

**2018年春季学期  
计算机学院大二软件构造课程**

**Lab 4实验报告**

|  |  |
| --- | --- |
| 姓名 | 冯运 |
| 学号 | 1160300524 |
| 班号 | 1603005 |
| 电子邮件 | 294889365@qq.com |
| 手机号码 | 18846188605 |

**目录**

[1 实验目标概述 1](#_Toc511890111)

[2 实验环境配置 1](#_Toc511890112)

[3 实验过程 1](#_Toc511890113)

[3.1 Error and Exception Handling 1](#_Toc511890114)

[3.1.1 针对输入文本文件的异常/错误处理 1](#_Toc511890115)

[3.1.2 针对输入图操作指令的异常/错误处理（可选） 1](#_Toc511890116)

[3.2 Assertion and Defensive Programming 2](#_Toc511890117)

[3.2.1 checkRep()检查invariants 2](#_Toc511890118)

[3.2.2 Assertion保障pre-/post-condition 2](#_Toc511890119)

[3.3 Logging 2](#_Toc511890120)

[3.3.1 写日志 2](#_Toc511890121)

[3.3.2 日志查询 2](#_Toc511890122)

[3.4 Testing for Robustness and Correctness 2](#_Toc511890123)

[3.4.1 Testing strategy 2](#_Toc511890124)

[3.4.2 测试用例设计 2](#_Toc511890125)

[3.4.3 测试运行结果与覆盖度报告 2](#_Toc511890126)

[3.5 FindBugs tool（可选） 2](#_Toc511890127)

[3.6 Debugging 3](#_Toc511890128)

[3.6.1 待调试程序 3](#_Toc511890129)

[3.6.2 理解待调试程序的过程 3](#_Toc511890130)

[3.6.3 发现并定位错误的过程 3](#_Toc511890131)

[3.6.4 如何修正错误 3](#_Toc511890132)

[3.6.5 结果 3](#_Toc511890133)

[4 实验进度记录 3](#_Toc511890134)

[5 实验过程中遇到的困难与解决途径 3](#_Toc511890135)

[6 实验过程中收获的经验、教训、感想 3](#_Toc511890136)

# 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Lab3中写好的ADT（Graph<L,E>）代码和基于该ADT的四个应用（GraphPoet、社交网络、计算机网络拓扑、电影网络）的代码，使用以下技术进行改造，提高其健壮性和正确性：1、错误处理2、异常处理3、Assertion和防御式编程4、日志5、调试技术6、黑盒测试及代码覆盖度

# 实验环境配置

实验需要配置elclipse,但是我之前已经习惯于使用jetbrain系列的开发ide,所以我这次使用了IDEA，同时下载安装了jdk，版本号是jdk1.8.0\_101，将jdk的安装路径加入了系统的环境变量。在IDEA里面下载安装了junit插件,配置好了java的测试环境。

Lab4的仓库url:

https://github.com/ComputerScienceHIT/Lab4-1160300524.git

# 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

### 针对输入文本文件的异常/错误处理

1. 针对输入的内容不合法的问题，我使用正确的正则表达式去匹配相应的语句，如果正则表达式无法识别，则抛出一个自定义的FormatException异常，这个异常的信息里包括具体的错误类型，错误的行数，方便用户去重新修改文件格式。这个异常统一负责所有的格式性问题，没有做也没必要做特别详细的格式问题区分。
2. 边中使用的节点在节点部分未定义：这个问题我放到了向图中添加边时处理，在添加边时，检测如果边中的点还未在图中，则抛出EdgeNullVertexException异常，然后交给调用的函数去处理异常。
3. 为节点定义的属性的数目与特定应用的图的约束不符：

对于这个异常，我放在了向点中添加属性时处理，（在具体的Vertex的子类中），如果输入的属性的数量不正确，或者值不在合法的范围内，都统一抛出VertexAttributeException异常，然后交给调用的函数去处理。

1. 在某种类型的图应用中引入了不应出现的节点类型：这个异常我放在了向图中添加点时处理，在向图中添加点时，检测加入的点的类型是否符合当前种类图的要求，如果不符合，就抛出VertexTypeException，然后交给调用的函数去处理
2. 在某种类型的图应用中引入了不应出现的边类型：这个异常我放在了向图中添加边时处理，在向图中添加边时，检测加入的边的类型是否符合当前种类图的要求，如果不符合，就抛出EdgeTypeException，然后交给调用的函数去处理。
3. 在无向图中引入了有向边：这个异常我放在了读入文件时处理，如果边后面的描述是 Yes，但是边的类型却是无向边，则抛出一个UndirectedEdgeException异常，只向日志中输出警告的异常信息，自动将有向边转成无向边，然后继续读取下一条边，不会中断文件的读取。
4. 在有向图中引入了无向边：这个异常我放在了读入文件时处理，如果边后面的描述是No，但是边的类型却是有向边，则抛出一个DirectedEdgeException异常，交给调用者去处理异常。
5. 在单重图中存在了多重边：这个异常的处理方式同5有矛盾，因为在单重图中存在了多充边这种情况是异常5的子集，但是二者的异常处理方式却不一样，所以存在矛盾。我按照异常5的方式处理了。
6. 在不应存在边的两个节点之间存在了边：这个异常的处理我放在了创建边时判断，如果添加的点不应该出现在这种类型的边中，则会抛出一个ExceptionVertexTypeException，然后交给调用者去处理异常。
7. 在不应存在超边的图中加入了超边：这个处理同异常5一样，这个异常是异常5的子集，所以如果异常5处理正确，这个异常也会自然得到处理。
8. 在不应出现loop的图中出现了loop：这个异常我放在了创建边时处理，如果某种类型的边不应出现loop，在创建时就会抛出EdgeLoopException，然后交给调用者去处理异常。
9. 某超边中包含的节点数小于2：这个异常我放在了创建超边时处理，如果创建的边的顶点小于2，就抛出一个HyperEdgeException，然后交给调用者去处理。
10. 带权边却未能给出权值：这个异常我放在了解析文件时处理，如果读取到带权边，但是它提供的weight是-1，则抛出一个EdgeWeightException，然后交给调用者去处理。
11. 带权边的权值不符合应用要求：这个异常我放在了创建边时处理，检测如果边的权值不符合边的类型要求，则抛出一个EdgeWeightException，然后交给调用者去处理。
12. 图、节点、边的label不满足正则表达式（\w+）的要求：这个异常放在了读入文件的部分，即设置在了正则表达式中，如果label不符合要求，就不会被正则表达式匹配到，就会抛出FormatException，然后交给调用者去处理异常。
13. 多个节点的label重复、多个边的label重复：如果点的label重复，我会抛出异常，然后停止解析文件。但是对于边的label重复，我将自动在edge的label后加上从\_0开始的后缀，从而提高容错性。

### 针对输入图操作指令的异常/错误处理（可选）

覆盖实验手册3.1节中(2)列出的各项任务。

点的命令

vertex --add label type 向图中添加名为label，类型为type的点

vertex --delete regex 删除所有label符合regex规则的点

vertex --update label

* label= 改变点的label值
* argument= 改变点的argument值，即点的属性

vertex --show label

* eccentricity 求点的eccentricity值
* degree 求点的degree值
* indegree 求点的indegree值
* outdegree 求点的outdegree值
* closenessCentrality 求点的closenessCentrality值
* betweennessCentrality 求点的 betweennessCentrality值

边的命令

edge --add label type [weighted=Y|N] [weight] [directed=Y|N] v1, v2

edge --delete regex 删除所有label符合regex的边

超边的命令

hyperedge --add

* label type vertex(1), ..., vertex(n) 向图中添加一个新的超边
* label vertex(1), ..., vertex(n) 向label 为 label的超边添加点

hyperedge --delete label regex 删除超边中label符合regex规则的点

图的命令

graph --add filepath 通过输入一个固定格式的文件，根据文件中提供的信息建立一个新的图

graph --show

* degreeCentrality 输出图的degreeCentrality值
* radius 输出图的radius值
* diameter 输出图的diameter值
* visible 将图的可视化结果输出
* distance="vertex1""vertex2" 输出vertex1 与 vertex2 在图中的 distance值
  1. 如果用户输入的命令不合法，我会提醒命令的错误细节，然后给出完整的指令集，帮助用户正确的输入命令。由于我的命令解析是层层进行的，所以发现的错误也是层层抛出，最后在总的函数中集中处理命令输入不合法的问题。在解析中遇到的错误会抛出CommandException。会将异常信息分别输出到控制台和日志文件中，用以提醒用户命令输入错误，但是不会终止程序，而是可以让用户持续地输入命令，知道用户输入exit才会终止程序。
  2. 指令的处理和文件的读入部分使用的是相同的工厂函数，在文件读入过程中处理过的异常，在解析命令时是相同的处理方式。所以这条要求我可以在文件读入的基础上，轻易的完成。

## Assertion and Defensive Programming

### checkRep()检查invariants

所有的父类的checkRep()是所有子类的checkRep()的交集

vertex.Vertex:

label不为null；

vertex.NetworkVertex（Server, Computer, Router的父类）:

ip地址符合规定的格式；

vertex.MovieProducer（Actor和Director的父类）:

(1) age大于0；

(2) gender是”M”或”F”；

vertex.Person:

(1) age大于0；

(2) gender是”M”或”F”；

vertex.Movie:

(1) 上映年份介于1900到2018之间；

(2) IMDb评分介于0到10之间；

edge.Edge:

label不为null；

edge.SimpleEdge（DirectedEdge和UndirectedEdge的父类）:

所有节点都不为null；

edge.HyperEdge:

(1) 包含的节点数大于1；

(2) 节点无重复；

edge. MovieActorRelation:

权重为正数；

edge.NetworkConnection:

(1) 权重为正数；

(2) 不存在环；

edge. SocialNetworkConnection(FriendTie, CommentTie, ForwardTie的父类):

(1) 权重介于0到1之间；

(2) 不存在环；

edge. WordNeighborhood:

权重为正数；

graph.ConcreteGraph:

(1) label不为null；

(2) 没有重复的边；

(3) 没有重复的点；

graph.SocialNetwork:

若边集不为空则所有边的权值之和为1；

### Assertion保障pre-/post-condition

1、我在对传入参数有特殊要求的函数入口处，对参数进行了合法性检验，并且在所有可能出现非法点或非法边的地方调用了这些方法进行检查，例如addEdge()和addVertex()；

2. 在SocialNetworkConnection和NetworkConnection的构造函数中添加了对环的检查

3. 在Vertex、Edge和ConcreteGraph的构造函数中添加label为null的检查，

4. 在Vertex的实现了fillVertexInfo()方法的子类的fillVertexInfo()方法中添加了对属性值的检查

5. 在Vertex的实现了fillVertexInfo()方法的子类的fillVertexInfo()方法中添加了对属性列表元素个数的检查，若元素个数与该类待填写的属性个数不同

6. 在边的工厂函数中添加了对传入的节点列表的节点类型的检查，若节点不应该出现在该类型的边中

7. 在边的工厂函数中添加了对传入的节点列表的节点个数的检查，如有向边和无向边的节点个数应该为2，超边的节点个数应该大于2

## Logging

### 写日志

我使用了java.uitl中的logger。

然后我对logging进行了简单的封装，创建了一个LoggerFactory的包，包中包含了两个类，

一个是工厂类，在使用时只需要提供logger的 name和日志文件的路径，即可获得一个Logger对象。

另一个类是对logger中的info，warining,severe三种不同程度的日志等级的封装，在具体的需要输出日志的位置，只需要直接调用info(),warning(),severe()方法即可。

然后我把建图过程中抛出的异常，用户输入的命令，以及命令的解析结果都输出到了日志中，方便查看程序的执行情况。

### 日志查询

## Testing for Robustness and Correctness

### Testing strategy

测试分两部分，一部分是测试正确性，一部分是测试健壮性。在测试正确性时，我根据规约的要求，构建了正常的函数输入，然后检测返回值是否正确。

测试健壮性的部分，我自己想了一些可能的非法输入，然后测试程序是否相应的抛出了异常。

如果程序抛出了想要的异常则测试通过，如果程序未抛出异常或抛出的别的类型的异常，则通过设置断言的方式使测试失败。

### 测试用例设计

由于测试的类和方法过多，不适宜在报告中罗列，详情请见项目中的test文件夹。

### 测试运行结果与覆盖度报告

## FindBugs tool（可选）

发现了哪些错误，每种错误代表什么不良的编程习惯

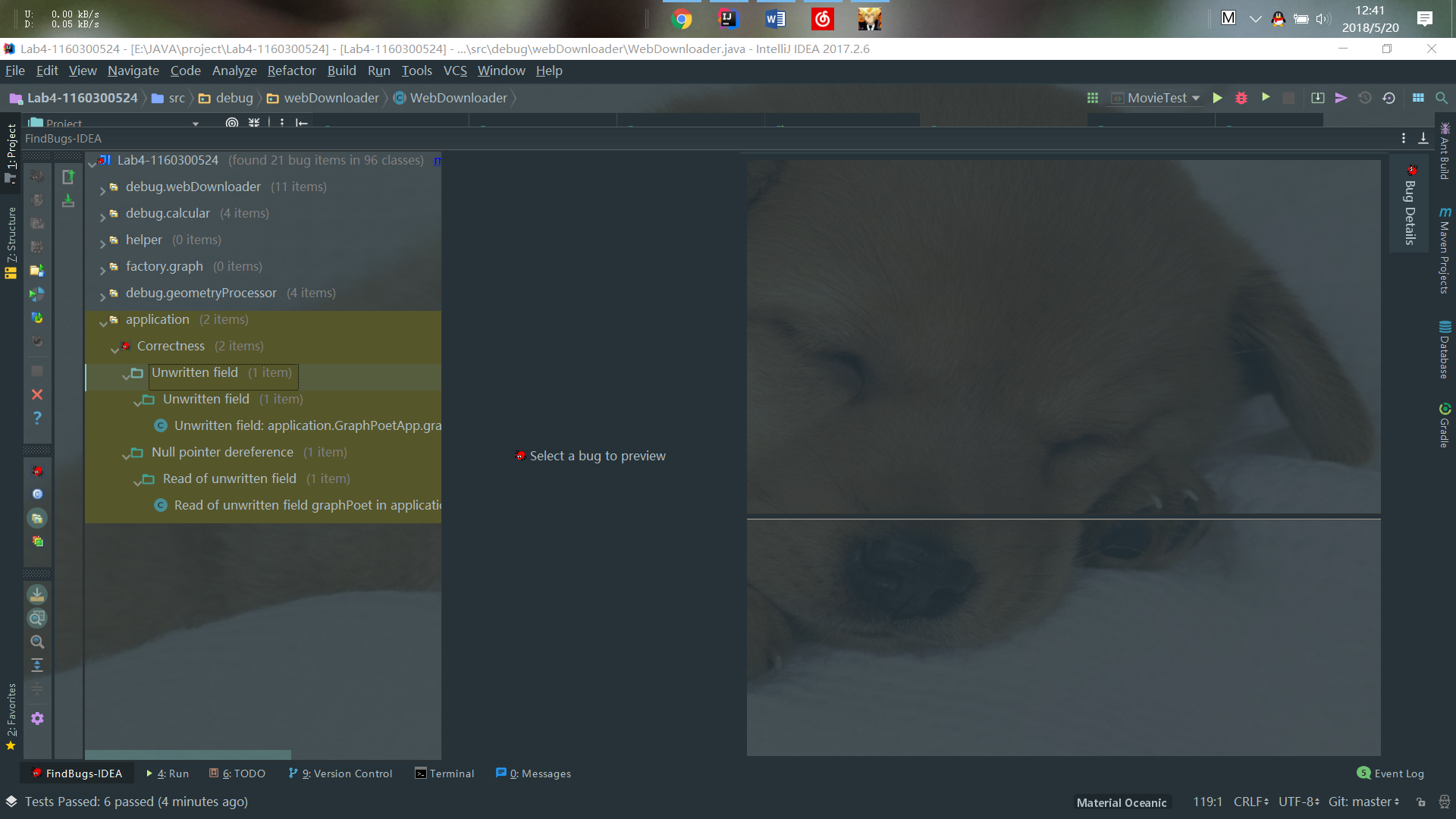
对代码修改，消除这些错误。

通过find bugs 找出过几种比较经典的可能bug

1. 命名不完全符合java的驼峰命名规范，比如CommandHelper，按照驼峰的命名规范，应该是commandHelper，这种规范问题我以后会在编程中避免的。
2. 有些可能返回null的函数，我未对返回值检查null,这样可能会引起NullPointerException的异常，以后编程时需要注意及时判断返回值是否为null
3. 在判断浮点数是否相等时，直接使用==，这样的写法有安全隐患，因为如果浮点数的存储是近似的，如果两个相等的数直接==比较会因误差而出现不想等的结果。

之后我逐项消除了这些bug，得到了如下的运行结果。

虽然显示的是21个bugs，但是有19个是debug文件引入的，不是我自己项目的。剩余的2个可能bug，经过自己的分析，自己的代码没有问题。



## Debugging

### 待调试程序

1. 计算器的程序
2. Circle,trangle,square
3. WebDownloader

### 理解待调试程序的过程

1. 计算器是一个GUI程序，然后给出了正确的GUI界面，然后我根据要求的界面来看错误程序的界面，发现有按键的键值是错误的。然后我在代码中查找关于setText函数的部分，发现是for循环过程中的下标设置错误，通过修改下标的值，就得到了正确的运行结果。
2. 首先查看了给出程序的主程序，大致了解了程序的大体结构，然后我查看了具体的shape接口，了解了它的三个子类的功能，这个整个程序的结构和功能就都了解了。
3. 这个程序我也是先从main函数看起，然后大致了解了它的功能，之后我从WebDownloader抽象类查看具体的实现，这个整个程序的结构和功能就都了解了。

### 发现并定位错误的过程

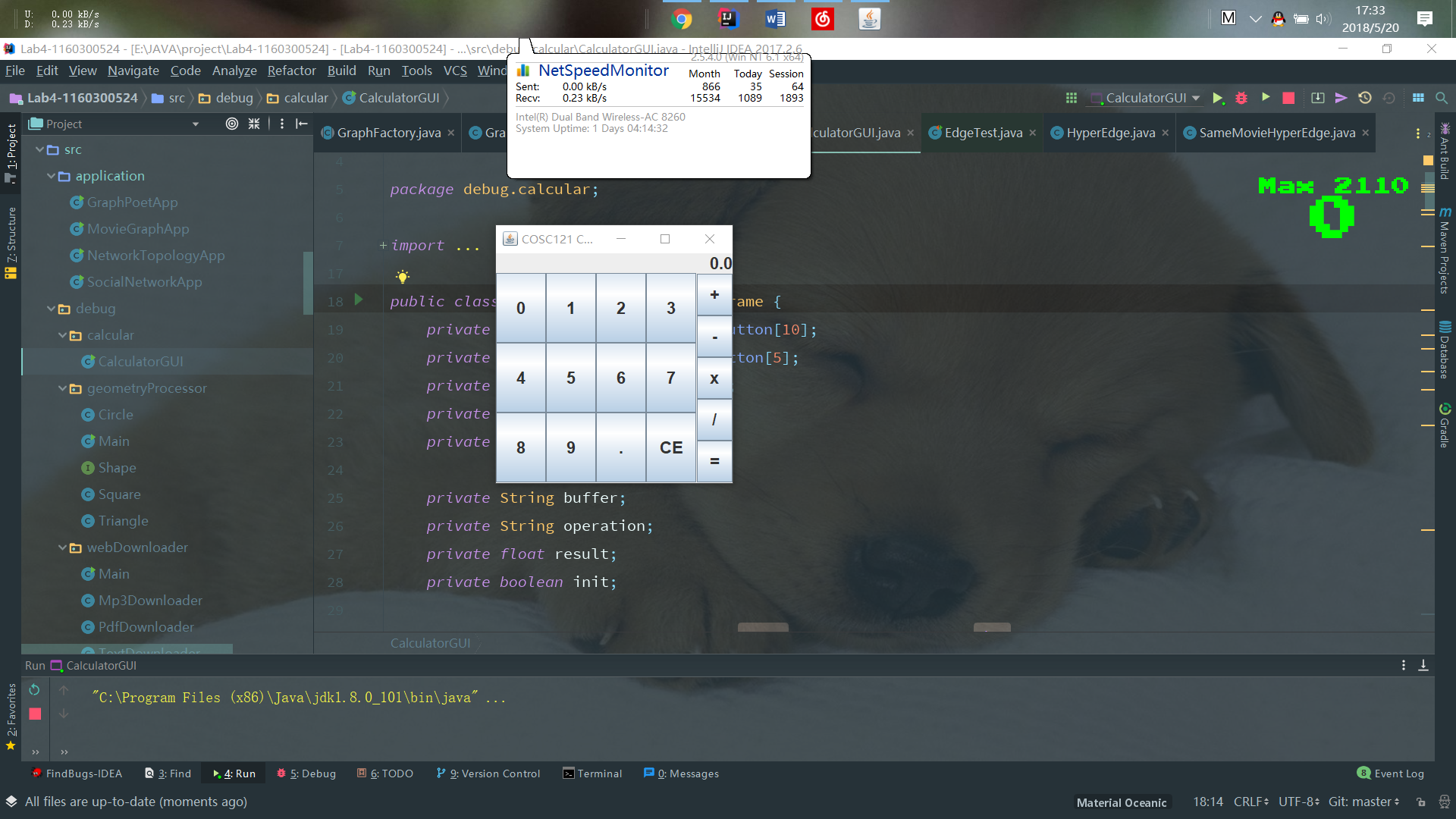
* 1. 然后我根据要求的界面来看错误程序的界面，发现有按键的键值是错误的。然后我在代码中查找关于setText函数的部分，发现是for循环过程中的下标设置错误，通过修改下标的值，就得到了正确的运行结果。
  2. 我发现在编译时就报了一些错误，发现 是它尝试实例化一个接口，这是一个简单的错误，之后又发现name输出的是颜色，然后我通过设置断点，定位到了具体的错误位置：getName()函数
  3. 发现在运行程序时，在我做了选择后，没有具体的功能执行，这个错误是在main函数中，然后发现下载的列表中.pdf变成了.pd，这个错误通过设置断点，定位到了webDownloader中。

### 如何修正错误

* + 1. 通过修改for循环的下标的值，就得到了正确的运行结果。
    2. 通过将实例化接口的地方改成实例化具体的实现类，就可以正常编译了，然后将getName()函数的return color;改成return name； 就修复了bug。
    3. 通过调整main函数中if条件判断的与改成或，即修复了bug，然后在webDownloader中，我将它获取文件名的部分，它在截取字符串的end位置+1，就消除了bug.

### 结果

1、运行结果



结果2

Printing all shapes grouped by array

Shape: Circle, Name: Circle00, Area: 163.5813784948879, Colour: Red

Shape: Square, Name: Square01, Area: 97.63381540744761, Colour: Green

Shape: Triangle, Name: Triangle02, Area: 57.76253814862329, Colour: Blue

Shape: Circle, Name: Circle03, Area: 279.01143541172485, Colour: Red

Shape: Square, Name: Square04, Area: 42.65156952538357, Colour: Green

Shape: Triangle, Name: Triangle05, Area: 102.5470462508207, Colour: Blue

Shape: Circle, Name: Circle06, Area: 12.116518408560234, Colour: Red

Shape: Square, Name: Square07, Area: 72.11477097979203, Colour: Green

Shape: Triangle, Name: Triangle08, Area: 15.334000001850036, Colour: Blue

Shape: Circle, Name: Circle09, Area: 50.504761623566246, Colour: Red

Shape: Square, Name: Square010, Area: 33.69147436278792, Colour: Green

Shape: Circle, Name: Circle10, Area: 253.35963776780693, Colour: Red

Shape: Square, Name: Square11, Area: 26.745870137221097, Colour: Green

Shape: Triangle, Name: Triangle12, Area: 20.426210552790486, Colour: Blue

Shape: Circle, Name: Circle13, Area: 179.0890458115951, Colour: Red

Shape: Square, Name: Square14, Area: 13.321339870418472, Colour: Green

Shape: Triangle, Name: Triangle15, Area: 96.11919396283737, Colour: Blue

Shape: Circle, Name: Circle16, Area: 2.293587851210051, Colour: Red

Shape: Square, Name: Square17, Area: 72.82835362510758, Colour: Green

Shape: Triangle, Name: Triangle18, Area: 118.28170181494372, Colour: Blue

Shape: Circle, Name: Circle19, Area: 1.7881610921603879, Colour: Red

Shape: Square, Name: Square110, Area: 3.713970990779881, Colour: Green

Shape: Circle, Name: Circle20, Area: 289.80858645678984, Colour: Red

Shape: Square, Name: Square21, Area: 32.22597213206574, Colour: Green

Shape: Triangle, Name: Triangle22, Area: 170.63892384003455, Colour: Blue

Shape: Circle, Name: Circle23, Area: 158.43261691717288, Colour: Red

Shape: Square, Name: Square24, Area: 40.64267702581195, Colour: Green

Shape: Triangle, Name: Triangle25, Area: 252.05431466043314, Colour: Blue

Shape: Circle, Name: Circle26, Area: 206.06277729057604, Colour: Red

Shape: Square, Name: Square27, Area: 4.65004415852795, Colour: Green

Shape: Triangle, Name: Triangle28, Area: 2.24876212498546, Colour: Blue

Shape: Circle, Name: Circle29, Area: 53.300481078784486, Colour: Red

Shape: Square, Name: Square210, Area: 0.11741687752994026, Colour: Green

Printing out shapes grouped by type...

Shape: Circle, Name: Circle00, Area: 163.5813784948879, Colour: Red

Shape: Circle, Name: Circle03, Area: 279.01143541172485, Colour: Red

Shape: Circle, Name: Circle06, Area: 12.116518408560234, Colour: Red

Shape: Circle, Name: Circle09, Area: 50.504761623566246, Colour: Red

Shape: Circle, Name: Circle10, Area: 253.35963776780693, Colour: Red

Shape: Circle, Name: Circle13, Area: 179.0890458115951, Colour: Red

Shape: Circle, Name: Circle16, Area: 2.293587851210051, Colour: Red

Shape: Circle, Name: Circle19, Area: 1.7881610921603879, Colour: Red

Shape: Circle, Name: Circle20, Area: 289.80858645678984, Colour: Red

Shape: Circle, Name: Circle23, Area: 158.43261691717288, Colour: Red

Shape: Circle, Name: Circle26, Area: 206.06277729057604, Colour: Red

Shape: Circle, Name: Circle29, Area: 53.300481078784486, Colour: Red

Shape: Square, Name: Square01, Area: 97.63381540744761, Colour: Green

Shape: Square, Name: Square04, Area: 42.65156952538357, Colour: Green

Shape: Square, Name: Square07, Area: 72.11477097979203, Colour: Green

Shape: Square, Name: Square010, Area: 33.69147436278792, Colour: Green

Shape: Square, Name: Square11, Area: 26.745870137221097, Colour: Green

Shape: Square, Name: Square14, Area: 13.321339870418472, Colour: Green

Shape: Square, Name: Square17, Area: 72.82835362510758, Colour: Green

Shape: Square, Name: Square110, Area: 3.713970990779881, Colour: Green

Shape: Square, Name: Square21, Area: 32.22597213206574, Colour: Green

Shape: Square, Name: Square24, Area: 40.64267702581195, Colour: Green

Shape: Square, Name: Square27, Area: 4.65004415852795, Colour: Green

Shape: Square, Name: Square210, Area: 0.11741687752994026, Colour: Green

Shape: Triangle, Name: Triangle02, Area: 57.76253814862329, Colour: Blue

Shape: Triangle, Name: Triangle05, Area: 102.5470462508207, Colour: Blue

Shape: Triangle, Name: Triangle08, Area: 15.334000001850036, Colour: Blue

Shape: Triangle, Name: Triangle12, Area: 20.426210552790486, Colour: Blue

Shape: Triangle, Name: Triangle15, Area: 96.11919396283737, Colour: Blue

Shape: Triangle, Name: Triangle18, Area: 118.28170181494372, Colour: Blue

Shape: Triangle, Name: Triangle22, Area: 170.63892384003455, Colour: Blue

Shape: Triangle, Name: Triangle25, Area: 252.05431466043314, Colour: Blue

Shape: Triangle, Name: Triangle28, Area: 2.24876212498546, Colour: Blue

程序四

Enter your desired file destination: test

Choose a file type to download:

1).pdf

2).mp3

3).txt

Choice:

1

test

Print files before downloading?

1) Yes

2) No

1

304\_All\_Notes\_by6.pdf

304\_1\_Intro\_by6.pdf

304\_1\_Intro.pdf

304\_2\_DBIntro\_by6.pdf

304\_2\_DBIntro.pdf

304\_3\_Relational\_by6.pdf

304\_3\_Relational.pdf

304\_4\_SQL\_DDL\_by6.pdf

304\_4\_SQL\_DDL.pdf

304\_5\_SQL\_by6.pdf

304\_5\_SQL.pdf

304\_5\_SQL\_answers.pdf

304\_6\_DBDesign\_by6.pdf

304\_6\_DBDesign.pdf

304\_7\_ER\_by6.pdf

304\_7\_ER.pdf

304\_7\_ER\_by6.pdf

304\_7\_ER.pdf

304\_8\_EER\_by6.pdf

304\_8\_EER.pdf

304\_9\_ER\_to\_Relational\_by6.pdf

304\_9\_ER\_to\_Relational.pdf

304\_10\_DBProgramming\_by6.pdf

304\_10\_DBProgramming.pdf

304\_11\_WWWProgramming\_by6.pdf

304\_11\_WWWProgramming.pdf

304\_12\_Normalization\_by6.pdf

304\_12\_Normalization.pdf

304\_13\_JSON\_by6.pdf

304\_13\_JSON.pdf

304\_14\_NoSQL\_by6.pdf

304\_14\_NoSQL.pdf

304\_15\_Triggers\_by6.pdf

304\_15\_Triggers.pdf

304\_16\_Views\_by6.pdf

304\_16\_Views.pdf

304\_17\_AdvancedSQL\_by6.pdf

304\_17\_AdvancedSQL.pdf

304\_18\_XML\_by6.pdf

304\_18\_XML.pdf

304\_19\_XMLQuery\_by6.pdf

304\_19\_XMLQuery.pdf

304\_20\_DataWarehouse\_by6.pdf

304\_20\_DataWarehouse.pdf

304\_21\_Course\_Summary\_by6.pdf

304\_21\_Course\_Summary.pdf

refs/rulesfordatastorage.pdf

total files found: 47

Download files?

1) Yes

2) No

Execution time = 60seconds

47 file(s) downloaded

Program exiting, goodbye!

# 实验进度记录

请尽可能详细的记录你的进度情况。

|  |  |  |  |
| --- | --- | --- | --- |
| 日期 | 时间段 | 计划任务 | 实际完成情况 |
| 5.15 | --- | 完成异常处理 | 完成 |
| 5.17 | ---- | 完成日志记录 | 完成 |
| 5.20 | ---- | 完成所有的内容 | 完成 |

# 实验过程中遇到的困难与解决途径

# 实验过程中收获的经验、教训、感想

本节除了总结你在实验过程中收获的经验和教训，也可就以下方面谈谈你的感受（非必须）：

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？
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？
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. 关于本实验的工作量、难度、deadline。
8. 到目前为止你对《软件构造》课程的评价和建议。

实验让我写的要猝死。。。