**《软件测试与质量保证》**

**实验指导书**

**天津城建大学**

**计算机与信息工程学院**

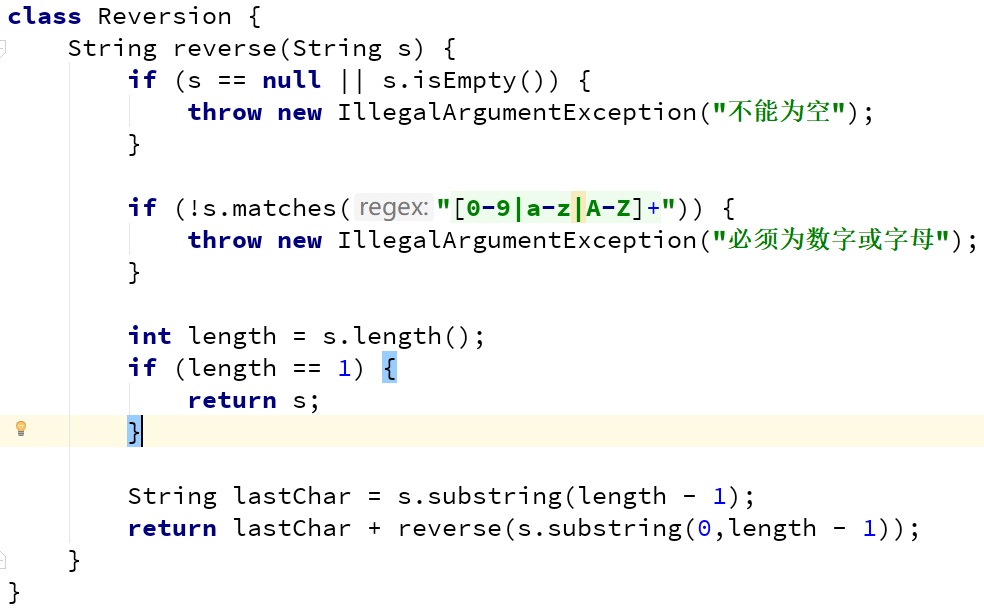
# 使用单元测试框架

## 实验目的

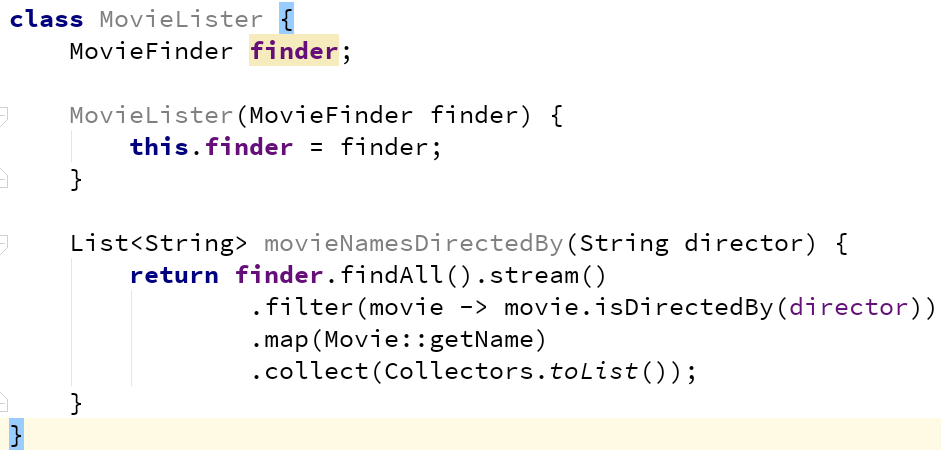
* 1. 掌握JUnit和Mockito框架的使用
  2. 掌握单元测试的概念

## 功能描述

1. 函数String reverse(String s)用于翻转字符串s。比如，当s=“hello world”时，reverse的返回值为“dlrow olleh”.



1. 接口MovieFinder的findAll方法用于查找所有电影。MovieLister依赖MovieFinder接口，其movieNamesDirectedBy方法用于返回指定导演的所有电影名字。



## 实验内容

1. **实验任务1：**使用IntelliJ IDEA创建名为junit-test的gradle工程并在build.gradle文件中添加junit5依赖。在主目录的包testing.junit下创建名为Reversion的类，它包含String reverse(String s)的方法用于翻转字符串。在测试目录的testing.junit包下创建名为ReversionTest的类用于测试reverse方法。测试用例包括：1）s为空或null，预期结果为抛出IllegalArgumentException。2）s不是由数字或字母组成，预期结果为抛出IllegalArgumentException。3）s只由一个字符组成，返回该字符自身。4）如果s的长度大于1且只由数字或字母组成，预期结果为翻转后的字符串
2. **实验任务2：**使用IntelliJ IDEA创建名为mockito-test的gradle工程并在build.gradle文件中添加junit5和mockito框架的依赖。在主目录包testng.mockito中完成MovieLister，Movie，MovieFinder相关类的实现。在测试目录中的包testing.mockito中创建类MovieListerTest用于对MovieLister进行单元测试。具体要求：1）使用Mockito模拟一个MovieFinder对象finder，然后使用它构造MovieLister对象lister。2）使用when方法为finder对象的findAll方法指定返回结果，编写相关断言验证list.movieNamesDirectedBy返回结果正确。

## 实验要求

1. 至少完成一个实验任务
2. 将每个工程打包为同名的jar包
3. 严格按实验报告模板中的格式与内容要求编写实验报告
4. 将所有jar包连同实验报告打包为zip文件，文件名为“你的学号\_unittest.zip”

## 思考

1. 用自己的话谈一谈你对单元测试的理解。

# 设计测试用例

## 实验目的

1. 掌握黑盒测试用例的设计技术
2. 掌握等价类划分测试用例的设计方法
3. 掌握边界值分析测试用例的设计方法

## 功能描述

1. 三角形问题：要求输入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必须满足如下条件：
   * ，，
   * ，，

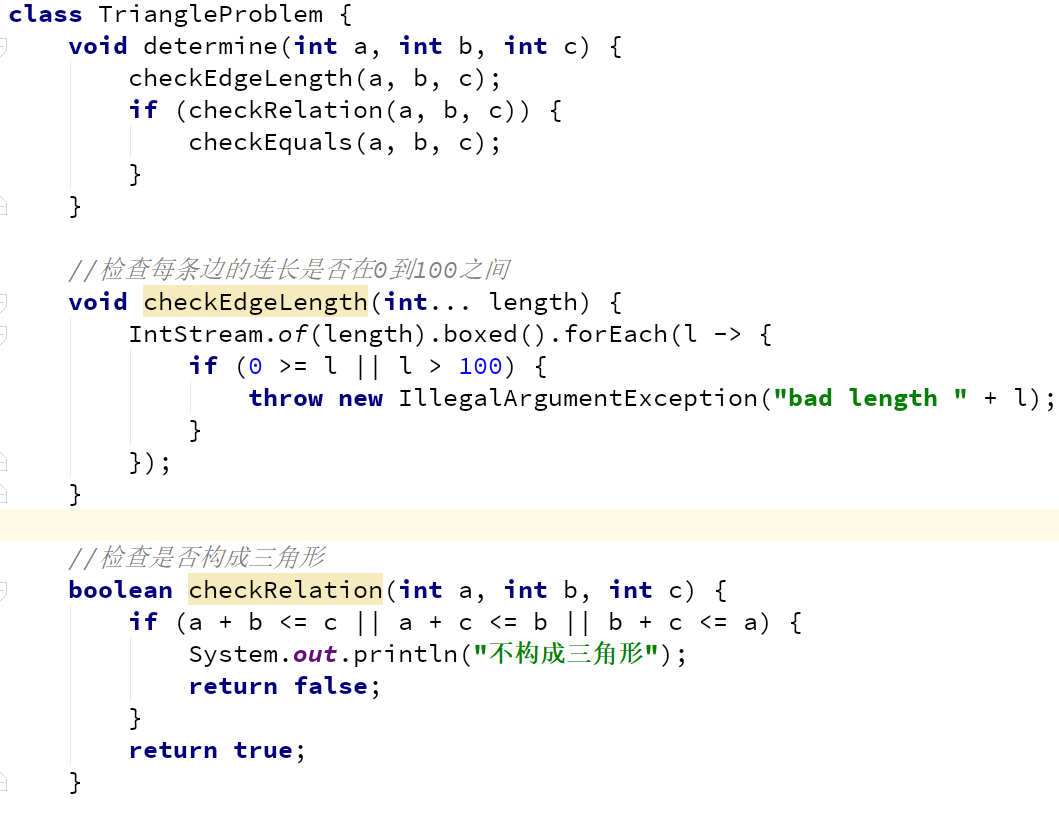
要求输出由这三条边构成的三角形的类型：等边、等腰、一般、非三角形

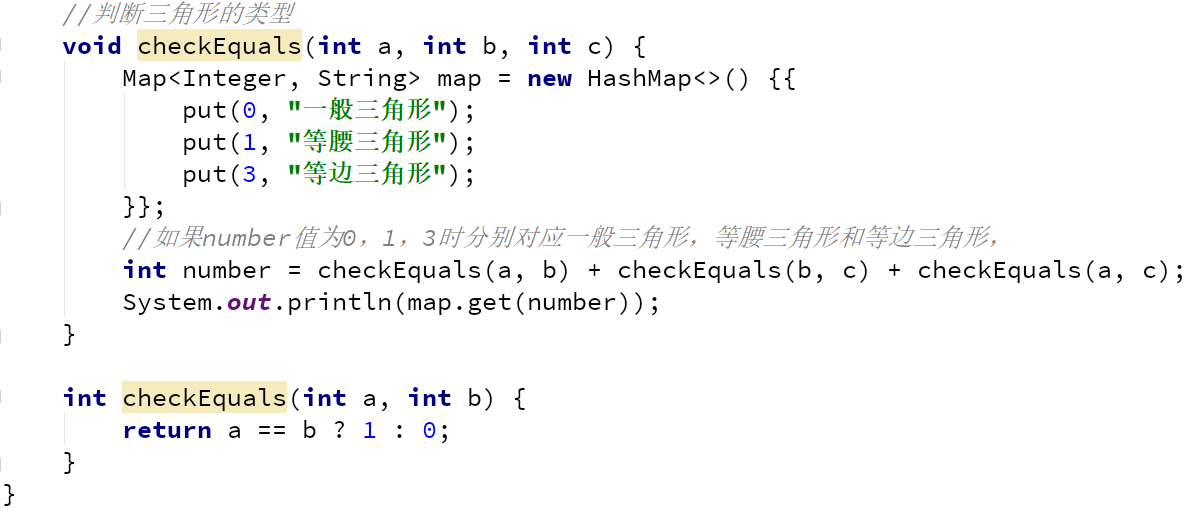
* + 如果不满足前三个输入条件，结束程序并打印信息
  + 如果满足前三个条件，但不满足后三个输入条件任何一个，输出“非三角形”
  + 如果满足前三个条件，且三条边长相等，则输出“等边三角形”
  + 如果满足前三个条件，恰好有两条边相等，则输出“等腰三角形”
  + 如果满足前三个条件，但三条边都不相等，则输出 “一般三角形”

## 实验内容

1. 使用IntelliJ IDEA创建名为blackbox-test的gradle工程并在build.gradle文件中添加JUnit5的依赖。在主目录src/main/java下创建blackbox.test的包并在其下创建TriangleProblem的类，并在该类中添加determine(int,int,int)的方法进行三角形类型判定。请使用等价类划分与边界值分析法分别设计测试用例。然后在测试目录src/test/java创建测试类TriangleProblemTest，使用JUnit5框架完成对这些测试用例的自动化测试。

以下是三角形问题的参考代码





## 实验要求

1. 将工程打包为名为black-test.jar包
2. 严格按照实验报告的模板编写实验报告。在实验报告中，必须包含：
   * 等价类划分，使用上课使用的表格
   * 按照等价类划分得到的测试用例表格
   * 边界值分析得到的测试用例表格
3. 将上述jar包与实验报告压缩为名为“你的学号\_blackbox.zip”的压缩包

## 思考

1. 你认为等价类划分与边界值分析法哪个更适合三角形问题？为什么？

# 评价测试用例

## 实验目的

1. 理解测试充分性评价与测试增强的概念
2. 掌握常见控制流充分性评价的测试准则
3. 了解测试覆盖工具的使用

## 功能描述

1. 同实验二的三角形问题

## 实验内容

1. **实验任务1**：使用IntelliJ IDEA创建名为whitebox-test的gradle工程并在build.gradle文件中添加JUnit5的依赖。在主目录src/main/java下创建whitebox.test的包并在其下创建TriangleProblem的类，在该类中添加determine(int,int,int)的方法进行三角形类型判定，参考代码可参照实验二。
2. **实验任务2**：使用路径覆盖(pathCoverage)、语句覆盖(statementCoverage)、条件覆盖(conditionCoverage)、判定覆盖(branchCoverage)、或条件/判定覆盖(conditionBranchCoverage)其中的一种设计测试用例。然后在测试目录src/test/java的whitebox.test包下创建测试类TriangleProblemTest，在其中添加对应覆盖准则的方法(如上)并使用JUnit5框架完成对这些测试用例的自动化测试。
3. **实验任务3**：使用C-USE覆盖准则(cuseCoverage)，P-USE覆盖准则(puseCoverage), ALL-USE覆盖准则(alluseCoverage)中的一种设计测试用例。在TriangleProblemTest类中添加对应的方法(方法名如上)并使用JUnit5框架完成对这些测试用例的自动化测试。

## 实验要求

1. 将工程打包为名为whitebox-test.jar包
2. 严格按照实验报告的模板编写实验报告。在实验报告中，对选择的每种覆盖准则：
   * 要画出控制流图和数据流图
   * 分析并描述其覆盖域，例如，如果选择路径覆盖，要列出所有的路径；如果选择C-USE覆盖准则，则要把每个变量定义所对应的dcu集合列出来。要求使用表格形式描述覆盖域。
   * 使用课堂上讲述的表格形式描述测试用例。每条测试用例要包含输入数据，预期结果，所覆盖的覆盖域中的元素
   * 计算你所设计的测试用例对于其对应覆盖准则的覆盖率，尽量做到完全覆盖。
3. 将上述jar包与实验报告压缩为名为“你的学号\_whitebox.zip”的压缩包

## 思考

1. 在实际的软件项目中，用自己的语言谈谈如何使用覆盖率评价测试用例。

# 使用系统测试框架

## 实验目的

1. 掌握系统测试的概念
2. 理解Selenium自动化测试的原理
3. 掌握Web应用测试的主要方法

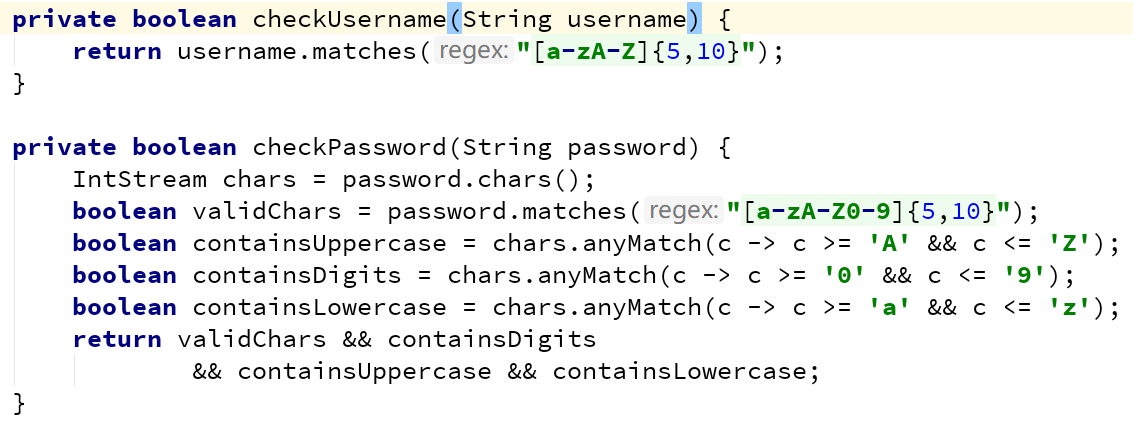
## 功能描述

1. 在Web应用中，Form表单的验证，如用户注册，是非常典型的应用场景。每一个表单其实都映射着相应的业务操作，因此，在填写表单时，必须遵守业务规则，否则输入是无效的。当用户填写完表单内容后，系统应该依据业务规则对输入的表单值进行验证，如果存在无效的输入，应该给予相应的提示。现有一个Web应用程序，它的基本功能是用户的注册功能。用户名和密码由字母和数字组成，且长度不少于5个字符，不多于10个字符。密码必须包含大小写字母和数字。

## 实验内容

1. 使用IntelliJ IDEA创建一个spring boot工程，工程名为selenium-test，在创建时选择web和thymeleaf组件。完成上述注册功能:
   * 在包selenium.test下创建User类和RegistrationApplication
   * 在主资源目录的templates框架下，新建两个html文件，register.html和success.html
   * 当用户在浏览器中输入<http://localhost/register>时，应该显示注册页面。用户输入结束后，点击注册按钮，如果输入有效，则注册成功，并重定向到success.html，否则跳转到register.html并显示注册失败
   * 使用黑盒测试技术设计测试用例，并使用selenium框架完成相应的测试代码。测试浏览器为chrome浏览器。

## 主要参考代码



## 实验要求

1. 将工程打包为名为selenium-test.jar包
2. 严格按照实验报告的模板编写实验报告。在实验报告中，必须包含测试用例的设计表格。
3. 将上述jar包与实验报告压缩为名为“你的学号\_selenium.zip”的压缩包

## 思考

1. 结合实验内容，用自己的语言谈谈Selenium框架的优缺点。

// parameterized test

@BeforeEach

Void init() {

driver

}

@AfterEach

Void teardown() {

Driver.quit();

}

test\_username\_cannot\_be\_empty() {

driver.get

element = driver.findElement(

element.sendkey(“”)

submit.submit();

}

Test\_password\_cannot\_be\_empty() {

driver =

driver.get

element = driver.findElement(

element.sendkey(“”)

submit.submit();

driver.quit();

}

学 号

天津城建大学

《软件测试与质量保证》

使用单元测试框架

|  |  |
| --- | --- |
| 学生姓名 |  |
| 专业、班级 |  |
| 指导教师 |  |
| 成绩 |  |

**计算机与信息工程学院计算机系**

**2018 年 3 月 20 日**

1. **实验内容**

描述本次实验内容

1. **设计与实现**

描述具体的测试的设计实现。使用合理的方式（如类图，关键代码片段，列表）描述测试用例及代码，不要简单地把代码复制过来了事。必要时可以把运行结果的截图包含进来

1. **问题与解决**

描述遇到的问题及解决办法

回答每个实验后的思考题