

**2019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](#_Toc3922818)

[2 实验环境配置 1](#_Toc3922819)

[3 实验过程 1](#_Toc3922820)

[3.1 Error and Exception Handling 2](#_Toc3922821)

[3.2 Assertion and Defensive Programming 2](#_Toc3922822)

[3.2.1 checkRep()检查invariants 2](#_Toc3922823)

[3.2.2 Assertion保障pre-/post-condition 2](#_Toc3922824)

[3.3 Logging 2](#_Toc3922825)

[3.3.1 写日志 2](#_Toc3922826)

[3.3.2 日志查询 2](#_Toc3922827)

[3.4 Testing for Robustness and Correctness 2](#_Toc3922828)

[3.4.1 Testing strategy 2](#_Toc3922829)

[3.4.2 测试用例设计 2](#_Toc3922830)

[3.4.3 测试运行结果与EclEmma覆盖度报告 2](#_Toc3922831)

[3.5 SpotBugs tool 2](#_Toc3922832)

[3.6 Debugging 3](#_Toc3922833)

[3.6.1 理解待调试程序的代码思想 3](#_Toc3922834)

[3.6.2 发现并定位错误的过程 3](#_Toc3922835)

[3.6.3 如何修正错误 3](#_Toc3922836)

[3.6.4 结果 3](#_Toc3922837)

[4 实验进度记录 3](#_Toc3922838)

[5 实验过程中遇到的困难与解决途径 3](#_Toc3922839)

[6 实验过程中收获的经验、教训、感想 4](#_Toc3922840)

[6.1 实验过程中收获的经验和教训 4](#_Toc3922841)

[6.2 针对以下方面的感受 4](#_Toc3922842)

# 实验目标概述

# 本次实验重点训练学生面向健壮性和正确性的编程技能，利用错误和异常处 理、断言与防御式编程技术、日志/断点等调试技术、黑盒测试编程技术，使程序 可在不同的健壮性/正确性需求下能恰当的处理各种例外与错误情况，在出错后 可优雅的退出或继续执行，发现错误之后可有效的定位错误并做出修改。 实验针对 Lab 3 中写好的 ADT 代码和基于该 ADT 的三个应用的代码，使用 以下技术进行改造，提高其健壮性和正确性：

# ⚫ 错误处理

# ⚫ 异常处理

# ⚫ Assertion 和防御式编程

# ⚫ 日志

# ⚫ 调试技术

# ⚫ 黑盒测试及代码覆盖度

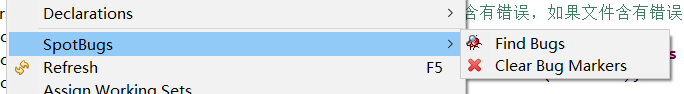
# 实验环境配置

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。

SpotBugs:

直接在Marketplace里面下载安装的：



在这里给出你的GitHub Lab4仓库的URL地址（Lab4-学号）。

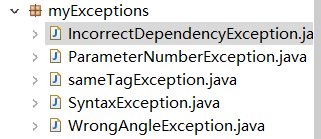
https://github.com/ComputerScienceHIT/Lab4-1170500704

# 实验过程

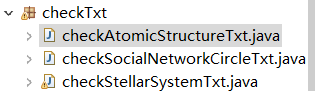
请仔细对照实验手册，针对每一项任务，在下面各节中记录你的实验过程、阐述你的设计思路和问题求解思路，可辅之以示意图或关键源代码加以说明（但千万不要把你的源代码全部粘贴过来！）。

## Error and Exception Handling

我首先自定义了一些异常类，写在myExceptions包里面：



写了检查输入txt文件的包checkTxt：



三个Java文件分别用来检测原子轨道的输入文件，社交轨道的输入文件，行星轨道的输入文件。

下面给出每个异常的功能：

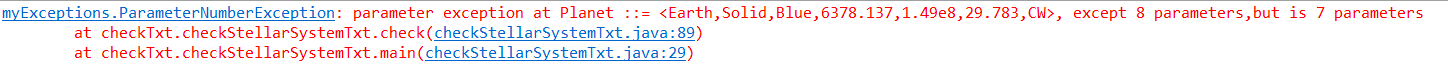
***IncorrectDependencyException：***

触发条件：依赖关系不正确

ParameterNumberException：

触发条件：txt文件传入的参数数量不正确

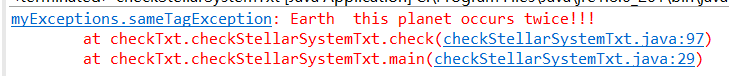
例如：行星的标准输入后面有8个参数，如果我现在给改成七个，那么运行checkStellarSystemTxt，Console将会输出：



***sameTagException***

触发条件：存在标签完全一样的元素

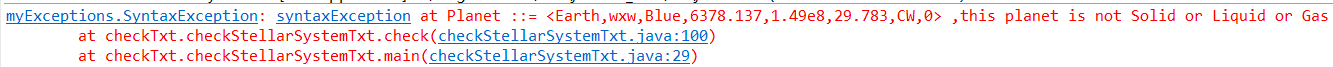
例如，如果我的StellarSystem.txt文件中有两个同名的行星，那么运行checkStellarSystemTxt，Console将会输出：



***SyntaxException:***

触发条件：语法不正确

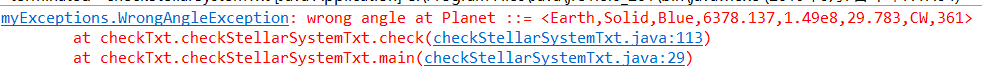
例如，行星的状态只有Solid,Liquid,Gas三个状态，如果我将行星的状态改成wxw,那么运行checkStellarSystemTxt，Console将会输出：



***WrongAngleException***

触发条件：角度越界，即角度不在0-360度这个范围

例如，如果我将行星的角度改成361,那么运行checkStellarSystemTxt，Console将会输出：



## Assertion and Defensive Programming

### checkRep()检查invariants

### Assertion保障pre-/post-condition

## Logging

### 写日志

### 日志查询

## 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

### 理解待调试程序的代码思想

### 发现并定位错误的过程

### 如何修正错误

### 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 |
| 5.6 | 19：00-20：30 | 学习异常处理 | 未完成 |
| 5.7 | 18：30-21：30 | 写StellarSystem.txt的异常处理 | 完成 |
|  |  |  |  |
|  |  |  |  |

# 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？如果你在完成Lab5的时候发现Lab5追求的是I/O大文件时的性能（时间/空间），你是否会回过头来修改你在Lab3和本实验里所做的各类defensive措施？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？
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%的闭卷考试有什么期望或建议？//请严肃的提出，杜绝开玩笑，教师会认真考虑你们的建议。