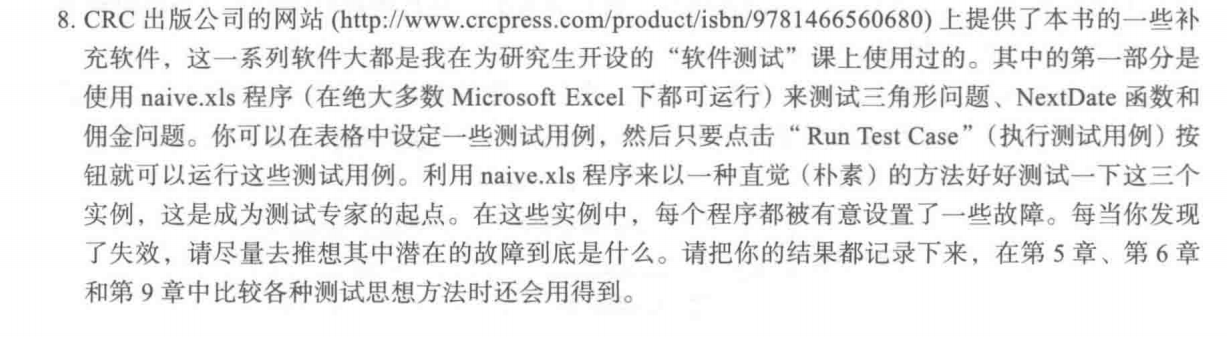
# 作业二

**要求**：

1. 使用Java语言实现以下三个程序；
2. 使用测试用例验证程序的正确性（JUnit）；
3. 测试用例数据可以参考naiveX.xls，并运行naiveX.xls;(p24页第8题）；

（在Excel中可以直接运行naïveX.xls文件，会自动生成相应结果，是Excel中VBA写的，这个要求并不是要我去写类似的Excel文件！！！让java运行结果与Excel对比即可吧）

1. 使用git管理代码（gitee或GitHub均可）；



**目的**：

1. 练习基本的编程能力；
2. 学习JUnit自动测试框架；
3. 学习Eclemma代码覆盖率分析；
4. 理解以下三个题目，这是后面讲解黑盒测试方法和白盒测试方法的基础案例；
5. **三角形题目（经典软件测试案例）**：

**输入**3个整数a、b和c分别作为三角形的三条边，要求a、b和c必须满足以下条件：

1、整数

2、3个数

3、边长大于等于1，小于等于100

4、任意两边之和大于第三边

**输出**为5种情况之一：

如果不满足条件1、2、3，则程序输出为“输入错误”。

如果不满足条件4，则程序输出为“非三角形”。

如果三条边相等，则程序输出为“等边三角形”。

如果恰好有两条边相等，则程序输出为“等腰三角形”。

如果三条边都不相等，则程序输出为“一般三角形”。

1. **NextDate函数问题**：NextDate函数输入为month(月份)、day(日期)和year(年)，输出为输入后一天的日期。例如，如果输入为：1964年8月16日，则输出为1964年8月17日。要求输入变量month、day和year都是整数值，并且满足以下条件：

Con1. 1≤month≤12

Con2. 1≤day≤31

Con3. 1900≤year≤2050

1. **佣金问题**

问题描述

从前有一位销售人员在亚利桑那州代销密苏里军械制造厂生产的步枪配件，包括枪机（lock）、枪托（stock）和枪管（barrel）。枪机售价45美元，枪托售价30美元，枪管售价25美元。销售人员每个月至少要卖出一个枪机，一个枪托和一个枪管（但是没有必要是一支完整的步枪），而制造厂的生产能力限制销售人员一个月最多只能卖出70个枪机、80个枪托和90个枪管。每走访过一个城镇之后，销售人员都要给密苏里军械厂发一封电报，汇报在这一城镇中销售枪机、枪托和枪管的数量。销售人员月末会再发一封很短的电报，通知“-1个枪机售出”。这样军械厂就知道当月的销售活动已经结束了，计算销售人员应得的佣金了。佣金计算方法如下：销售总额1000美元以下（含1000美元）部分的佣金为10%，1000至1800美元之间部分的佣金为15%，超过1800美元的部分的佣金为20%。

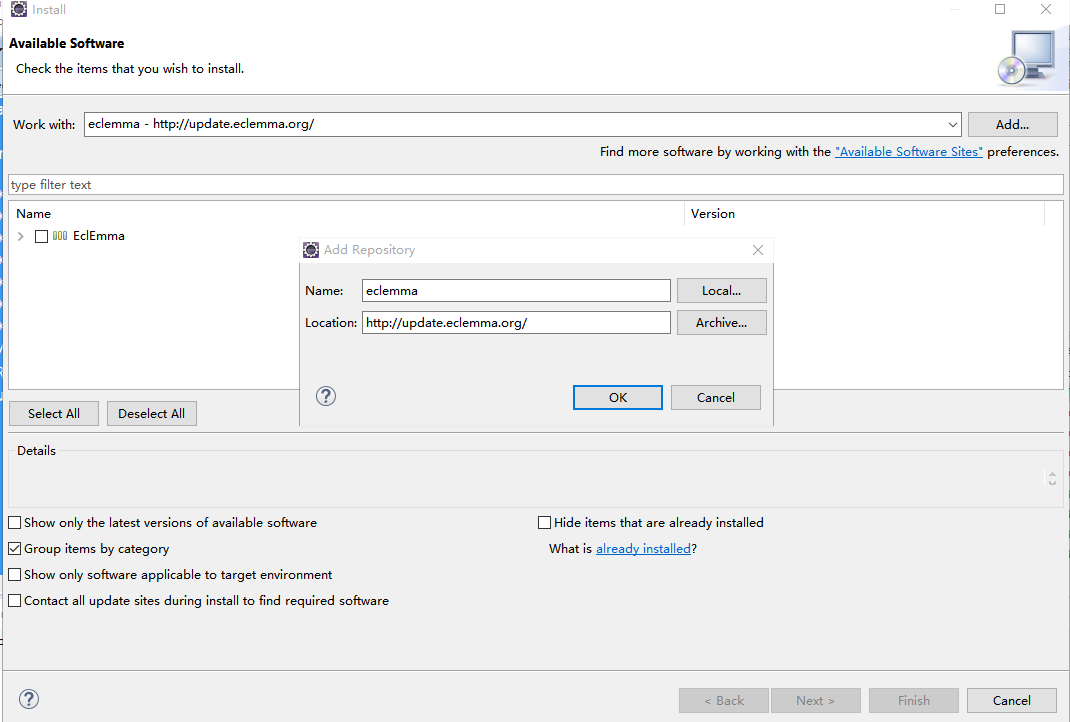
讨论：

这个佣金程序可分为三个部分：输入数据处理部分，验证输入数据的有效性，销售额统计计算部分，以及佣金计算部分。此处可省略了对输入数据有效性的验证，可使用条件循环语句While来模拟对电报的处理。

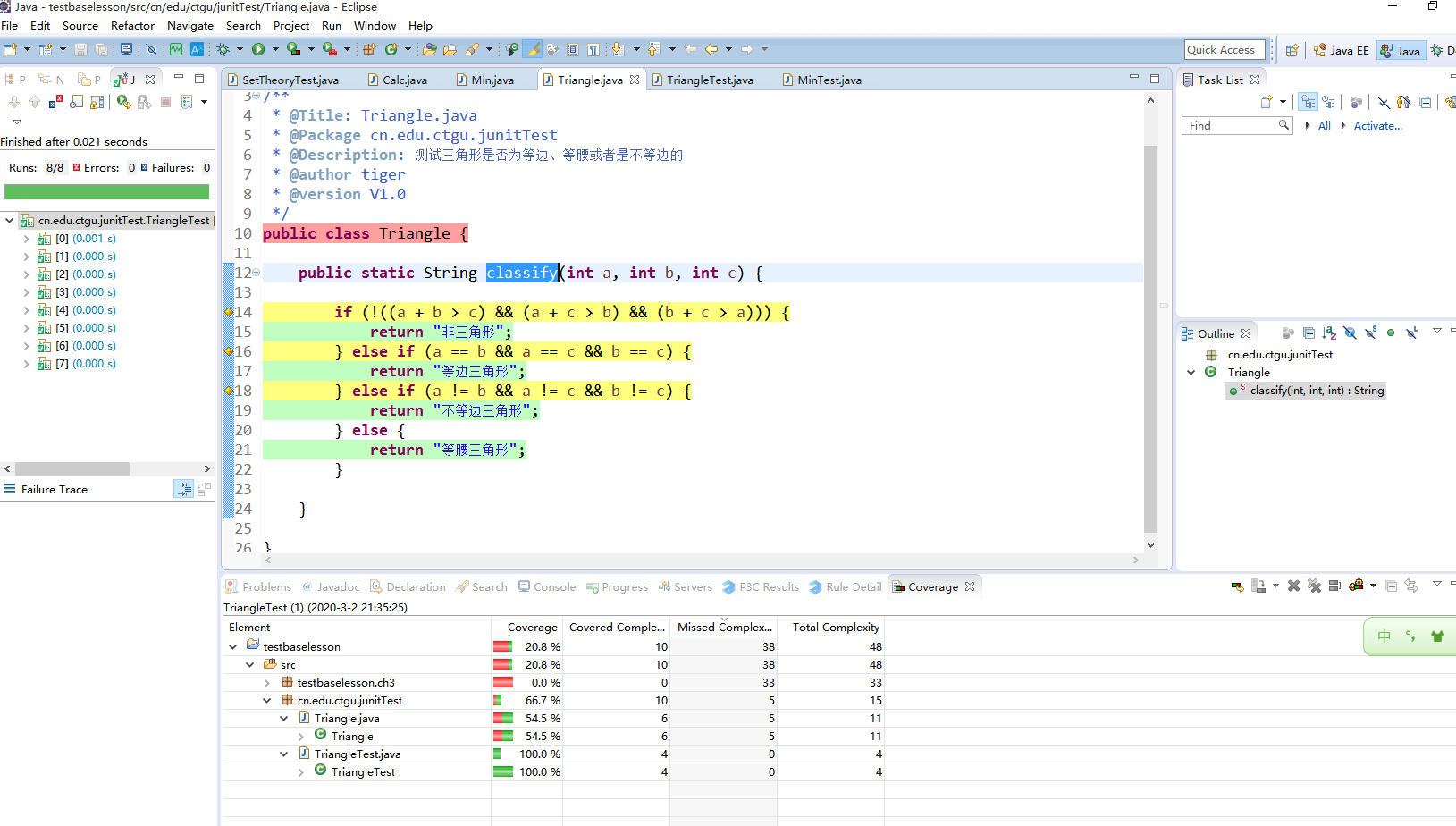
实验步骤参考题目1示例：

**1. Eclemma的安装和使用**

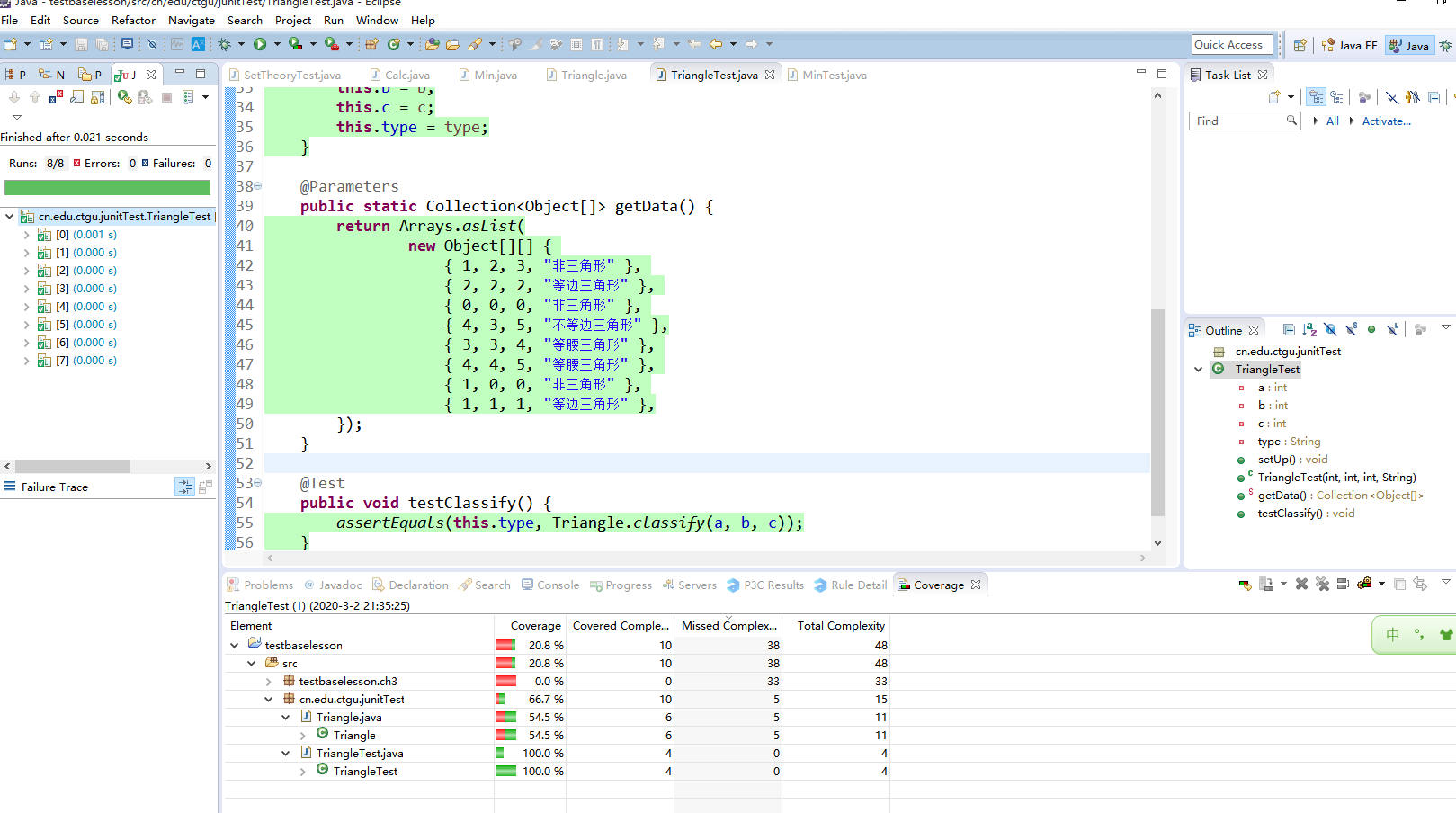
  在线安装：http://update.eclemma.org/



**2. Eclemma测试结果**



Junit测试结果：



用例全部测试通过。

**4.使用代码**

 三角形类：

**package** cn.edu.ctgu.junitTest;

/\*\*

\* **@Title**: Triangle.java

\* **@Package** cn.edu.ctgu.junitTest

\* **@Description**: 测试三角形是否为等边、等腰或者是不等边的

\* **@author** tiger

\* **@version** V1.0

\*/

**public** **class** Triangle {

**public** **static** String classify(**int** a, **int** b, **int** c) {

**if** (!((a + b > c) && (a + c > b) && (b + c > a))) {

**return** "非三角形";

} **else** **if** (a == b && a == c && b == c) {

**return** "等边三角形";

} **else** **if** (a != b && a != c && b != c) {

**return** "不等边三角形";

} **else** {

**return** "等腰三角形";

}

}

}

测试类：

package cn.edu.ctgu.junitTest;

import static org.junit.Assert.\*;

import java.util.Arrays;

import java.util.Collection;

import org.junit.Before;

import org.junit.Test;

import org.junit.runner.RunWith;

import org.junit.runners.Parameterized;

import org.junit.runners.Parameterized.Parameters;

/\*\*

\* @Title: TriangleTest.java

\* @Package cn.edu.ctgu.junitTest

\* @Description: 三角形测试类

\* @author tiger

\* @version V1.0

\*/

// 参数化运行器

@RunWith(Parameterized.class)

public class TriangleTest {

private int a, b, c;

private String type;

@Before

public void setUp() throws Exception {

}

public TriangleTest(int a, int b, int c, String type) {

this.a = a;

this.b = b;

this.c = c;

this.type = type;

}

@Parameters

public static Collection<Object[]> getData() {

return Arrays.asList(

new Object[][] {

{ 10, 20, 30, "非三角形" },

{ 2, 2, 2, "等边三角形" },

{ 0, 0, 0, "非三角形" },

{ 4, 3, 5, "不等边三角形" },

{ 3, 3, 4, "等腰三角形" },

{ 4, 4, 5, "等腰三角形" },

{ 1, 0, 0, "非三角形" },

{ 1, 1, 1, "等边三角形" },

});

}

@Test

public void testClassify() {

assertEquals(this.type, Triangle.classify(a, b, c));

}

}