

**2018年春季学期  
计算机学院大二软件构造课程**

**Lab 1实验报告**

|  |  |
| --- | --- |
| 姓名 | 郭潇 |
| 学号 | 1160300521 |
| 班号 | 1603005 |
| 电子邮件 | wqsnzszw@163.com |
| 手机号码 | 18523735573 |

**目录**

[1 实验目标概述 1](#_Toc509142307)

[2 实验环境配置 1](#_Toc509142308)

[3 实验过程 1](#_Toc509142309)

[3.1 Magic Squares 1](#_Toc509142310)

[3.1.1 isLegalMagicSquare() 1](#_Toc509142311)

[3.1.2 generateMagicSquare() 1](#_Toc509142312)

[3.2 Turtle Graphics 1](#_Toc509142313)

[3.2.1 Problem 1: Clone and import 1](#_Toc509142314)

[3.2.2 Problem 3: Turtle graphics and drawSquare 1](#_Toc509142315)

[3.2.3 Problem 5: Drawing polygons 1](#_Toc509142316)

[3.2.4 Problem 6: Calculating headings 1](#_Toc509142317)

[3.2.5 Problem 7: Personal art 1](#_Toc509142318)

[3.2.6 Submitting 1](#_Toc509142319)

[3.3 Social Network 1](#_Toc509142320)

[3.3.1 设计/实现FriendshipGraph类 1](#_Toc509142321)

[3.3.2 设计/实现Person类 1](#_Toc509142322)

[3.3.3 设计/实现客户端代码main() 1](#_Toc509142323)

[3.3.4 设计/实现测试用例 1](#_Toc509142324)

[3.4 Tweet Tweet（选作，额外记分） 1](#_Toc509142325)

[3.4.1 设计/实现getTimespan()函数 1](#_Toc509142326)

[3.4.2 设计/实现getMentionedUsers函数 1](#_Toc509142327)

[3.4.3 设计/实现writtenBy函数 1](#_Toc509142328)

[3.4.4 设计/实现inTimespan函数 1](#_Toc509142329)

[3.4.5 设计/实现containing函数 1](#_Toc509142330)

[3.4.6 设计/实现guessFollowsGraph函数 1](#_Toc509142331)

[3.4.7 设计/实现influencers函数 1](#_Toc509142332)

[3.4.8 设计/实现getHashtags函数 1](#_Toc509142333)

[3.4.9 Extract的测试用例 1](#_Toc509142334)

[3.4.10 Filter的测试用例 1](#_Toc509142335)

[3.4.11 SocialNetwork的测试用例 1](#_Toc509142336)

[3.4.12 MySocialNetwork的测试用例 1](#_Toc509142337)

[4 实验进度记录 1](#_Toc509142338)

[5 实验过程中遇到的困难与解决途径 1](#_Toc509142339)

[6 实验过程中收获的经验、教训、感想 1](#_Toc509142340)

# 实验目标概述

本次实验通过求解四个问题（其中一个可选），训练基本Java编程技能，能够利用Java OO开发基本的功能模块，能够阅读理解已有代码框架并根据功能需求补全代码，能够为所开发的代码编写基本的测试程序并完成测试，初步保证所开发代码的正确性。另一方面，利用Git作为代码配置管理的工具，学会Git的基本使用方法。

基本的Java OO编程

基于Eclipse IDE进行Java编程

基于JUnit的测试

基于Git的代码配置管理

# 实验环境配置

依次从官网下载安装jdk，eclipse，git。在使用git创建仓库时，对git的基本指令不熟悉。在网上查阅教程后才完成相关配置。

在这里给出你的GitHub Lab1仓库的URL地址（Lab1-学号）。

https://github.com/ComputerScienceHIT/Lab1-1160300521

# 实验过程

请仔细对照实验手册，针对四个问题中的每一项任务，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！）。

为了条理清晰，可根据需要在各节增加三级标题。

## Magic Squares

1. 利用isLegalMagicSquare()函数读取文件，判断文件是否符合格式，是否为magicsquare。
2. 利用generateMagicSquare()根据输入的数字生成相应的magicsquare并将其写入文件。

### isLegalMagicSquare()

过程：

1. 读入文件，利用split()函数将其按照“\t”划分并转化为int类型
2. 利用正则表达式判断读入的举证是否符合格式
3. 若读入的文件符合矩阵格式，从列、行、对角线的值是否相等判断其是否为magicsquare

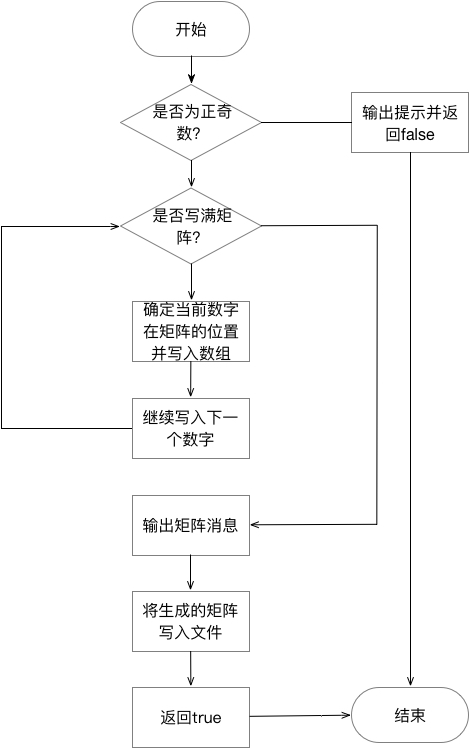
结果：

1. 若读入的矩阵不符合给定的格式，则返回false
2. 若给定的矩阵的各列、行、对角线的值相等，则返回true，否则返回false

### generateMagicSquare()

过程：

1. 判断输入的数是否为正的奇数
2. 根据输入的数字生成相应的magicsquare
3. 将生成的矩阵写入文件



结果：

1. 若输入的数不是正奇数，则返回false
2. 生产的矩阵写入文件后返回true

## Turtle Graphics

1. 利用forward()和turn()函数完成drawSquare()
2. 利用正多边形边和角度之间的关系完成problem5中的函数并通过测试
3. 利用坐标中两点之间角度关系完成problem6中函数

### Problem 1: Clone and import

如何从GitHub获取该任务的代码、在本地创建git仓库、使用git管理本地开发。

1. 使用git clone 从提供网址获取实验代码
2. 在eclipse中import实验项目
3. 安装git，利用git init创建仓库
4. 利用git add，git commit 提交文件，git status 查看需要提交的文件

### Problem 3: Turtle graphics and drawSquare

过程：

1. 利用forward函数前进相应的步数
2. 利用turn函数旋转90度
3. 重复上述过程绘制正方形

### Problem 5: Drawing polygons

过程：

1. 利用正多边形边与角度的关系确定各个内角的大小
2. 前进适当的步数后旋转相应角度
3. 循环上述过程绘制正多边形

### Problem 6: Calculating headings

过程：

1. 根据坐标中两点间角度的关系完成calculateHeadingToPoint()函数
2. 从x坐标和y坐标的list中选取对应的点调用calculateHeadingToPoint()函数
3. 将输出的结果写入结果的list中

结果：

返回改变角度的list

### Problem 7: Personal art

调用color()函数改变颜色，通过循环和drawRegularPolygon()turtle.turn()函数生成图案

### Submitting

1 git add 将修改添加至本地缓存

2.git commit -m 'msg' 将本地缓存保存到本地仓库中

3.git push 将本地仓库推送至服务器.

## Social Network

1. 在friendgraph中加入新的人相当于加入新的点，加入新的朋友关系相当于加入新的边
2. 从一个顶点利用广度搜索，最先搜索到制定是朋友所用的距离则为两人之间最短路径，若搜索完仍未找到指定朋友则返回-1
3. 在调用相应函数时，应注意null，重复和相同的问题并抛出相应异常

### 设计/实现FriendshipGraph类

设计：

1. Friendgraph为一个适当大小的类型为person的数组，第一列为各个顶点，每一行对应第一个元素的朋友
2. AddVertex()函数加入新的顶点，addEdge()加入新的边
3. Getdistence()函数调用solve()进行广度搜索，返回最短距离
4. 利用isingraph确定重复名字，isin确定某点是否被搜索

结果：

1. 若加入点和边时存在person为null、重复和已存在的情况，则抛出异常
2. getdistance确定两个perosn间距离是否合法，不合法则抛出异常，合法则调用solve函数返回最短距离
3. isgraph和isin分别判断名字重复和是否已被访问，若重复、已被访问则返回true；否则返回false

### 设计/实现Person类

设计：

将person的name设为priva，利用preosn的构造函数对其进行初始化，利用getname方法改变name

### 设计/实现客户端代码main()

设计：按照实验要求，将无向图扩展为有向图，并添加题目中的相应操作。

结果：与题目中预期结果一致

### 设计/实现测试用例

设计：

1. 测试加入一个为null的perosn
2. 测试加入相同名字的perosn
3. 测试加入一条含有null perosn 的边
4. 测试加入一条含有相同 perosn 的边
5. 测试加入一条含有重复的边
6. 测试实验的简单例子

结果：

1. 对1到6的测试抛出异常
2. 对实验给出的例子返回相同结果

## Tweet Tweet（选作，额外记分）

### 设计/实现getTimespan()函数

过程：

1. 判断tweets是否为空
2. 对输入tweets的时间进行排序
3. 将排序后的第一个时间作为start，最后一个作为end，生成相应的timespan

结果：

1. 若tweets为空则返回当前时间点
2. 返回tweets包含的timespan

### 设计/实现getMentionedUsers函数

过程：

1. 遍历每一条tweet，利用split()拆分每一个单词
2. 利用正则表达式判断是否有符合条件的作者名
3. 将符合条件的作者名改为小写加入到存储结果的set中

结果：  
返回存储符合题意作者名的set

### 设计/实现writtenBy函数

过程：

1. 遍历每一条tweet，获取其作者名并转化为小写
2. 将作者名与给定的作者进行比较
3. 若有给定作者的tweet，则加入到存储结果的list中

结果：

返回存储相应作者tweet的list

### 设计/实现inTimespan函数

过程：

1. 遍历每一条tweet，获取其timespan
2. 判断当前tweet的timesoan的start是否大于标准timesoan的start，end是否小于标准timesoan的end
3. 若符合条件则加入到存储tweet的list中

结果：

返回存储相应timespan的tweet的list

### 设计/实现containing函数

过程：

1.遍历每一条tweet，利用split()拆分每一个单词并转化为小写

2.判断tweet中是否含有相应的单词

3.将符合条件的tweet存储结果的list中

结果：  
返回存储含有相应单词tweet的list

### 设计/实现guessFollowsGraph函数

过程：

1. 遍历输入存有tweet的List，保存其提到的其他作者名
2. 将除了作者@自己的tweet除去，其他以作者名为key，tweet为value存储到map中
3. 调用getHashtags函数获取tweet中的关键词，以作者名为key，关键词为value存储到map
4. 寻找具有相同关键词的作者，互相添加到map中

结果：

返回一个存有作者名和tweet的map

### 设计/实现influencers函数

过程：

1. 遍历fllowGraph。计算每个被关注的作者的权值并保存到map中
2. 将没个未被关注的作者权值设为0
3. 对关注作者的权值进行排序

结果：

返回存有按关注人数排序的作者名的List

### 设计/实现getHashtags函数

过程：

1. 将输入的tweet内容按单词拆分
2. 利用正则表达式判断关键字将其添加到set中

结果：

返回当前tweet中所有关键字的set

### Extract的测试用例

1. 一条tweet的开始时间和结束时间相同
2. 一条tweet的开始时间和结束时间不相同
3. 测试一条空的tweet
4. 两条tweet的开始时间和结束时间不相同
5. 一条没有提到他人的tweet
6. 一条提到一个其他人的tweet
7. 一条提到两个其他人的tweet
8. 两条提到两个其他人的tweet
9. 一条含有邮箱格式的tweet

### Filter的测试用例

1. 一条含有指定作者的tweet
2. 一条不含有指定作者的tweet
3. 两条含有相同指定作者的tweet
4. 两条含有不相同指定作者的tweet
5. 测试两个时间都在指定timespan中
6. 测试两个时间都不在指定timespan中
7. 测试两个时间只有一个在指定timespan中
8. 一条含有指定单词的tweet
9. 一条不含有指定单词的tweet
10. 多条含有不相同指定单词的tweet

### SocialNetwork的测试用例

1. 测试一个空followgraph的影响关系
2. 测试一个followgraph没有相互影响
3. 测试一个followgraph有相互影响
4. 测试一个followgraph没有关注
5. 测试一个followgraph有一对关注
6. 测试一个followgraph有两队关注

### MySocialNetwork的测试用例

1. 测试一个followgraph有一个关键词
2. 测试一个followgraph有两个关键词

# 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 |
| 2018-02-28 | 14:00-15:30 | 配置实验环境，了解实验任务和git的基本操作 | 按计划完成 |
| 2018-03-01 | 18:00-21:00 | 完成实验1.1Magicsquare | 遇到困难，未完成 |
| 2018-03-02 | 15:30-17:30 | 完成实验1.1Magicsquare | 按时完成 |
| 2018-03-02 | 18:00-21:00 | 完成实验1.2 | 按时完成 |
| 2018-03-03 | 14:00-17:00 | 完成实验1.3的实现部分 | 按时完成 |
| 2018-03-04 | 15:00-17:00 | 完成实验1.3的测试部分 | 遇到困难，未完成 |
| 2018-03-05 | 14:00-15:30 | 完成实验1.3 | 按时完成 |
| 2018-03-05 | 18:00-21:00 | 完成实验1.4的problem1和2 | 按时完成 |
| 2018-03-06 | 15:00-17:30 | 完成实验1.4的problem3 | 按时完成 |
| 2018-03-07 | 18:30-21:00 | 完成实验1.4的problem4 | 按时完成 |
| 2018-03-08 | 13:00-17:00 | 完成实验1.4的测试部分 | 按时完成 |

# 实验过程中遇到的困难与解决途径

1. 困难：在is Magicsquare函数中，对读取文件的操作不熟悉，无法判断读入文件是否符合给定的格式

解决：学习读写文件的操作，学习正则表达式的相关内容，利用正则表达式判断文件的格式

1. 困难：在完成1.2的过程中，由于英语阅读能力没有完全理解实验要求，浪费太多时间

解决：借助工具和同学交流，了解实验的具体要求和细节

1. 困难：在编写实验1.3和1.4的测试部分时，对junit不熟悉，不知道如何测试抛出异常而浪费太多时间

解决：学习junit的先关只是，特别是检测抛出异常的方法

# 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. 在本次实验中熟悉了之前不常使用的正则表达式、map、文件读写操作
2. 知道了git的基本操作，但是对其的实现和机构并不了解，希望能通过今后的学习掌握相关知识。