

电子科技大学

实验报告

学生姓名：Lolipop 学号：2018091202000 指导教师：xx

实验地点：信软楼 304 实验时间：2019.12.12

一、实验名称：软件工程基础编码与测试

二、实验学时：4

三、实验目的：

1. 掌握软件编码规范
2. 掌握单元测试的方法，包括白盒测试和黑盒测试
3. 能对实际程序单元测试，能够撰写简单的测试报告

四、实验原理：

1. 软件质量可以明确表示是否符合功能和性能要求，明确记载开发标准和所有专业开发软件的期望的隐性特点。
2. 软件测试是在某种指定的条件下对系统或组件的某些方面进行评估的过程，是分析软件各项目以检测现有的结果和应有结果之间的差异，并评估软件各项的特征的过程。
3. 软件测试是软件质量保证的主要方法，占总成本的 60%以上。
4. 软件测试的评估标准：覆盖率，故障插入，变异分值
5. 软件测试技术：黑盒测试，白盒测试，灰盒测试
6. 黑盒测试：忽略程序的内部机制，测试程序的每项功能是否符合要求
7. 白盒测试：考虑程序的内部机制，测试程序的每种内部操作是否符合要求
8. 单元测试又称模块测试，是针对软件设计的最小单位——程序模块，进行正确性检验的测试工作。其目的在于发现各模块内部可能存在的各种差错。
9. 单元测试的内容：模块接口测试、局部数据结构测试、路径测试、错误处理测试、边界测试、

五、实验内容：

1. 输入三个整数 a, b 和 c 分别作为三角形的三条边，判断由这三条边构成

的三角形类型是：等边三角形、等腰三角形、一般三角形、或非三角形（即不能构成一个三角形）。

- ① 编写该程序，注意编程规范
 - ② 按照等价类划分方法设计测试用例，并进行测试
 - ③ 按照边界值方法设计测试用例，并进行测试
2. 输入不超过 50 个数（以-1 作为输入结束标志），计算其中有效的学生分数的个数，及有效分数的总分和平均值。
 - ① 编写程序，注意编程规范
 - ② 导出程序的控制流图，确定独立路径集合（基本路径集合）
 - ③ 为每条路径设计测试用例，并进行测试
 3. 细化实验 2 的系统设计，选择 2 个关键模块进行编码实现
 4. 试用白盒或黑盒方法测试实现的模块
 5. 撰写软件测试报告

六、 实验器材（设备、元器件）：

搭载 Windows 操作系统的个人电脑

七、 实验步骤：

1. 根据编写规范，编写所需实验和项目代码。
2. 按照等价类划分法测试判断三角形程序，得到相应数据结果。
3. 按照边界值方法测试判断三角形程序，得到相应数据结果。
4. 导出计算学生分数程序的控制流图，得到独立路径集合。
5. 为得到的计算学生分数程序的独立路径集合设计测试用例，进行测试并记录相应数据结果。
6. 使用黑盒测试对项目实现的模块进行测试，记录测试结果。
7. 根据上述测试所得到的结果进行软件测试报告的撰写。

八、 实验结果与分析（含重要数据结果分析或核心代码流程分析）

- 1) 判断三角形程序等价类划分测试，划分类如表 1 所示。

表 1 判断三角形程序等价类划分表

输入条件	有效等价类	无效等价类
三条边长 a, b, c	三条边长相等，两边之和大于第三边（1） 两条边长相等，两边之和大于第三边（2） 边长各不相同，两边之和大于第三边（3）	两边之和小于等于第三边（4）

- ① $a = 3, b = 3, c = 3$ 三条边长相等，两边之和大于第三边
- ② $a = 3, b = 4, c = 4$ 两条边长相等，两边之和大于第三边

- ③ $a = 3, b = 4, c = 5$ 三边长不相等，两边之和大于第三边
 ④ $a = 0, b = 3, b = 3$ 两边之和小于等于第三边

2) 判断三角形程序边界值分析测试，数据表 2 所示。

表 2 判断三角形程序边界值分析表

序号	输入条件			预期输出
	a	b	c	
1	100	100	1	等腰三角形
2	100	100	0	非三角形
3	100	1	100	等腰三角形
4	100	0	100	非三角形
5	1	100	100	等腰三角形
6	0	100	100	非三角形
7	100	100	199	等腰三角形
8	100	100	200	非三角形
9	101	100	99	一般三角形
10	100	200	100	非三角形
11	199	100	100	等腰三角形
12	200	100	100	非三角形

3) 计算学生分数程序控制流图，如图 1 所示。

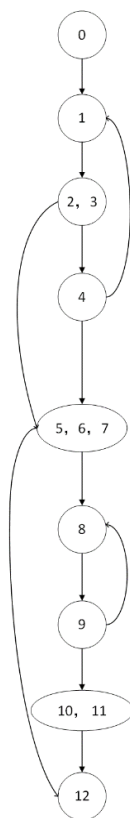


图 1 计算学生分数程序控制流图

4) 计算学生分数程序基本路径集合及测试用例:

① 0-1-2-3-5-6-7-12 (只输入-1)

```
Input students' grades(a blank to divide)
-1
未输入有效成绩！
```

② 0-1-2-3-4-1-2-3-5-6-7-12 (输入不超过 50 个无效的成绩)

```
Input students' grades(a blank to divide)
hhoi uoikfgj hjfowifhoi -1
Exception in thread "main" java.util.InputMismatchException
    at java.base/java.util.Scanner.throwFor(Scanner.java:939)
    at java.base/java.util.Scanner.next(Scanner.java:1594)
    at java.base/java.util.Scanner.nextFloat(Scanner.java:2496)
    at Grade.main(Grade.java:18)
```

③ 0-1-2-3-4-1-2-3-4-5-6-7-12 (输入超过 50 个无效成绩)

[illegible]

④ 0-1-2-3-4-1-2-3-4-5-6-7-8-9-8-10-11-12 (输入超过 50 个有效成绩)

[illegible]

⑤ 0-1-2-3-4-1-2-3-5-6-7-8-9-10-11-12 (输入 1 个有效成绩)

```
Input students' grades(a blank to divide)
100 -1
有效成绩数：1
有效成绩总分：100.0
有效成绩均分：100.0
```

⑥ 0-1-2-3-4-1-2-3-5-6-7-8-9-8-10-11-12(输入大于 1 个不超过 50 个有效成绩)

```
Input students' grades(a blank to divide)
91 92 93 94 95 -1
有效成绩数：5
有效成绩总分：465.0
有效成绩均分：93.0
```

5) 人事资源管理系统黑盒测试等价类划分如表 3 所示。

表 3 人事资源管理系统黑盒测试等价类划分

输入条件	有效等价类	无效等价类
考勤日期设置	考勤日期为当前日期及之后 (1)	考勤日期在当前日期之前 (3)
开始考勤时间与结束考勤时间	开始考勤时间在结束考勤时间之前 (2)	开始考勤时间在结束考勤时间之后 (4)

- ① 考勤日期在当前日期之前, 开始考勤时间在结束考勤时间之后 (3)(4)
- ② 考勤日期在当前日期之前, 开始考勤时间在结束考勤时间之前 (3)(2)
- ③ 考勤日期为当前日期及之后, 开始考勤时间在结束考勤时间之后 (1)(4)
- ④ 考勤日期为当前日期及之后, 开始考勤时间在结束考勤时间之前 (1)(2)

九、 总结及心得体会：

本次实验涉及了编码开发和测试软件两个部分, 是项目开发阶段的提炼。结合以往开发的经验, 在开发设计判断三角形程序和计算学生分数程序时, 自然而然地添加了一些数据校验的操作。但是测试时仍有导致程序崩溃停止运行的情况, 此时就颇感测试的重要性, 可以帮助软件开发时考虑到种种想得到想不到的情况, 并设身处地地站在用户的角度, 去增加用户的体验, 降低用户因为不小心的输入错误而出现意外的情况。

经过测试所发现的问题, 可以帮助我们进一步编写更加健壮且安全的代码, 开发出更加现代高效的程序, 满足用户的需求。

十、 对本实验过程及方法、手段的改进建议：

在编码开发时可以参考学习较为知名的编码规范, 如阿里公司程序员编码规范等, 规范自己代码的书写, 提高代码的可读性, 增加程序可维护性。

测试用例时综合考虑白盒测试和黑盒测试的特性, 选择更好的方法或两者都进行来对软件进行测试, 增加测试的可靠性, 为维护和进一步编写代码提供依据。

报告评分：

指导教师签字：

附录：

软件测试报告

项目名称：人事资源管理系统

日期：2019/12/22

1 引言

1.1 目的

本软件测试报告为 2019 年电子科技大学软件工程课程实验——开发设计人事资源管理系统项目的测试报告，旨在：

- 1. 总结开发测试阶段的结果，描述系统开发进程是否符合需求。
- 2. 分析测试的过程，信息，为以后制定相关测试计划提供参考。
- 3. 分析系统存在的缺陷，为修复和预防程序 Bug 提供建议。

1.2 范围

测试范围包括人事资源管理系统开发中的“试用员工”和“添加考勤项目”两个模块。

1.3 预期的读者和阅读建议

预期读者包括测试人员、开发人员、项目管理者、质量管理人员和项目预期使用者。建议阅读时始终怀揣“吹毛求疵”的心理，发现项目测试管理中仍不足的地方，在下一次进行修补改正。

1.4 缩写和术语

1.5 参考文献

2 测试时间、地点和人员

序号	测试时间	地点	人员
1	2019/12/22	电子科技大学	xxx
2	2019/12/22	电子科技大学	xxx
3	2019/12/22	电子科技大学	xx

3 测试环境及方法

使用设备为 Pixel 2，基于 Android Studio 自带的模拟器与 Android API 29。
选择采用黑盒测试的测试方法。

4 测试用例执行情况

测试用例序号	1
测试目的及说明	测试“试用员工”模块，验证前后端数据交互功能
测试输入	选择试用“小明”
期望测试结果	“小明”移出“试用员工”界面
判断准则	数据库中“待用员工”中不再存在“小明”的相关信息，而“在职员工”中出现“小明”的相关信息
实际测试结果	刷新“试用员工”界面不再显示“小明”；查看数据库，“在职员工”中出现“小明”的相关信息
测试人员	xxx
测试日期	2019/12/22
测试评价	通过测试
问题标识	
问题处理	

测试用例序号	2
测试目的及说明	测试“添加考勤项目”模块，验证模块逻辑
测试输入	设定项目名为“test”；设定考勤日期在当前日期之前；设定开始考勤时间在结束考勤时间之后
期望测试结果	出现相关警告提示，并拒绝操作
判断准则	考勤日期在当前日期之后，开始考勤时间在结束考勤时间之前
实际测试结果	设定的考勤日期和开始考勤时间时分别弹出警告提示窗口，对应数据不发生修改
测试人员	xxx
测试日期	2019/12/22

测试评价	通过测试
问题标识	
问题处理	

测试用例序号	3
测试目的及说明	测试“添加考勤项目”模块，验证模块逻辑
测试输入	设定项目名为空；设定考勤日期在当前日期之后；设定开始考勤时间在结束考勤时间之前
期望测试结果	出现相关警告提示，并拒绝操作
判断准则	项目无法保存，并提示“项目名不能为空”
实际测试结果	提示“项目名不能为空”，拒绝用户保存操作
测试人员	xxx
测试日期	2019/12/22
测试评价	通过测试
问题标识	
问题处理	

测试用例序号	4
测试目的及说明	测试“添加考勤项目”模块，验证前后端数据交互功能
测试输入	设定项目名为“test”；设定考勤日期在当前日期之后；设定开始考勤时间在结束考勤时间之前
期望测试结果	考勤项目成功保存
判断准则	数据库中出现“test”的考勤项目
实际测试结果	数据库中出现“test”的考勤项目；刷新“考勤项目”页面显示“test”考勤项目
测试人员	xx
测试日期	2019/12/22
测试评价	通过测试
问题标识	
问题处理	

5 测试结果分析

5.1 测试执行的充分性

本次测试考虑了“试用员工”模块和“添加考勤信息”模块的主要的输入值和容易发生错误的输入值，较为充分地完成了相关的测试执行。

5.2 缺陷个数、分类及原因分析

5.3 遗留问题

未测试当数据库放在实际服务器上时，软件通过网络连接时前后端交互情况。

6 测试结论

本次测试基本通过了软件的可能输入情况，遗留的问题将在以后将服务器搭载在服务器上后进一步测试解决。

通过测试，可以发现本软件所开发的模块已基本实现前后端数据库交互的功能，满足公司系统对数据共享的需求；此外还具备基本的数据校验功能，能够降低用户操作错误的出现可能。

在日后的开发中应该再接再厉，最终实现软件的长期稳定运行。