

湖南科技大学计算机科学与工程学院

软件测试实验报告

专业班级： 计算机科学与技术三班

姓 名： 郭怀

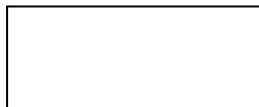
学 号： 1705010303

指导教师： 何庭钦

时 间： 2020-5-19

地 点：

指导教师评语：



签名：

年 月 日

实验名称	逻辑覆盖测试技术实验		
实验性质 (必修、选修)		实验类型（验证、 设计、创新、综合）	
实验课时		实验日期	2020-5-19
实验仪器设备以及实验 软硬件要求	硬件环境：笔记本电脑一台 软件环境：Java SDK Eclipse Junit4 eclemma 程序流程图绘制：ProcessOn		
实验目的	(1) 巩固基于控制流白盒测试知识，对于给定的待测程序，能熟练应用基本控制流覆盖方法设计测试用例； (2) 通过绘制程序控制流程图，实现对程序源代码的逻辑描述； (3) 掌握逻辑短路对测试的影响； (4) 培养严谨和系统的测试精神，学习测试用例的设计和分析。		
实验内容（实验原理、运用的理论知识、算法、程序、步骤和方法）			
由于内容过多，所以实验内容在此表格后面单独列出。			
实验结果与分析			
在这次实验中，实际用各种覆盖方法来设计测试用例，在做实验的过程中纠正了我的一些错误认识，让我学到了很多。同时学会了一些新工具的使用，用这些工具可以提高我测试的效率并清晰直观地把测试效果呈现出来。			

实验三：逻辑覆盖测试技术实验

一. 代码和程序流程图

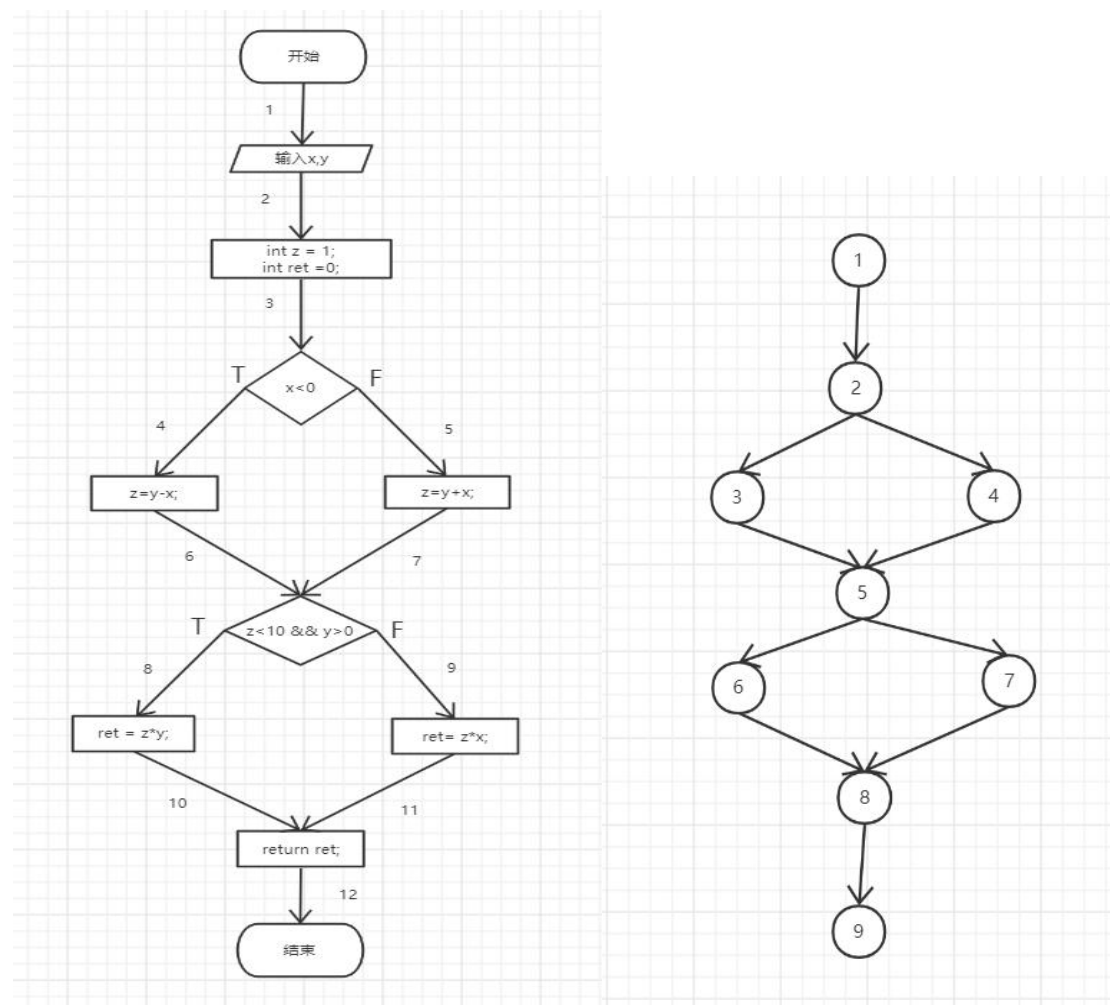
代码：

```
package test;

public class GetNum {

    public static int getNmb(int x, int y) {
        int z = 1;
        int ret = 0;
        if (x < 0) {
            z = y - x;
        } else {
            z = y + x;
        }
        if (z < 10 && y > 0) {
            ret = z * y;
        } else {
            ret = z * x;
        }
        return ret;
    }
}
```

对应的程序流程图和简化的流图



二. 测试用例设计

1.语句覆盖

编号	输入	期待的输出	执行路径
1	x=1 y=12	13	1->2->3->5->7->9->11->12
2	x= -1 y=2	6	1->2->3->4->6->8->10->12

2.判定覆盖

编号	输入	判定一 (x<0)	判定二 (z<10&& y>0)	期待的输出	执行路径
1	x=1 y=12	F	F	13	1->2->3->5->7->9->11->12
2	x= -1 y=2	T	T	6	1->2->3->4->6->8->10->12

3.条件覆盖

编号	输入	条件一 (x<0)	条件二 (z<10)	条件三 (y>0)	期待的输出	执行路径
1	x=1 y=12	F	F	T	13	1->2->3->5->7->9->11->12
2	x= -1 y=-2	T	T	F	1	1->2->3->4->6->9->11->12

4.判定条件覆盖

编号	输入	判定一 (x<0)	判定二 (z<10&& y>0)	条件一 (x<0)	条件二 (z<10)	条件三 (y>0)	期待的输出	执行路径
1	x=1 y=12	F	F	F	F	T	13	1->2->3->5->7->9->11->12
2	x= -1 y= -2	T	F	T	T	F	1	1->2->3->4->6->9->11->12
3	x= -1 y=2	T	T	T	T	T	6	1->2->3->4->6->8->10->12

5.条件组合覆盖

第一个判定中只有一个条件 $x < 0$ ，第二个判定中有两个条件 $z < 10$ 和 $y > 0$ 。这三个条件在两个判定中总共有 6 个组合：

第一个判定中：(1) $x < 0$ (2) $x \geq 0$

第二个判定中：(3) $z < 10, y > 0$ (4) $z < 10, y \leq 0$ (5) $z \geq 10, y > 0$ (6) $z \geq 10, y \leq 0$

下面 4 个测试用例覆盖了这 6 个组合

编号	输入	条件一 ($x < 0$)	条件二 ($z < 10$)	条件三 ($y > 0$)	覆盖的条件组合	期待的输出	执行路径
1	$x=1 \ y=2$	F	T	T	(2)、(3)	6	1->2->3->5->7->8->10->12
2	$x=3 \ y=-1$	F	T	F	(2)、(4)	6	1->2->3->5->7->9->11->12
3	$x=-6 \ y=6$	T	F	T	(1)、(5)	-72	1->2->3->4->6->9->11->12
4	$x=15 \ y=-2$	F	F	F	(2)、(6)	195	1->2->3->5->7->9->11->12

三. 测试

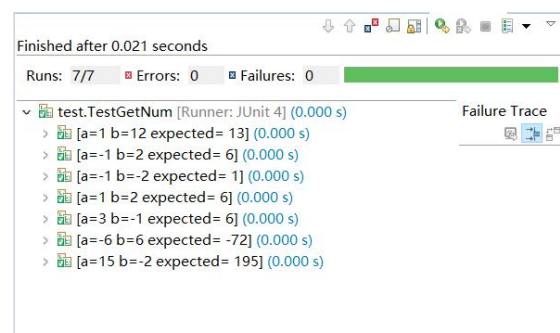
将上面五种覆盖方法得到的测试用例综合起来获得如下的测试用例表：

编号	输入	来源	期待的输出	实际输出
1	$x=1 \ y=12$	语句覆盖 判定覆盖 条件覆盖 判定条件覆盖	13	13
2	$x=-1 \ y=2$	语句覆盖 判定覆盖 判定条件覆盖	6	6
3	$x=-1 \ y=-2$	条件覆盖 判定条件覆盖	1	1
4	$x=1 \ y=2$	条件组合覆盖	6	6
5	$x=3 \ y=-1$	条件组合覆盖	6	6
6	$x=-6 \ y=6$	条件组合覆盖	-72	-72
7	$x=15 \ y=-2$	条件组合覆盖	195	195

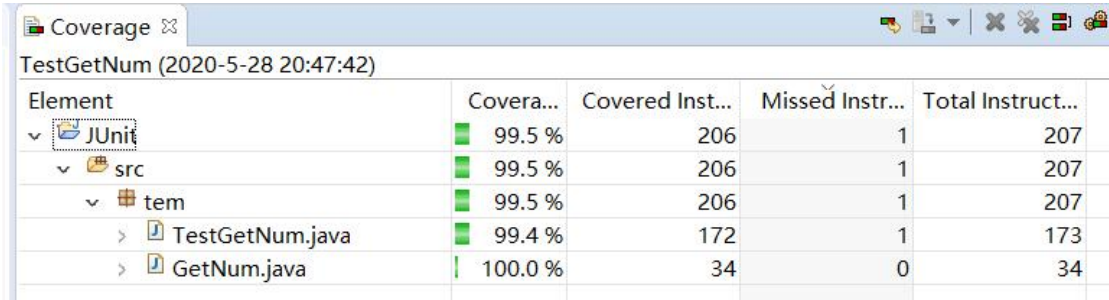
被测试类（GetNum.java）和测试类（TestGetNum.java）

```
1 package test;
2
3 public class GetNum {
4
5     public static int getNmb(int x, int y) {
6         int z = 1;
7         int ret = 0;
8         if (x < 0) {
9             z = y - x;
10        } else {
11            z = y + x;
12        }
13        if (z < 10 && y > 0) {
14            ret = z * y;
15        } else {
16            ret = z * x;
17        }
18        return ret;
19    }
20 }
21
22
23 @RunWith(Parameterized.class)
24 public class TestGetNum {
25
26     int a;
27     int b;
28     int expected;
29
30     public TestGetNum(int a, int b, int expected) {
31         super();
32         this.a = a;
33         this.b = b;
34         this.expected = expected;
35     }
36
37     @Parameters(name = "a={0} b={1} expected= {2}")
38     public static List datas() {
39         return (List) Arrays.asList(
40             new Object[][]{
41
42                 {1,12,13},
43                 {-1,2,6},
44                 {-1,-2,1},
45                 {1,2,6},
46                 {3,-1,6},
47                 {-6,6,-72},
48                 {15,-2,195},
49             }
50         );
51     }
52
53     @Test
54     public void test() {
55         GetNum getNum=new GetNum();
56         assertTrue(expected==getNum.getNmb(a, b));
57     }
58 }
59
60 }
```

使用 junit 进行测试:



用 eclemma 测试覆盖率:



The screenshot shows the Coverage tool window for a test named 'TestGetNum' run on 2020-5-28 at 20:47:42. The table displays coverage data for the JUnit test suite, including the source directory 'src' and the temporary directory 'tem'. Two Java files are listed: 'TestGetNum.java' and 'GetNum.java'. The coverage for 'TestGetNum.java' is 99.4% (172 covered, 1 missed instruction), and for 'GetNum.java' it is 100.0% (34 covered, 0 missed instructions).

Element	Covera...	Covered Inst...	Missed Instr...	Total Instruct...
JUnit	99.5 %	206	1	207
src	99.5 %	206	1	207
tem	99.5 %	206	1	207
TestGetNum.java	99.4 %	172	1	173
GetNum.java	100.0 %	34	0	34

覆盖率报告见附件 (html 格式)