****

**软件测试技术第一次实验报告**



**学 院 软件学院**

**专 业 软件工程**

**年 级 2016**

**姓 名 刘岳森**

**2019年 3 月 12 日**

# 软件测试技术第一次实验报告

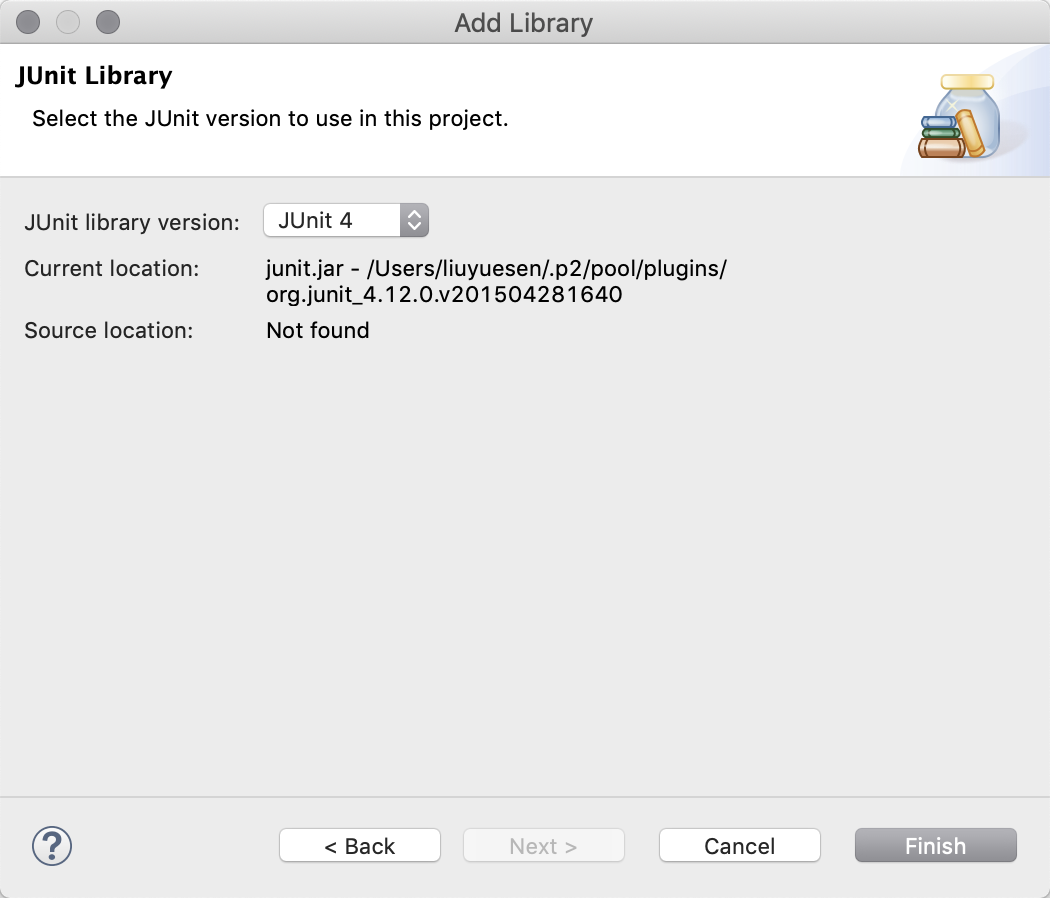
1. 需求分析（描述具体需求）

1.在eclipse中安装Junit Hamcrest

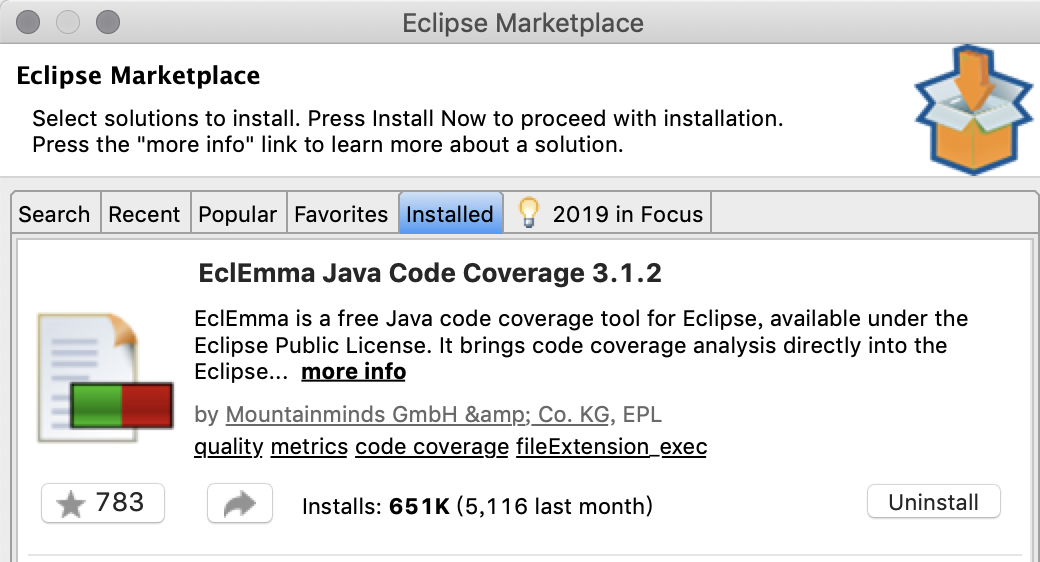
2.在eclipse中安装Eclemma

3.写一个解决triangle问题的程序，并且用Junit进行单元测试

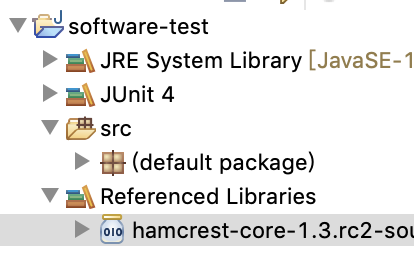
1. 概要设计（简单描述设计思路，配合UML图）
2. 安装Junit：



我们可以看到如图的使用界面，说明我们的Junit已经安装好了



在这里我们可以看到Eclemma我们已经安装好了



在这里我们也已经把hamcrest下载包安装好了

1. 详细设计（详细描述具体如何实现，附代码及说明）

首先编写解决问题的代码，代码如下

**import** java.io.IOException;

**import** junit.framework.TestCase;

**public** **class** test1 **extends** TestCase{

**public** **static** **boolean** main(**int** number) **throws** IOException {

// **TODO** Auto-generated method stub

**int**[] list = {50,20,5,5,1,1,1};

**boolean** answer = **false**;

**for** (**int** i =0;i<7;i++) {

**if** (number>=list[i]) {

number-=list[i];

}

**if** (number == 0) {

answer = **true**;

**break**;

}

}

**if** (answer == **true**) {

**return** **true**;

}

**else** {

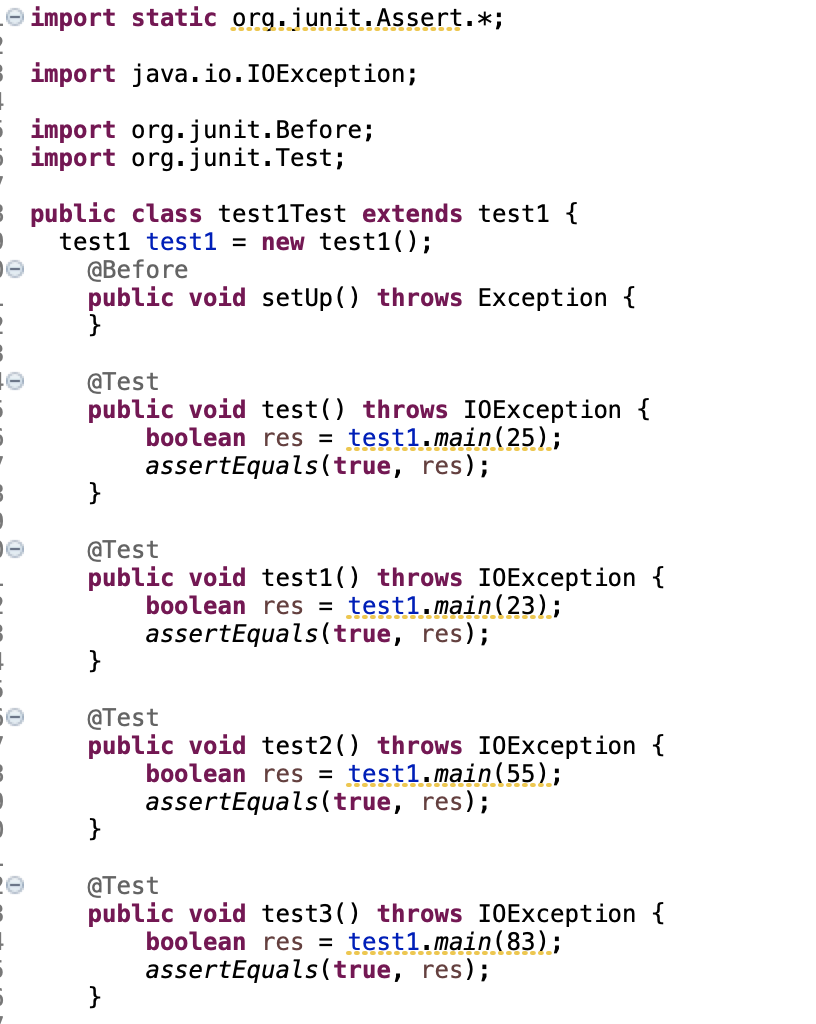
**return** **false**;

}

}

}

之后通过junit自动生成测试代码，并在测试代码里添加测试样例

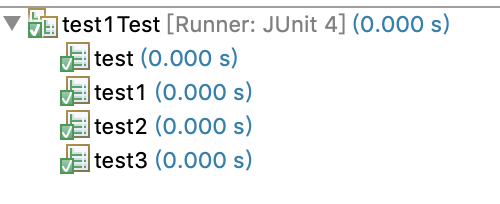


1. 调试分析（在实验过程中遇到的问题以及如何解决）

测试结果如图

在一开始由于对单元测试的步骤不太清楚，一开始没有跑通，后来就ok了

1. 测试结果（描述输入和输出）



1. 总结

在一开始安装了Hamcrest和Eclemma这两个插件，但是感觉并没有在后面的测试过程中用到，还是不太清楚他们各自的作用以及他们和junit之间的关系。