****

**软件测试技术第一次实验报告**



**学 院 智能与计算学部**

**专 业 软件工程**

**年 级 2016**

**姓 名 王志航**

**2019年 3 月 12 日**

# 软件测试技术第一次实验报告

1. 需求分析（描述具体需求）

1.在Eclipse安装JUnit（4.12）和 hamcrest（1.3）

2.在Eclipse安装Eclemma

3.编写JAVA程序，用JUnit对程序进行测试。问题描述：口袋里有一张50元，一张20元，两张5元的钞票和三个1元的硬币。写一个程序，看看你能不能拿出一个给定的数字（X）元。

二．概要设计（简单描述设计思路，配合UML图）

Junit和hancrest安装：

添加外部的jar，首先自己去网上下载相应的包（可以从www.junit.org上下载，也可以从CSDN上搜索）；然后在新建的项目上右键——》构建路径——》添加外部文档——》找到你下载的jar包即可

程序设计：

1．将所有的硬币从大到小排列

2. 检查输入是否符合要求，是否为0到83之间的整数。

3. 用背包问题的思路解决，将硬币从大到小用循环一次判断，要分割的钱是否大于手头有的最大的一枚。

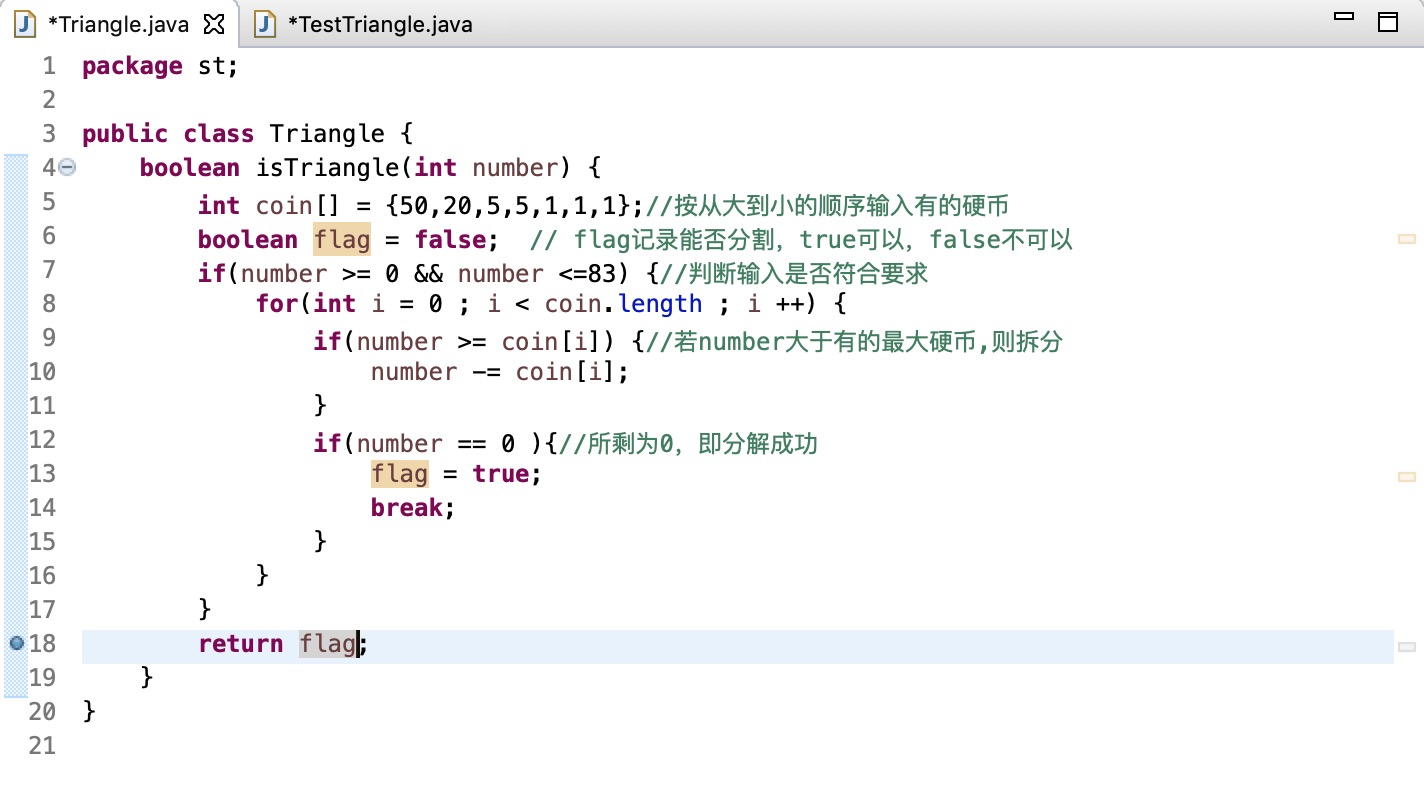
4.判断要分割的减去最大的是否为0（即分解成功），若是则返回true，否则返回false

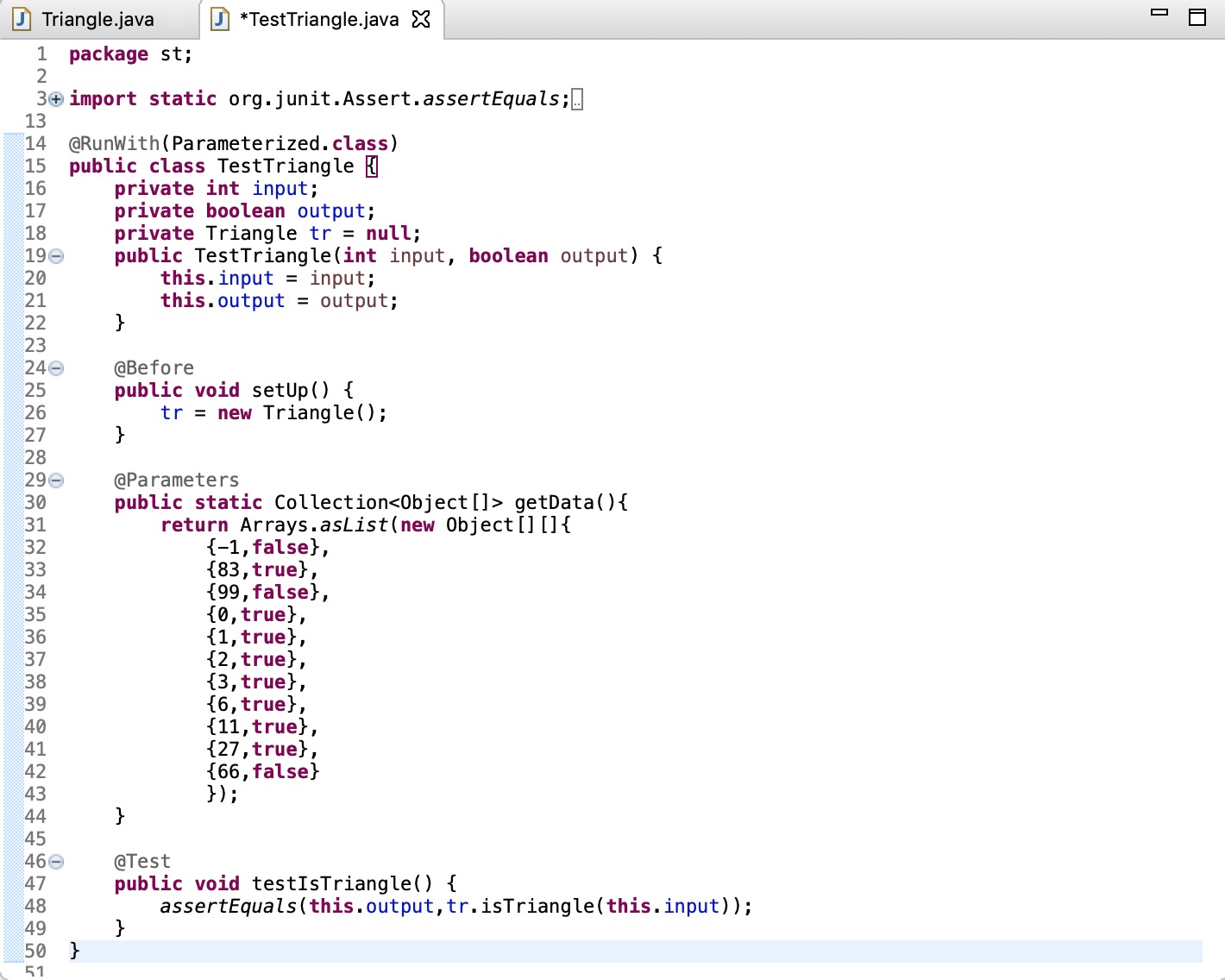
测试程序：

1.输入测试用例，关注循环和边界条件；

2.输入期望输出结果，得到输出。

1. 详细设计（详细描述具体如何实现，附代码及说明）





1. 调试分析（在实验过程中遇到的问题以及如何解决）

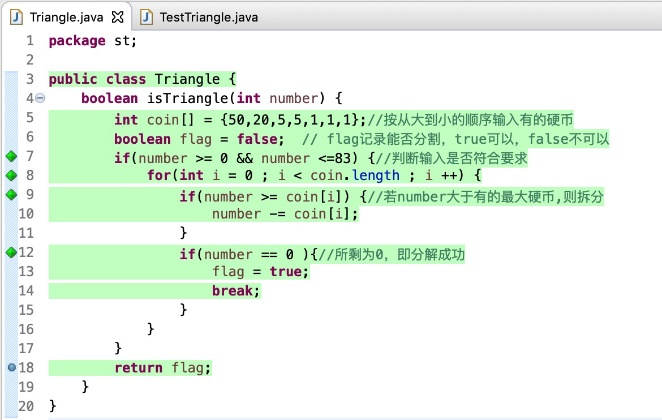
1.首先遇到的就是覆盖不全的问题，通过调整测试用例，变成了部分覆盖。在考虑for循环处的运行方式，增加测试用例，将其调整为100%完全覆盖。

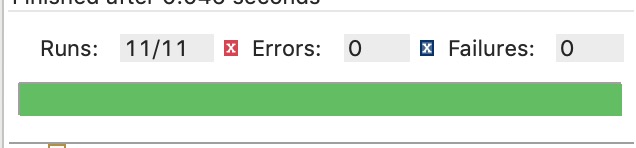
2.用数组来保存所有的钱，避免了复杂的循环，具有可扩展性。

1. 测试结果（描述输入和输出）

输入为一个所要分割的数字，将其限定为0到83之间，若能分割输出true，否则输出false。

结果如下图：





C:\Users\w\Documents\Tencent Files\78910421\FileRecv\MobileFile\C288F5C2-54A0-4775-B0FD-B1F57820BD3E.png

1. 总结

在实验中学会了运用junit写测试用例，并且学会了运用hamcrest测试覆盖率，将最近几节课所学的知识加以实践的运用，对测试有了新的认识。

url：

https://github.com/YourBrotherTT/softwareTesting.git