

**2020年春季学期  
计算机学院《软件构造》课程**

**Lab 4实验报告**

|  |  |
| --- | --- |
| 姓名 |  |
| 学号 |  |
| 班号 |  |
| 电子邮件 |  |
| 手机号码 |  |

**目录**

[1 实验目标概述 1](#_Toc39518185)

[2 实验环境配置 1](#_Toc39518186)

[3 实验过程 1](#_Toc39518187)

[3.1 Error and Exception Handling 2](#_Toc39518188)

[3.1.1 处理输入文本中的三类错误 2](#_Toc39518189)

[3.1.2 处理客户端操作时产生的异常 2](#_Toc39518190)

[3.2 Assertion and Defensive Programming 2](#_Toc39518191)

[3.2.1 checkRep()检查rep invariants 2](#_Toc39518192)

[3.2.2 Assertion/异常机制来保障pre-/post-condition 2](#_Toc39518193)

[3.2.3 你的代码的防御式策略概述 2](#_Toc39518194)

[3.3 Logging 2](#_Toc39518195)

[3.3.1 异常处理的日志功能 2](#_Toc39518196)

[3.3.2 应用层操作的日志功能 2](#_Toc39518197)

[3.3.3 日志查询功能 2](#_Toc39518198)

[3.4 Testing for Robustness and Correctness 2](#_Toc39518199)

[3.4.1 Testing strategy 2](#_Toc39518200)

[3.4.2 测试用例设计 2](#_Toc39518201)

[3.4.3 测试运行结果与EclEmma覆盖度报告 2](#_Toc39518202)

[3.5 SpotBugs tool 2](#_Toc39518203)

[3.6 Debugging 3](#_Toc39518204)

[3.6.1 EventManager程序 3](#_Toc39518205)

[3.6.2 LowestPrice程序 3](#_Toc39518206)

[3.6.3 FlightClient/Flight/Plane程序 3](#_Toc39518207)

[4 实验进度记录 3](#_Toc39518208)

[5 实验过程中遇到的困难与解决途径 4](#_Toc39518209)

[6 实验过程中收获的经验、教训、感想 4](#_Toc39518210)

[6.1 实验过程中收获的经验和教训 4](#_Toc39518211)

[6.2 针对以下方面的感受 4](#_Toc39518212)

# 实验目标概述

根据实验手册简要撰写。

# 实验环境配置

简要陈述你配置本次实验所需环境的过程，必要时可以给出屏幕截图。

特别是要记录配置过程中遇到的问题和困难，以及如何解决的。

在这里给出你的GitHub Lab4仓库的URL地址（Lab4-学号）。

# 实验过程

请仔细对照实验手册，针对每一项任务，在下面各节中记录你的实验过程、阐述你的设计思路和问题求解思路，可辅之以示意图或关键源代码加以说明（但千万不要把你的源代码全部粘贴过来！）。

## Error and Exception Handling

### 处理输入文本中的三类错误

### 处理客户端操作时产生的异常

## Assertion and Defensive Programming

### checkRep()检查rep invariants

### Assertion/异常机制来保障pre-/post-condition

### 你的代码的防御式策略概述

## Logging

### 异常处理的日志功能

### 应用层操作的日志功能

### 日志查询功能

## Testing for Robustness and Correctness

### Testing strategy

### 测试用例设计

### 测试运行结果与EclEmma覆盖度报告

## SpotBugs tool

发现了哪些错误，每种错误代表什么不良的编程习惯

对代码修改，消除这些错误。

## Debugging

### EventManager程序

理解待调试程序的代码思想

发现并定位错误的过程

你如何修正错误

修复之后的测试结果

### LowestPrice程序

理解待调试程序的代码思想

发现并定位错误的过程

你如何修正错误

修复之后的测试结果

### FlightClient/Flight/Plane程序

理解待调试程序的代码思想

发现并定位错误的过程

你如何修正错误

修复之后的测试结果

# 实验进度记录

请使用表格方式记录你的进度情况，以超过半小时的连续编程时间为一行。

每次结束编程时，请向该表格中增加一行。不要事后胡乱填写。

不要嫌烦，该表格可帮助你汇总你在每个任务上付出的时间和精力，发现自己不擅长的任务，后续有意识的弥补。

|  |  |  |  |
| --- | --- | --- | --- |
| 日期 | 时间段 | 计划任务 | 实际完成情况 |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |

# 实验过程中遇到的困难与解决途径

|  |  |
| --- | --- |
| 遇到的难点 | 解决途径 |
|  |  |
|  |  |
|  |  |

# 实验过程中收获的经验、教训、感想

## 实验过程中收获的经验和教训

## 针对以下方面的感受

1. 健壮性和正确性，二者对编程中程序员的思路有什么不同的影响？
2. 为了应对1%可能出现的错误或异常，需要增加很多行的代码，这是否划算？（考虑这个反例：民航飞机上为何不安装降落伞？）
3. “让自己的程序能应对更多的异常情况”和“让客户端/程序的用户承担确保正确性的职责”，二者有什么差异？你在哪些编程场景下会考虑遵循前者、在哪些场景下考虑遵循后者？
4. 过分谨慎的“防御”（excessively defensive）真的有必要吗？你如何看待过分防御所带来的性能损耗？如何在二者之间取得平衡？
5. 通过调试发现并定位错误，你自己的编程经历中有总结出一些有效的方法吗？请分享之。Assertion和log技术是否会帮助你更有效的定位错误？
6. 怎么才是“充分的测试”？代码覆盖度100%是否就意味着100%充分的测试？
7. Debug一个错误的程序，有乐趣吗？体验一下无注释、无文档的程序修改。
8. 关于本实验的工作量、难度、deadline。
9. 到目前为止你对《软件构造》课程的评价和建议。
10. 期末考试临近，你对占成绩60%的闭卷考试有什么预期？