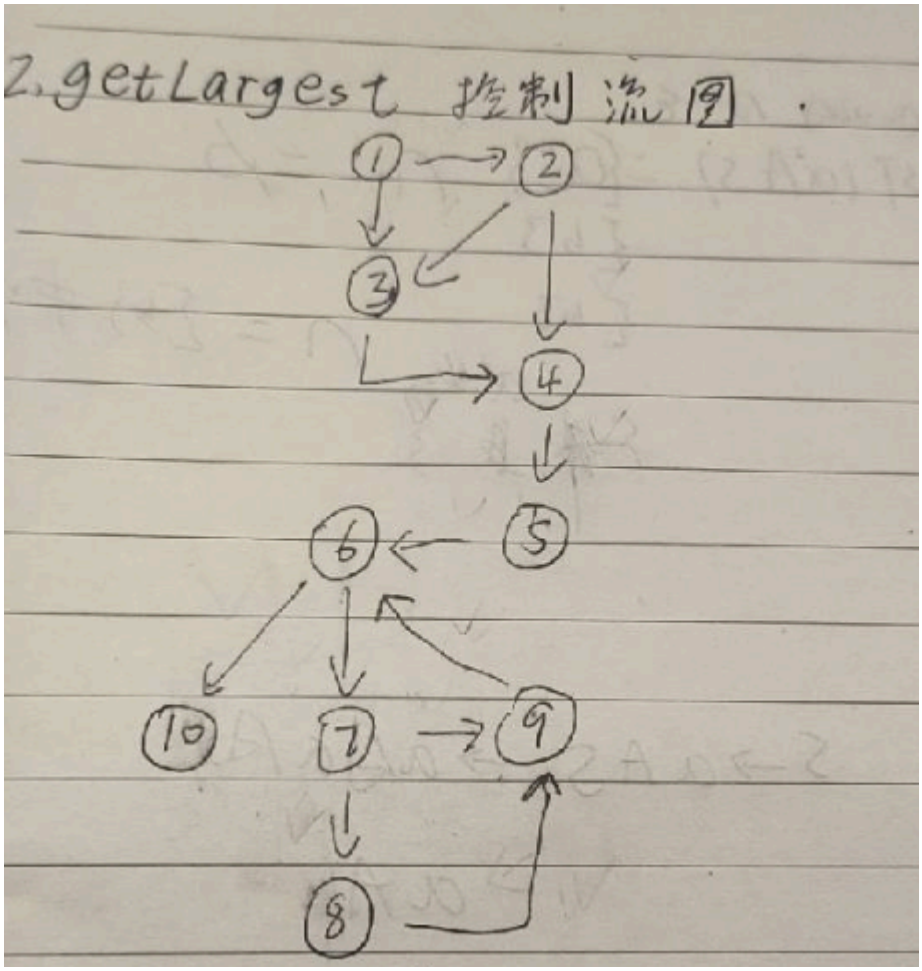
元素	类(%)	方法(%)	行(%)	分支(%)
com.os.design	16% (...	30% (4/13)	48% (23/4...	56% (25/4...
Caculater	0% (0/...	0% (0/4)	0% (0/4)	100% (0/0)
CreatAbug	0% (0/...	0% (0/1)	0% (0/2)	100% (0/0)
HomeWork1	0% (0/...	0% (0/1)	0% (0/5)	0% (0/4)
Largest	0% (0/...	0% (0/1)	0% (0/7)	0% (0/8)
Main	0% (0/...	0% (0/1)	0% (0/1)	100% (0/0)
Triangle	100% ...	80% (4/5)	82% (23/2...	78% (25/3...

getLargest()

流程图



控制流图



基本路径及其测试用例

基本路径

(1) **1 - 3 - 4 - 5 - 6 - 10**

(2) **1 - 2 - 4 - 5 - 6 - 10**

(3) **1 - 2 - 3 - 4 - 5 - 6 - 10**

(4) **1 - 2 - 4 - 5 - 6 - 7 - 9 - 6 - 10**

(5) **1 - 2 - 4 - 5 - 6 - 7 - 8 - 9 - 6 - 10**

基本路径	是否可行	测试用例
(1)	可行	array = null
(2)	不可行	无
(3)	可行	array = []
(4)	可行	array = [2, 1]
(5)	可行	array = [1, 2]

MCDC 覆盖

用例	输入	条件 1	条件 2	条件 3	条件 4	结果
1	null	T	-	F	-	异常
2	[]	F	T	F	-	异常
3	[1,2]	F	F	T/F	T/F	2

基本路径

(1) **1 - 3 - 4 - 5 - 6 - 10**

(2) **1 - 2 - 4 - 5 - 6 - 10**

(3) **1 - 2 - 3 - 4 - 5 - 6 - 10**

(4) **1 - 2 - 4 - 5 - 6 - 7 - 9 - 6 - 10**

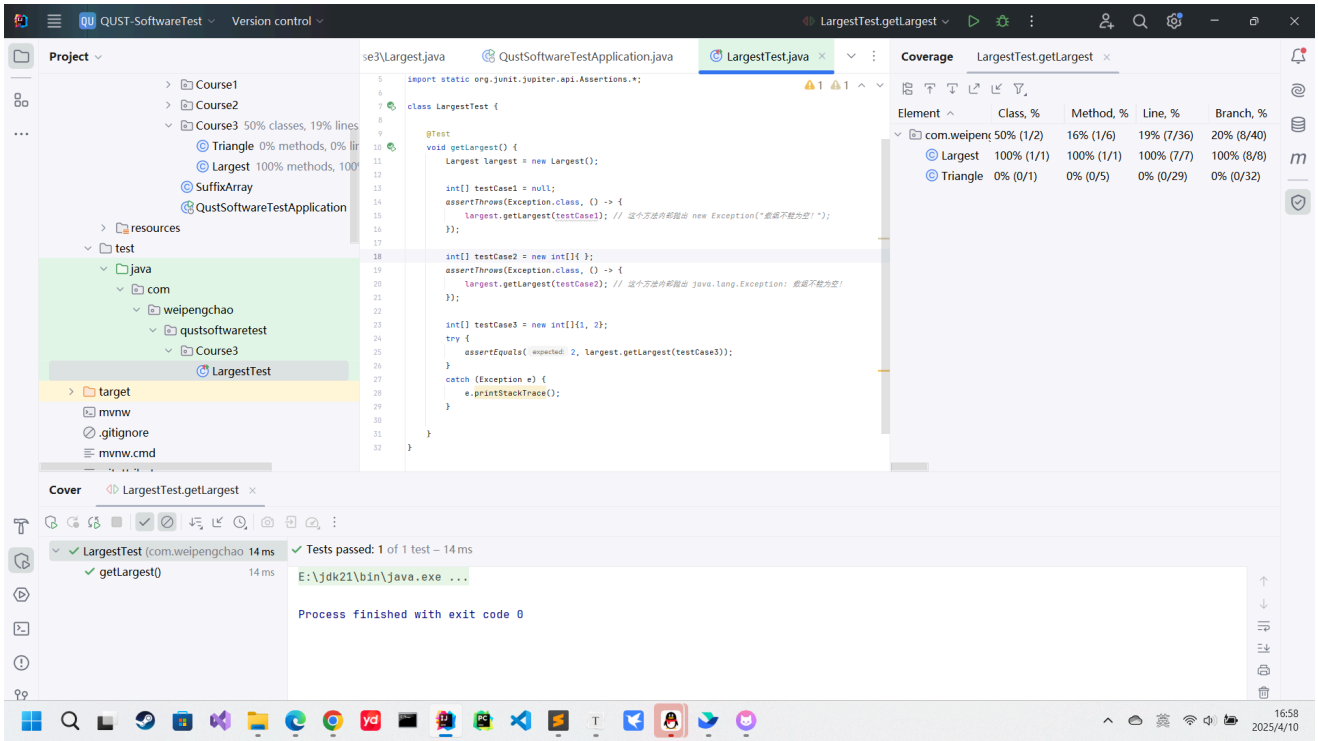
(5) **1 - 2 - 4 - 5 - 6 - 7 - 8 - 9 - 6 - 10**

基本路径测试用例

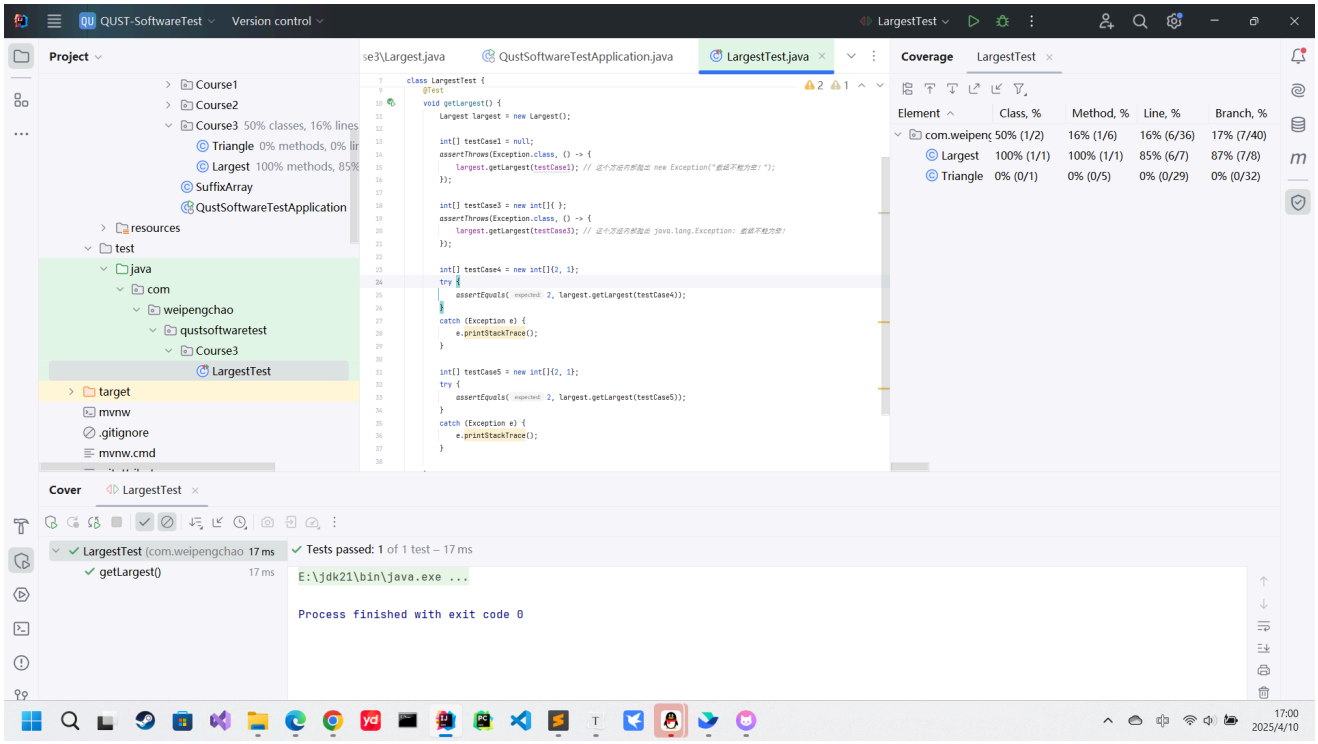
基本路径	是否可行	测试用例
(1)	可行	array = null
(2)	不可行	无
(3)	可行	array = []
(4)	可行	array = [2, 1]
(5)	可行	array = [1, 2]

编程截图

MCDC



基本路径测试


QUST-SoftwareTest
Version control
LargestTest
Project
Course1
Course2
Course3 50% classes, 16% lines
Triangle 0% methods, 0% lin
Largest 100% methods, 85%
SuffixArray
QustSoftwareTestApplication
resources
test
java
com
weipengchao
qustsoftwaretest
Course3
LargestTest
target
mvnw
.gitignore
mvnw.cmd
se3\Largest.java
QustSoftwareTestApplication.java
LargestTest.java
class LargestTest {
@Test
void getLargest() {
Largest largest = new Largest();
int[] testCase1 = null;
assertThrows(Exception.class, () -> {
largest.getLargest(testCase1); // 这个方法将抛出 new Exception("数组不能为空！");
});
int[] testCase3 = new int[]{ };
assertThrows(Exception.class, () -> {
largest.getLargest(testCase3); // 这个方法将抛出 java.lang.Exception: 数组不能为空!
});
int[] testCase4 = new int[]{2, 1};
try {
assertEquals(expected: 2, largest.getLargest(testCase4));
}
catch (Exception e) {
e.printStackTrace();
}
int[] testCase5 = new int[]{2, 1};
try {
assertEquals(expected: 2, largest.getLargest(testCase5));
}
catch (Exception e) {
e.printStackTrace();
}
Coverage
LargestTest
Element
Class, %
Method, %
Line, %
Branch, %
com.weipeng 50% (1/2) 16% (1/6) 16% (6/36) 17% (7/40)
Largest 100% (1/1) 100% (1/1) 85% (6/7) 87% (7/8)
Triangle 0% (0/1) 0% (0/5) 0% (0/29) 0% (0/32)
Cover
LargestTest
LargestTest (com.weipengchao 17 ms
getLargest() 17 ms
Tests passed: 1 of 1 test – 17 ms
E:\jdk21\bin\java.exe ...
Process finished with exit code 0
17:00
2025/4/10