

SHANGHAI JIAO TONG UNIVERSITY



课程名称： 高级软件测试

学生姓名/学号: 杨扬 117037910042

学生姓名/学号: 汪沄 117037910033

学生姓名/学号: 王念 117037910035

学生姓名/学号: 刁子恒 117037910019

专 业: 软件工程

指导教师: 姚建国

学院(系): 电子信息与电气工程学院

目录

[一、简介 3](#_Toc497076302)

[1.1、编写目的 3](#_Toc497076303)

[1.2、测试范围 3](#_Toc497076304)

[二、测试资源 3](#_Toc497076305)

[2.1、人力资源 3](#_Toc497076306)

[2.2、测试对象 3](#_Toc497076307)

[2.3、测试环境 3](#_Toc497076308)

[2.4、测试工具 4](#_Toc497076309)

[三、测试过程 4](#_Toc497076310)

[3.1、ValidDate类 4](#_Toc497076311)

[3.1.1isLeap( )方法 4](#_Toc497076312)

[3.1.2 validCombine( )方法 5](#_Toc497076313)

[3.1.3 桩函数的应用 8](#_Toc497076314)

[3.2、CalRadius类 9](#_Toc497076315)

[3.2.1、isCollinear( )方法 9](#_Toc497076316)

[3.2.2、calRadius( )方法 10](#_Toc497076317)

[3.2.3 桩函数的应用 11](#_Toc497076318)

[四、遇到的困难及解决办法 12](#_Toc497076319)

[五、感想 13](#_Toc497076320)

# 一、简介

## 1.1、编写目的

使用等价类划分方法对于所选的两个模块进行测试，并使用 pair-wise 的方法减少测试用例，并比较两种方法的代码覆盖率。

## 1.2、测试范围

本次测试主要测试两个模块的功能，使用等价类划分的方法生成 Junit 测试，存在调用其他函数时，使用桩函数代替。并使用pair-wise方法减少测试用例，比较两种方法的覆盖率。

# 二、测试资源

## 2.1、人力资源

|  |  |  |
| --- | --- | --- |
| 姓名 | 角色 | 具体职责 |
| 杨扬 | 组长 | 代码与文档编写，总体策划 |
| 汪沄 | 方案设计 | 策划总体规模，测试内容 |
| 刁子恒 | 测试设计 | 规划测试方案 |
| 王念 | 测试执行 | 测试技术设计；记录测试情况 |

## 2.2、测试对象

|  |  |
| --- | --- |
| 测试对象 | 作用 |
| ValidDate | 判断输入的日期是否有效 |
| CalRadius | 给定三个点计算圆的半径 |

## 2.3、测试环境

|  |  |
| --- | --- |
| 测试环境 | 内容 |
| 软件环境 | Eclipse  jdk-9.0.1 |

## 2.4、测试工具

|  |  |  |  |
| --- | --- | --- | --- |
| 用途 | 工具 | 厂商 | 版本 |
| 测试 | JUnit | JUnit.org | 4.8.1 |
| 覆盖率计算 | IntelliJ IDEA code coverage runner | jetbrains |  |

# 三、测试过程

# 3.1、ValidDate类

## 3.1.1isLeap( )方法

isLeap( )方法判断输入年份是否为闰年。

* 根据输入条件划分等价类：

对于年份来说，限制有效输入在1500年到2050年之间。所以根据输入条件划分如下图所示：

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **输入域等价类** | **编号** | **有效等价类** | **编号** | **无效等价类** |
| 闰年 | 1 | 1500～2050，且是平年，如1997，2015 | 4 | 2050以上 |
|  | 2 | 1500～2050，且是非整百闰年，如2016 | 5 | 1500以下 |
|  | 3 | 1500～2050，且是整百闰年，如2000 |  |  |

* 生成测试用例：

|  |  |  |  |
| --- | --- | --- | --- |
| **年** | **覆盖类** | **状态** | **是否通过** |
| **1997** | **有效类1** |  | **通过** |
| **2016** | **有效类2** |  | **通过** |
| **2000** | **有效类3** |  | **通过** |
| **2051** | **无效类4** | **无效** | **通过** |
| **1499** | **无效类5** | **无效** | **通过** |

* 代码覆盖率：



## 3.1.2 validCombine( )方法

validCombine( )方法判断年月日的组合是否有效（是否遗漏了30天，月份不能为1,3,5,7,8,10,12；28天2月不是平年）。

* 根据输入条件划分等价类：

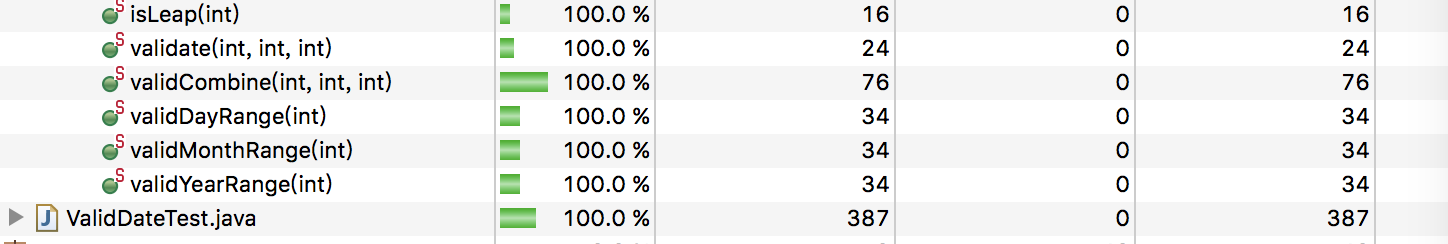
对于日期来说，输入必须在1～31之间；对于月份来说，输入必须在1～12之间，对于年份来说，限制有效输入在1500年到2050年之间。所以根据输入条件划分如下图所示：

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **输入域等价类** | **编号** | **有效等价类** | **编号** | **无效等价类** |
| 年 | 1 | 1500～2050，且是平年，如1997，2015 | 3 | 2050以上 |
|  | 2 | 1500～2050，且是闰年，如2000，2016 | 4 | 1500以下 |
| 月 | 5 | 1、3、5、7、8、10、12 | 8 | 1以下 |
|  | 6 | 4、6、9、11 | 9 | 12以上 |
|  | 7 | 2 |  |  |
| 日 | 10 | 1～28 | 14 | 1以下 |
|  | 11 | 29 | 15 | 31以上 |
|  | 12 | 30 |  |  |
|  | 13 | 31 |  |  |

* 生成测试用例：

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **年** | **月** | **日** | **覆盖类** | | | **状态** | **是否通过** |
| **1997** | **1** | **1** | **有效类1** | **有效类5** | **有效类10** |  | **通过** |
| **1997** | **1** | **29** | **有效类1** | **有效类5** | **有效类11** |  | **通过** |
| **1997** | **1** | **30** | **有效类1** | **有效类5** | **有效类12** |  | **通过** |
| **1997** | **1** | **31** | **有效类1** | **有效类5** | **有效类13** |  | **通过** |
| **1997** | **1** | **0** | **有效类1** | **有效类5** | **无效类14** | **无效** | **通过** |
| **1997** | **1** | **32** | **有效类1** | **有效类5** | **无效类15** | **无效** | **通过** |
| **1997** | **5** | **1** | **有效类1** | **有效类6** | **有效类10** |  | **通过** |
| **1997** | **5** | **29** | **有效类1** | **有效类6** | **有效类11** |  | **通过** |
| **1997** | **5** | **30** | **有效类1** | **有效类6** | **有效类12** |  | **通过** |
| **1997** | **5** | **31** | **有效类1** | **有效类6** | **有效类13** |  | **通过** |
| **1997** | **5** | **0** | **有效类1** | **有效类6** | **无效类14** | **无效** | **通过** |
| **1997** | **5** | **32** | **有效类1** | **有效类6** | **无效类15** | **无效** | **通过** |
| **1997** | **2** | **1** | **有效类1** | **有效类7** | **有效类10** |  | **通过** |
| **1997** | **2** | **29** | **有效类1** | **有效类7** | **有效类11** |  | **通过** |
| **1997** | **2** | **30** | **有效类1** | **有效类7** | **有效类12** |  | **通过** |
| **1997** | **2** | **31** | **有效类1** | **有效类7** | **有效类13** |  | **通过** |
| **1997** | **2** | **0** | **有效类1** | **有效类7** | **无效类14** | **无效** | **通过** |
| **1997** | **2** | **32** | **有效类1** | **有效类7** | **无效类15** | **无效** | **通过** |
| **1997** | **0** | **1** | **有效类1** | **无效类8** | **有效类10** | **无效** | **通过** |
| **1997** | **0** | **29** | **有效类1** | **无效类8** | **有效类11** | **无效** | **通过** |
| **1997** | **0** | **30** | **有效类1** | **无效类8** | **有效类12** | **无效** | **通过** |
| **1997** | **0** | **31** | **有效类1** | **无效类8** | **有效类13** | **无效** | **通过** |
| **1997** | **0** | **0** | **有效类1** | **无效类8** | **无效类14** | **无效** | **通过** |
| **1997** | **0** | **32** | **有效类1** | **无效类8** | **无效类15** | **无效** | **通过** |
| **1997** | **13** | **1** | **有效类1** | **无效类9** | **有效类10** | **无效** | **通过** |
| **1997** | **13** | **29** | **有效类1** | **无效类9** | **有效类11** | **无效** | **通过** |
| **1997** | **13** | **30** | **有效类1** | **无效类9** | **有效类12** | **无效** | **通过** |
| **1997** | **13** | **31** | **有效类1** | **无效类9** | **有效类13** | **无效** | **通过** |
| **1997** | **13** | **0** | **有效类1** | **无效类9** | **无效类14** | **无效** | **通过** |
| **1997** | **13** | **32** | **有效类1** | **无效类9** | **无效类15** | **无效** | **通过** |
| **2016** | **1** | **1** | **有效类2** | **有效类5** | **有效类10** |  | **通过** |
| **2016** | **1** | **29** | **有效类2** | **有效类5** | **有效类11** |  | **通过** |
| **2016** | **1** | **30** | **有效类2** | **有效类5** | **有效类12** |  | **通过** |
| **2016** | **1** | **31** | **有效类2** | **有效类5** | **有效类13** |  | **通过** |
| **2016** | **1** | **0** | **有效类2** | **有效类5** | **无效类14** | **无效** | **通过** |
| **2016** | **1** | **32** | **有效类2** | **有效类5** | **无效类15** | **无效** | **通过** |
| **2016** | **5** | **1** | **有效类2** | **有效类6** | **有效类10** |  | **通过** |
| **2016** | **5** | **29** | **有效类2** | **有效类6** | **有效类11** |  | **通过** |
| **2016** | **5** | **30** | **有效类2** | **有效类6** | **有效类12** |  | **通过** |
| **2016** | **5** | **31** | **有效类2** | **有效类6** | **有效类13** |  | **通过** |
| **2016** | **5** | **0** | **有效类2** | **有效类6** | **无效类14** | **无效** | **通过** |
| **2016** | **5** | **32** | **有效类2** | **有效类6** | **无效类15** | **无效** | **通过** |
| **2016** | **2** | **1** | **有效类2** | **有效类7** | **有效类10** |  | **通过** |
| **2016** | **2** | **29** | **有效类2** | **有效类7** | **有效类11** |  | **通过** |
| **2016** | **2** | **30** | **有效类2** | **有效类7** | **有效类12** |  | **通过** |
| **2016** | **2** | **31** | **有效类2** | **有效类7** | **有效类13** |  | **通过** |
| **2016** | **2** | **0** | **有效类2** | **有效类7** | **无效类14** | **无效** | **通过** |
| **2016** | **2** | **32** | **有效类2** | **有效类7** | **无效类15** | **无效** | **通过** |
| **2016** | **0** | **1** | **有效类2** | **无效类8** | **有效类10** | **无效** | **通过** |
| **2016** | **0** | **29** | **有效类2** | **无效类8** | **有效类11** | **无效** | **通过** |
| **2016** | **0** | **30** | **有效类2** | **无效类8** | **有效类12** | **无效** | **通过** |
| **2016** | **0** | **31** | **有效类2** | **无效类8** | **有效类13** | **无效** | **通过** |
| **2016** | **0** | **0** | **有效类2** | **无效类8** | **无效类14** | **无效** | **通过** |
| **2016** | **0** | **32** | **有效类2** | **无效类8** | **无效类15** | **无效** | **通过** |
| **2016** | **13** | **1** | **有效类2** | **无效类9** | **有效类10** | **无效** | **通过** |
| **2016** | **13** | **29** | **有效类2** | **无效类9** | **有效类11** | **无效** | **通过** |
| **2016** | **13** | **30** | **有效类2** | **无效类9** | **有效类12** | **无效** | **通过** |
| **2016** | **13** | **31** | **有效类2** | **无效类9** | **有效类13** | **无效** | **通过** |
| **2016** | **13** | **0** | **有效类2** | **无效类9** | **无效类14** | **无效** | **通过** |
| **2016** | **13** | **32** | **有效类2** | **无效类9** | **无效类15** | **无效** | **通过** |
| **2051** | **1** | **1** | **无效类3** | **有效类5** | **有效类10** | **无效** | **通过** |
| **2051** | **1** | **29** | **无效类3** | **有效类5** | **有效类11** | **无效** | **通过** |
| **2051** | **1** | **30** | **无效类3** | **有效类5** | **有效类12** | **无效** | **通过** |
| **2051** | **1** | **31** | **无效类3** | **有效类5** | **有效类13** | **无效** | **通过** |
| **2051** | **1** | **0** | **无效类3** | **有效类5** | **无效类14** | **无效** | **通过** |
| **2051** | **1** | **32** | **无效类3** | **有效类5** | **无效类15** | **无效** | **通过** |
| **2051** | **5** | **1** | **无效类3** | **有效类6** | **有效类10** | **无效** | **通过** |
| **2051** | **5** | **29** | **无效类3** | **有效类6** | **有效类11** | **无效** | **通过** |
| **2051** | **5** | **30** | **无效类3** | **有效类6** | **有效类12** | **无效** | **通过** |
| **2051** | **5** | **31** | **无效类3** | **有效类6** | **有效类13** | **无效** | **通过** |
| **2051** | **5** | **0** | **无效类3** | **有效类6** | **无效类14** | **无效** | **通过** |
| **2051** | **5** | **32** | **无效类3** | **有效类6** | **无效类15** | **无效** | **通过** |
| **2051** | **2** | **1** | **无效类3** | **有效类7** | **有效类10** | **无效** | **通过** |
| **2051** | **2** | **29** | **无效类3** | **有效类7** | **有效类11** | **无效** | **通过** |
| **2051** | **2** | **30** | **无效类3** | **有效类7** | **有效类12** | **无效** | **通过** |
| **2051** | **2** | **31** | **无效类3** | **有效类7** | **有效类13** | **无效** | **通过** |
| **2051** | **2** | **0** | **无效类3** | **有效类7** | **无效类14** | **无效** | **通过** |
| **2051** | **2** | **32** | **无效类3** | **有效类7** | **无效类15** | **无效** | **通过** |
| **2051** | **0** | **1** | **无效类3** | **无效类8** | **有效类10** | **无效** | **通过** |
| **2051** | **0** | **29** | **无效类3** | **无效类8** | **有效类11** | **无效** | **通过** |
| **2051** | **0** | **30** | **无效类3** | **无效类8** | **有效类12** | **无效** | **通过** |
| **2051** | **0** | **31** | **无效类3** | **无效类8** | **有效类13** | **无效** | **通过** |
| **2051** | **0** | **0** | **无效类3** | **无效类8** | **无效类14** | **无效** | **通过** |
| **2051** | **0** | **32** | **无效类3** | **无效类8** | **无效类15** | **无效** | **通过** |
| **2051** | **13** | **1** | **无效类3** | **无效类9** | **有效类10** | **无效** | **通过** |
| **2051** | **13** | **29** | **无效类3** | **无效类9** | **有效类11** | **无效** | **通过** |
| **2051** | **13** | **30** | **无效类3** | **无效类9** | **有效类12** | **无效** | **通过** |
| **2051** | **13** | **31** | **无效类3** | **无效类9** | **有效类13** | **无效** | **通过** |
| **2051** | **13** | **0** | **无效类3** | **无效类9** | **无效类14** | **无效** | **通过** |
| **2051** | **13** | **32** | **无效类3** | **无效类9** | **无效类15** | **无效** | **通过** |
| **1499** | **1** | **1** | **无效类4** | **有效类5** | **有效类10** | **无效** | **通过** |
| **1499** | **1** | **29** | **无效类4** | **有效类5** | **有效类11** | **无效** | **通过** |
| **1499** | **1** | **30** | **无效类4** | **有效类5** | **有效类12** | **无效** | **通过** |
| **1499** | **1** | **31** | **无效类4** | **有效类5** | **有效类13** | **无效** | **通过** |
| **1499** | **1** | **0** | **无效类4** | **有效类5** | **无效类14** | **无效** | **通过** |
| **1499** | **1** | **32** | **无效类4** | **有效类5** | **无效类15** | **无效** | **通过** |
| **1499** | **5** | **1** | **无效类4** | **有效类6** | **有效类10** | **无效** | **通过** |
| **1499** | **5** | **29** | **无效类4** | **有效类6** | **有效类11** | **无效** | **通过** |
| **1499** | **5** | **30** | **无效类4** | **有效类6** | **有效类12** | **无效** | **通过** |
| **1499** | **5** | **31** | **无效类4** | **有效类6** | **有效类13** | **无效** | **通过** |
| **1499** | **5** | **0** | **无效类4** | **有效类6** | **无效类14** | **无效** | **通过** |
| **1499** | **5** | **32** | **无效类4** | **有效类6** | **无效类15** | **无效** | **通过** |
| **1499** | **2** | **1** | **无效类4** | **有效类7** | **有效类10** | **无效** | **通过** |
| **1499** | **2** | **29** | **无效类4** | **有效类7** | **有效类11** | **无效** | **通过** |
| **1499** | **2** | **30** | **无效类4** | **有效类7** | **有效类12** | **无效** | **通过** |
| **1499** | **2** | **31** | **无效类4** | **有效类7** | **有效类13** | **无效** | **通过** |
| **1499** | **2** | **0** | **无效类4** | **有效类7** | **无效类14** | **无效** | **通过** |
| **1499** | **2** | **32** | **无效类4** | **有效类7** | **无效类15** | **无效** | **通过** |
| **1499** | **0** | **1** | **无效类4** | **无效类8** | **有效类10** | **无效** | **通过** |
| **1499** | **0** | **29** | **无效类4** | **无效类8** | **有效类11** | **无效** | **通过** |
| **1499** | **0** | **30** | **无效类4** | **无效类8** | **有效类12** | **无效** | **通过** |
| **1499** | **0** | **31** | **无效类4** | **无效类8** | **有效类13** | **无效** | **通过** |
| **1499** | **0** | **0** | **无效类4** | **无效类8** | **无效类14** | **无效** | **通过** |
| **1499** | **0** | **32** | **无效类4** | **无效类8** | **无效类15** | **无效** | **通过** |
| **1499** | **13** | **1** | **无效类4** | **无效类9** | **有效类10** | **无效** | **通过** |
| **1499** | **13** | **29** | **无效类4** | **无效类9** | **有效类11** | **无效** | **通过** |
| **1499** | **13** | **30** | **无效类4** | **无效类9** | **有效类12** | **无效** | **通过** |
| **1499** | **13** | **31** | **无效类4** | **无效类9** | **有效类13** | **无效** | **通过** |
| **1499** | **13** | **0** | **无效类4** | **无效类9** | **无效类14** | **无效** | **通过** |
| **1499** | **13** | **32** | **无效类4** | **无效类9** | **无效类15** | **无效** | **通过** |

* 代码覆盖率：

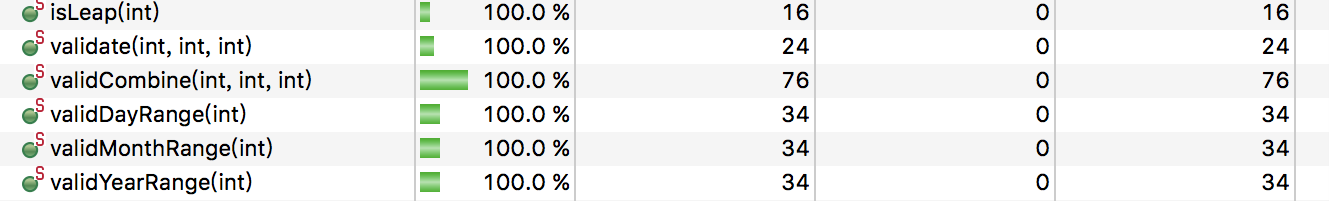


* 使用pair-wise减少测试用例

使用pair-wise后，将用例从120减少至24个。

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| 年份 | 月份 | 日 | **覆盖类** | | | **状态** | **是否通过** |
| 2051 | 1 | 31 | **无效类3** | **有效类5** | **无效类15** | **无效** | **通过** |
| 1499 | 1 | 32 | **无效类4** | **有效类5** | **无效类15** | **无效** | **通过** |
| 2016 | 1 | 29 | **有效类2** | **有效类5** | **有效类11** |  | **通过** |
| 2016 | 5 | 0 | **有效类2** | **有效类6** | **无效类14** | **无效** | **通过** |
| 1997 | 5 | 1 | **有效类1** | **有效类6** | **有效类10** |  | **通过** |
| 2016 | 2 | 32 | **有效类2** | **有效类7** | **无效类15** | **无效** | **通过** |
| 2051 | 5 | 32 | **无效类3** | **有效类6** | **无效类15** | **无效** | **通过** |
| 1499 | 2 | 29 | **无效类4** | **有效类7** | **有效类11** | **无效** | **通过** |
| 1997 | 1 | 29 | **有效类1** | **有效类5** | **有效类11** |  | **通过** |
| 1499 | 5 | 30 | **无效类4** | **有效类6** | **有效类12** | **无效** | **通过** |
| 1997 | 2 | 31 | **有效类1** | **有效类7** | **有效类13** |  | **通过** |
| 1997 | 1 | 30 | **有效类1** | **有效类5** | **有效类12** |  | **通过** |
| 1499 | 2 | 0 | **无效类4** | **有效类7** | **无效类14** | **无效** | **通过** |
| 1499 | 5 | 31 | **无效类4** | **有效类6** | **有效类13** | **无效** | **通过** |
| 1997 | 1 | 0 | **有效类1** | **有效类5** | **无效类14** | **无效** | **通过** |
| 2051 | 2 | 30 | **无效类3** | **有效类7** | **有效类12** | **无效** | **通过** |
| 2051 | 2 | 1 | **无效类3** | **有效类7** | **有效类10** | **无效** | **通过** |
| 1997 | 5 | 32 | **有效类1** | **有效类6** | **无效类15** | **无效** | **通过** |
| 1499 | 1 | 1 | **无效类4** | **有效类5** | **有效类10** | **无效** | **通过** |
| 2016 | 5 | 30 | **有效类2** | **有效类6** | **有效类12** |  | **通过** |
| 2016 | 1 | 31 | **有效类2** | **有效类5** | **有效类13** |  | **通过** |
| 2016 | 2 | 1 | **有效类2** | **有效类7** | **有效类10** |  | **通过** |
| 2051 | 1 | 0 | **无效类3** | **有效类5** | **无效类14** | **无效** | **通过** |
| 2051 | 5 | 29 | **无效类3** | **有效类6** | **有效类11** | **无效** | **通过** |

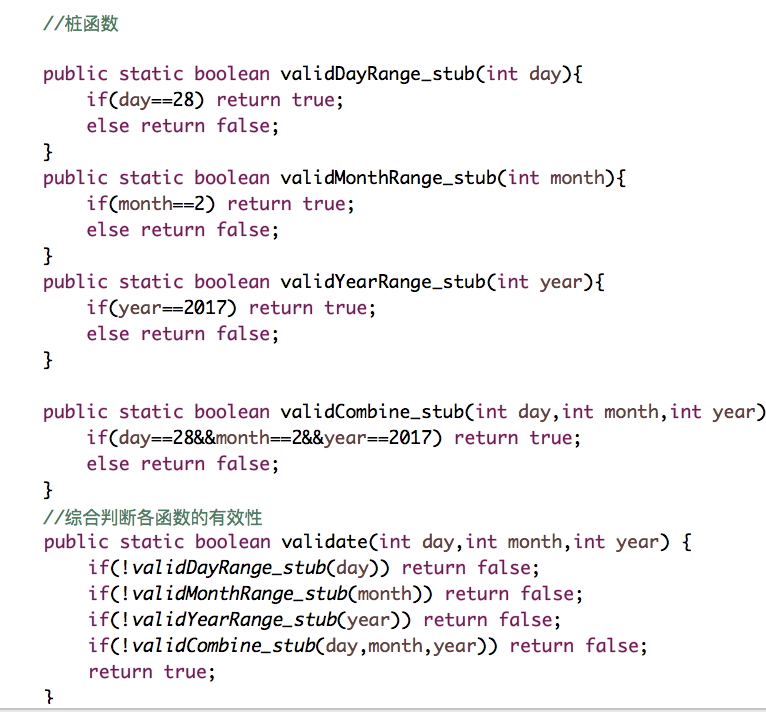
可以看到覆盖率依然为100%。



## 3.1.3 桩函数的应用

在对函数validate的测试中，我们使用了打桩这一方法。

下图为对validate这个函数用2017年2月28日这个日期进行测试时使用桩函数的示例。Validate函数测试用例可见所附代码，覆盖率见上图。



# 3.2、CalRadius类

## 3.2.1、isCollinear( )方法

isCollinear( )方法判断三点是否共线，当三点共线时，无法唯一确定三点共圆。

* 根据输入条件划分等价类：

对于二维平面坐标系，输入坐标并没有限制要求，对于给定三点坐标，判定是否在一条直线上的方法，接收三个参数皆为点（Point）对象。所以根据输入条件划分如下图所示：

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **输入域等价类** | **编号** | **有效等价类** | **编号** | **无效等价类** |
| p1 | 1 | 输入三点共线 | 4 | 输入参数为null |
| p2 | 2 | 输入存在两点为同一点 | 5 | 输入三点不共线 |
| p3 | 3 | 输入三点共点 |  |  |

* 生成测试用例：在p1，p2，p3的取值中，我们用x,y 指代Point对象的横坐标与纵坐标

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **用例** | **p1** | **p2** | **p3** | **预期输出** | **覆盖的有效等价类** | **覆盖的无效等价类** | **实际测试结果** |
| 1 | 0,0 | 0,4 | 0,8 | true | 1 |  | true |
| 2 | 0,0 | 0,4 | 3,0 | flase |  | 5 | false |
| 3 | 0,0 | 0,0 | 0,8 | true | 2 |  | true |
| 4 | 0,0 | 0,0 | 0,0 | true | 3 |  | true |
| 5 | null | null | null | 异常 |  | 4 | 异常 |

* 代码覆盖率：

方法代码行数覆盖率100%，分支覆盖率100%

**../Library/Containers/com.tencent.qq/Data/Library/Caches/Images/0C41B83DB0ED0126D7A7A3441D74422D.jpg**

在对函数isCollinear的测试中，我们对equal函数进行了打桩。

用桩函数替代进行测试，具体可见所附代码。

## 3.2.2、calRadius( )方法

calRadius() 方法在给定三个点的情况下计算出这三点是否唯一确定共圆，若唯一确定，返回一个Circle对象，其中包含圆心的Point对象和Double类型的半径长度。若不能确定唯一的共圆，则返回null。

* 根据输入条件划分等价类：

对于二维平面坐标系，输入坐标并没有限制要求，对于给定三点坐标，判定是否在一条直线上的方法，接收三个参数皆为点（Point）对象。所以根据输入条件划分如下图所示：

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **输入域等价类** | **编号** | **有效等价类** | **编号** | **无效等价类** |
| p1 | 1 | 输入三点不共线 | 2 | 输入参数为null |
| p2 |  |  | 3 | 输入三点共线 |
| p3 |  |  | 4 | 输入存在两点为同一点 |
|  |  |  | 5 | 输入三点共点 |

* 生成测试用例：

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **用例** | **p1** | **p2** | **p3** | **预期输出** | **覆盖的有效等价类** | **覆盖的无效等价类** | **实际测试结果** |
| 1 | 0,0 | 0,4 | 0,8 | null |  | 3 | null |
| 2 | 0,8 | 0,8 | 0,8 | null |  | 5 | null |
| 3 | 0,0 | 0,4 | 3,0 | (1.5,2.5),r=2.5 | 1 |  | (1.5,2.5),r=2.5 |
| 4 | 0,0 | 0,0 | 0,8 | null |  | 4 | null |
| 5 | null | null | null | null |  | 2 | null |

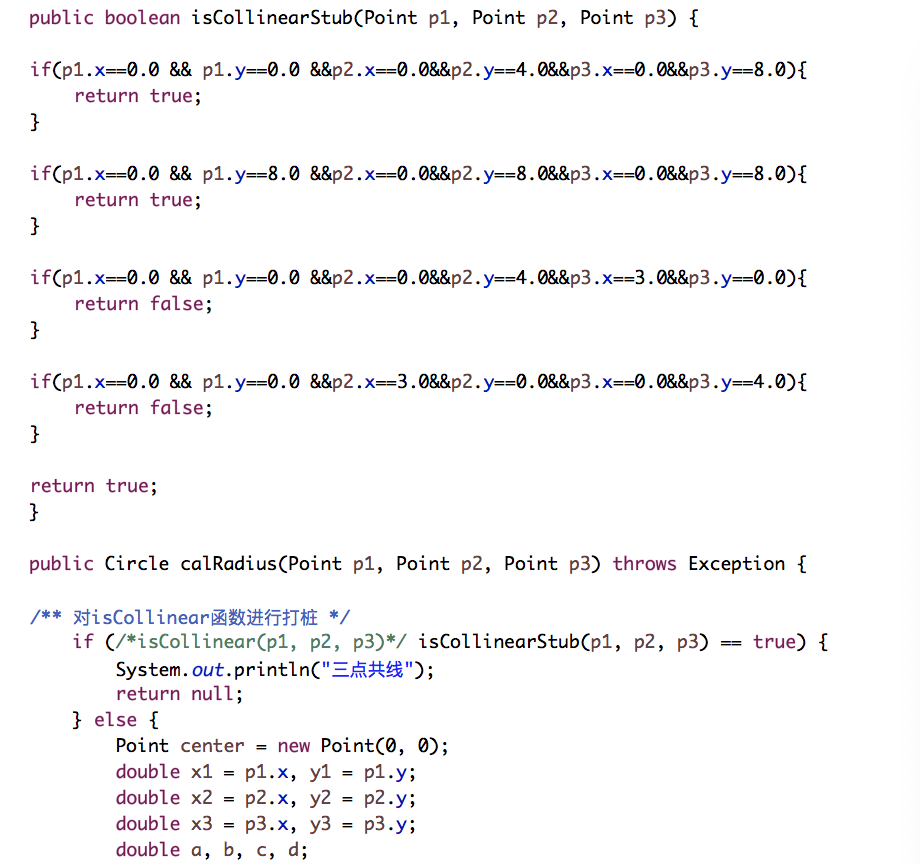
* 代码覆盖率：

**../Library/Containers/com.tencent.qq/Data/Library/Caches/Images/C7F77D3E9BBDA8C0C6706627F58626A9.jpg**

## 3.2.3 桩函数的应用

在对函数calRadius的测试中，我们对isCollinear函数进行了打桩。

用桩函数替代进行测试。

****

# 四、遇到的困难及解决办法

* **问题一**：

团队协作问题，由于主要交流方式为线上交流，所以良好的交流与协作工具有利于提高团队效率。

* **方法和总结：**

采取的解决方案如下：

1. 使用 GitHub 代码托管平台进行对整个项目工程的代码进行版本管理。
2. 采用即时交流工具进行组内交流。
3. 定期举行线下讨论。

* **问题二**：

集成开发环境和版本管理工具的使用问题。团队成员统一采用Intellij Idea，在通过版本管理工具进行工程文件的导入和修改时出现了工程报错，以及和git 仓库中的状态不统一的问题。

* **方法和总结：**

采取的解决方案如下：

1. 对于项目工程中的代码部分，统一使用src/main 作为源代码路径，src/test作为测试代码路径，并统一使用IntellijIdea中的相关设置；项目使用 cn.edu.sjtu.software 作为包名。
2. 每次进行工程编辑前从git远端拉取最新进度，编辑完成后确保编译无问题及时上传。避免另开项目复制粘贴等不符合git版本管理实现的方法，出现代码冲突问题及时沟通进行处理

* **问题三:**

由于小组成员中有多名来自其他专业，最明显的差距就是基础知识薄弱，如java编程语言急需系统的学习，对IntelliJ IDEA等软件的操作还不够熟悉，这也给我们今后的学习指出了明确的方向。

* **方法和总结：**

采取的解决方案如下：

在做大作业之前，就通过慕课等网络平台学习相关课程，还有队员专门学习了《Java无难事》，并做了详细笔记，期间结合大作业所需内容一边复习课上知识点，一边查阅相关资料文档，以求将理论与实践相结合，并确保小组内部能独立完成。

* **问题四**

对项目全局把握不够准确。

由于本次作业要求提供具有源码的程序，起初我们自己想结合课上所讲内容自己编写两个项目，虽然出发点是好的，但后期发现所编写项目很难符合作业要求，如组员集思广益选择的一元二次方程项目过多关注打桩，但测试用例却较少，且难于与pair-wise方法结合。

* **方法和总结：**

采取的解决方案如下：

通过多次请教助教果断重起炉灶，重新设计项目，所以第二次所选择的项目相较于第一次更适合本次作业。今后我们再遇到类似问题时需要做更全面的统筹。

* **问题五**

作业进度没有把握好

虽然为了准备此次作业，我们从第一节课得知作业使用Java语言编程后提前很久就开始学习相关课程，但也恰恰因为过度关注基础编程知识学习而没有提早开始大作业，总想着把java语言学深入一些再启动，直接导致作业后期时间紧张。需知实践出真知，大作业完成期间的进步更胜过单一看书学习的效果。

# 五、感想

最后，感谢姚老师和助教彭博学长的耐心的指导，期间组内成员多次向彭博学长请教大作业中不确定的细节，尤其是我们前期确定的一元二次方程项目出现漏洞时，彭博学长帮我们做了细致的分析，使我们意识到了需要做的改进。组员一致表示，这次大作业使我们将理论与实践相结合，并一定程度上提高了专业技能，从而加深了对课上所学知识的理解。