

**2019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 | 杜钰欣 |
| 学号 | 1170300320 |
| 班号 | 1703003 |
| 电子邮件 | 18745718808@163.com |
| 手机号码 | 18745718808 |

**目录**

[1 实验目标概述 1](#_Toc506282558)

[2 实验环境配置 1](#_Toc506282559)

[3 实验过程 1](#_Toc506282560)

[3.1 Magic Squares 1](#_Toc506282561)

[3.1.1 isLegalMagicSquare() 1](#_Toc506282562)

[3.1.2 generateMagicSquare() 1](#_Toc506282563)

[3.2 Turtle Graphics 1](#_Toc506282564)

[3.2.1 Problem 1: Clone and import 2](#_Toc506282565)

[3.2.2 Problem 3: Turtle graphics and drawSquare 2](#_Toc506282566)

[3.2.3 Problem 5: Drawing polygons 2](#_Toc506282567)

[3.2.4 Problem 6: Calculating headings 2](#_Toc506282568)

[3.2.5 Problem 7: Personal art 2](#_Toc506282569)

[3.2.6 Submitting 2](#_Toc506282570)

[3.3 Social Network 2](#_Toc506282571)

[3.3.1 设计/实现FriendshipGraph类 2](#_Toc506282572)

[3.3.2 设计/实现Person类 2](#_Toc506282573)

[3.3.3 设计/实现客户端代码main() 2](#_Toc506282574)

[3.3.4 设计/实现测试用例 2](#_Toc506282575)

[3.4 Tweet Tweet（选作，额外记分） 3](#_Toc506282576)

[4 实验进度记录 3](#_Toc506282577)

[5 实验过程中遇到的困难与解决途径 3](#_Toc506282578)

[6 实验过程中收获的经验、教训、感想 3](#_Toc506282579)

# 实验目标概述

本次实验通过求解四个问题，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。

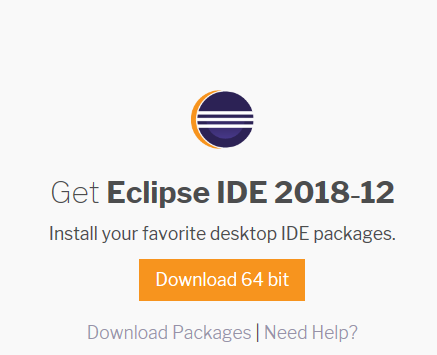
* 基本的Java OO编程
* 基于Eclipse IDE进行Java编程
* 基于JUnit的测试
* 基于Git的代码配置管理

# 实验环境配置

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。

安装eclipse



安装JDK 11.0.2



困难：因为电脑之前曾经安装过JDK的其他版本在安装eclipse的时候因为

其不知道用哪个版本的环境所以一直无法正常安装

解决：最后我重新格式化了电脑，将所有文件备份后重新安装系统再安装

安装github

注册github账号

在这里给出你的GitHub Lab1仓库的URL地址（Lab1-学号）。

Lab1-1170300320 https://github.com/ComputerScienceHIT/Lab1-1170300320.git

# 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

该实验基于一个有特殊性质的方阵，他的横向纵向和对角线方向的加和都是相同的，第一部分要求能够判断给出的例子是否是一个合法的幻方，它需要满足最基本的格式要求，每个幻方中出现的数字都必须是整型数字，分隔需要用制表符来完成，并且满足加和性质的要求，第二部分给出了生成幻方的代码，但是有一点错误，即在输入偶数和负数的时候，只需要加上判断条件即可

### isLegalMagicSquare()

设计：首先读取给出文本中的幻方内容，然后判断格式要求是否满足，最后判断加和是否正确

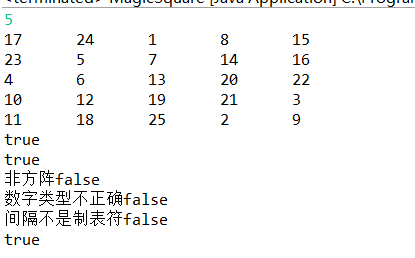
实现思路：读取：BufferedReader,FileReader

判断：.contains判断是否含空格间接判断制表符格式

分割：.split方法

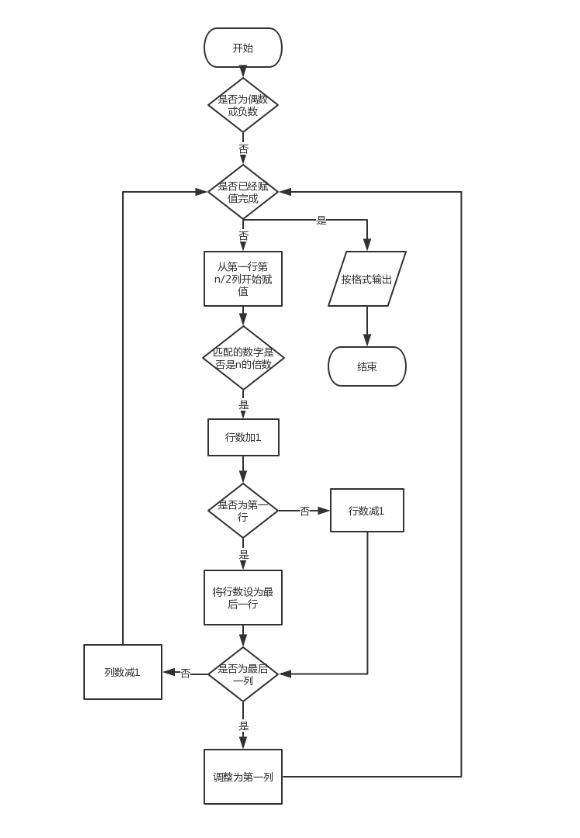
格式判断：.isInt

过程：首先判断这个文本里是否有空格，来确定他是否是用制表符分隔的，然后按行读取文本中的内容，最后将最初的第一行用制表符分割计算出他含有多少个数字即可得到这个幻方的大小。同时计算出行的和，进行第二行的时候时时比较加和结果，数字个数。

运行结果：

### generateMagicSquare()

中文注释在代码中添加

流程图：

出现异常的原因：

Exception in thread "main" java.lang.ArrayIndexOutOfBoundsException（数组越界）

Exception in thread "main" java.lang.NegativeArraySizeException（定义幻方的大小的时候数组长度不能为负）

## Turtle Graphics

本实验利用一个已经存在的turtle工具进行图形绘制，并完成基本的数学中的正多边形的边数和角度互相求解的问题，其中的凸包问题之前没有接触过代码的写法，查阅了网上关于凸包的算法最后成功修改好

### Problem 1: Clone and import

方案:在想要获取的位置右键,gitbash here,git init来创建本地git仓库，输入git clone加所给网址

### Problem 3: Turtle graphics and drawSquare

方案：已给.draw()函数和.turn()函数只需按照正方形长度适时拐弯即可

### Problem5: calculatePolygonSidesFromAngle

方案：根据角度计算出每一个外角的度数,利用多边形外角和恒为360度求得有多少个外角,即有多少个内角,多少条边.

### Problem 5: Drawing polygons

方案：该方法和正方形类似只是需要先计算正多边形的内角然后根据内角度数来进行.turn操作

### Problem4:calculateRegularPolygonAngle

方案：计算正多边形内角大小，最简单的方法就是利用凸多边形的外角和都是360然后根据边数计算每个外角大小，然后180-外角就是内角度数

### Problem 6: calculateBearingToPoint

方案：通过连接起始点和偏转点这样构成一条向量，利用math库中的函数atan2计算角度，困难:需要注意的是本题为0的角度和极坐标为0的角度不一样，第一次写好程序后,test没有通过,在观察输出后发现了问题,所以需要在转换后将角度做改变，然后分组判断输出结果

**3.2.7 Problem7: calculateBearings**

方案：此方法是上一条的累积，只要时时改变当前点，不断匹配下一个偏转点即可

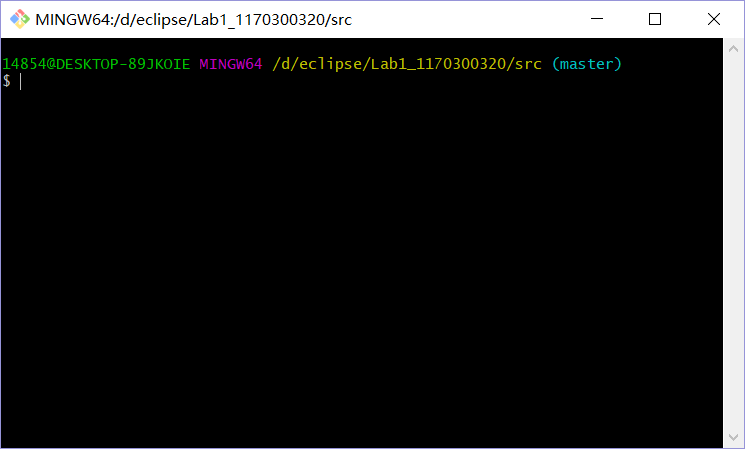
### Problem 8 : convexHull

方案：利用两点确定一条直线，可将给出的点的集合两两一条计算出直线的斜率，当这两点构成的直线使其他点均分布于一侧时这条线就是凸包上的线，注意当有三个点甚至更多在同一条直线上时，要选择横坐标分别最大最小或纵坐标最大最小的两点加入凸包的集合中

### Problem 9: Personal art

方案：画出一个边长不断减少的漩涡正方形

### Submitting

在当前文件夹git bash here 

依次输入 git init;git add P1;git commit -m ‘layout’;git push origin master即可

## Social Network

本实验就是数据结构中关于邻接表邻接矩阵遍历无向图的java型问题，所以难度相对不那么高，但由于第一次接触java所以对person类的定义不是很熟练，修改了很多次，最后终于和实验给好的代码准确的匹配好

### 设计/实现FriendshipGraph类

方法：添加点；添加边；获得两人之间的最短距离

添加点：新添加一个Person类型的量将括号中的内容作为他的姓名，并赋予他一个编号，实时监测名字的输入是否重复，若重复直接退出程序

添加边：将两人关系的relation[][]设为1记为两人之间可直接联系

获取两人之间的最短距离：采用Floyd算法

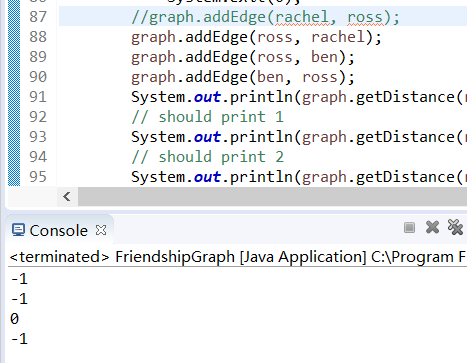
### 设计/实现Person类

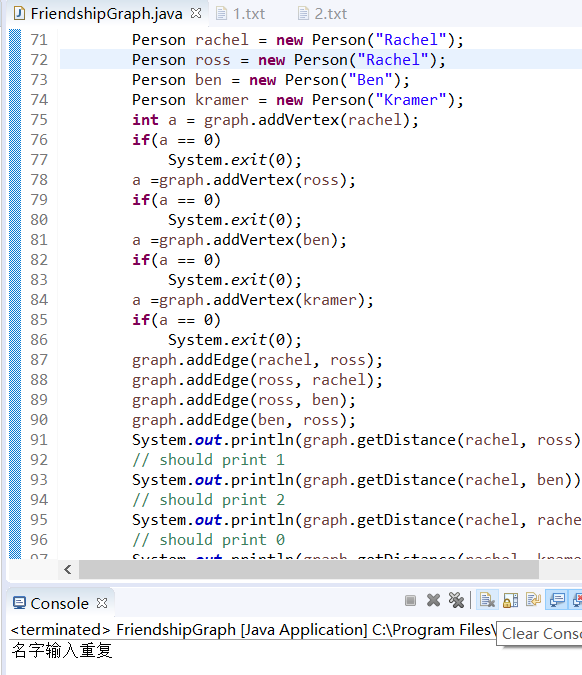
根据输入添加此人的姓名，若姓名没有重复那么就赋予他一个新的编号

### 设计/实现客户端代码main()

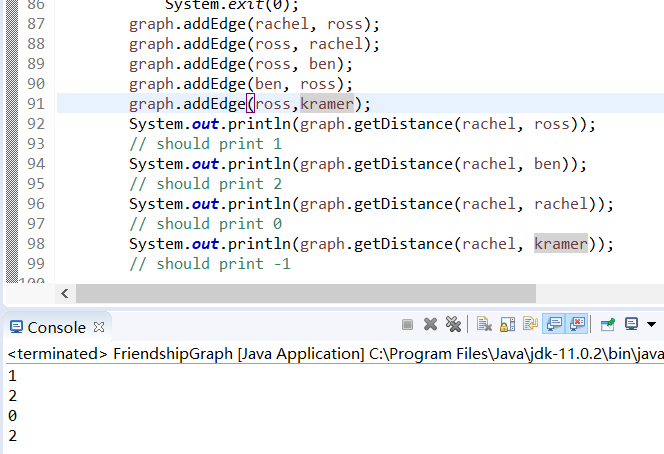
参照实验手册完成给出的例子测试

结果如下：

删去rachel->ross结果：

Ross替换为rachel: 

### 设计/实现测试用例

改变后的测试样例（增加一条边）：

## Tweet Tweet（选作，额外记分）

本实验是一个有实际意义的问题，他选取社交网络的背景，研究人与人之间关注与被关注，关注者的多少决定了此人的受欢迎程度，其中还包括推测某人可能关注了谁，我拓展的功能是将关注同一个话题的人同时检索出来放在一起，还是用到了正则表达式。

### Problem1 Extract

#### Timespan

方案：根据Tweet中给出的关于start和end的方法得到最早开始的时间和最晚结束的时间即可

#### getMentionedUsers

方案：根据正则表达式



在判断@他人的同时排除引用邮箱的可能，最后将被@的人输出，并且在@人名的前后都用空格避免了发现有效字符的可能

### Problem 2 Filter

#### writtenBy

方案：利用tweet中的getauthor方法，同时将两个字符串小写然后进行比较，若比较后发现完全一致则将此条tweet加入输出列表中

#### inTimespan

方案：利用tweet中关于start和end的方法与给出的时间戳进行比较，若不仅开始时间晚于开始计算时间，结束时间又早于计算时间那么这条tweet即应加入输出列表

#### containing

方案:利用indexOf方法来搜索tweet.getText中是否含有想要搜索的关键词

### Problem3 SocialNetwork

#### guessFollowsGraph

方案：相当于利用之前写过的getmentionedusers方法，只是此方法的输出格式为map形式，所以需要先同一个人写的tweet全部取出放在一起，然后对他们集中进行getmentionedusers检索将所得到的@的名字取出来全部放在同一个key的value中

#### influencers

方案：基于上一个方法，得到一个人大概的followgraph后,先将每个人的follow者的名字加入map中并将他们的value置零，然后开始按照key遍历map出现加入的名字时就将其后的数字加1，这样就可以统计每个人他们的关注着数目有多少，最后利用arrays.sort函数进行降幂排序

### Problem 4 MySocialNetwork

#### sametopic

方案：利用正则表达式，将参与同一个话题的用户加入同一个列表中

# 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 |
| 2019-02-27 | 14:00-24:00 | 配置环境 | 按计划完成 |
| 2019-2-28 | 14:00-22:00 | 完成P1 | 按计划完成 |
| 2019-3-1 | 14:00-23:00 | 开始P2剩convexhull | 按计划完成 |
| 2019-3-2 | 14:00-22:00 | 完成P2修改convexhul | 按计划完成 |
| 2019-3-3 | 14:00-22:00 | 开始Friendship修改Person | 未按计划完成 |
| 2019-3-4 | 14:00-22:00 | 完成Friendship | 按计划完成 |
| 2019-3-6 | 15:00-22:00 | 开始twitter并完成extract,filter | 按计划完成 |
| 2019-3-7 | 15:00-22:00 | Getmentionedname函数 | 未按计划完成 |
| 2019-3-10 | 19:00-22:00 | 修改getmentionednames函数 | 延期1小时完成 |
| 2019-3-11 | 16:00-17:00 | 完成getsmarter的最终修改，并测试socialnetwork的几个方法是否正确 | 按计划完成 |

# 实验过程中遇到的困难与解决途径

P1因为第一次写java语言的实验所以很多语法还不是很清楚到网上找了很多java入门的教程，最后才在两天内勉强完成第一部分内容

P2这部分开始做的还很顺利，因为经历过第一部分对java已经稍微有了了解，但是进行到凸包问题的时候，本来写完以为可以通过但是由于有一条边上多于两个点的情况没有考虑到，所以又多花了一天时间来修改

P3因为对类的定义不够清晰所以在person如何编写处停留了很长时间，最后不断删改，终于符合了给出的括号中包含人名的格式

P4但看要求并没有觉得这部分实验很困难，而且第一遍test很好通过，但最后检查代码的时候发现有的部分写的还不正确，但test并不能全部检查出来，这时才发现实验所给的test代码并不能真正的检查出代码实现的功能，所以花费了很长时间在修改main函数来验证功能的真正实现上，最后终于完成

# 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. Java编程语言是否对你的口味？

第一次学习还不是很熟练,与之前学过的C语言感觉有很大的不同

1. 关于Eclipse IDE

下载花了一下午的时间因为之前安装过jdk的其他版本,这次又安装了新版本所以安装时费了很长时间,最后不得已格式化了电脑

1. 关于Git

但看指导的pdf也不怎么能弄懂,但实践几次真正的上传和删除,更新,就能对git的基本操作有比较基础的认识和了解了

1. 关于CMU和MIT的作业

在阅读原版英文材料上还是有一定的难度,在意思的理解上难避免会有偏差,所以在实验过程后与其他人交流功能的实现对理解实验内容有很大的帮助

1. 关于本实验的工作量、难度、deadline

本次实验作为第一次入门的java实验我觉得难度还是比较合适的,可能这次实验的时间有一部分我用来安装配置环境,所以还利用的不够充分,下一次实验应该情况会好很多

1. 关于初接触“软件构造”课程

第一次这样正式的学习java语言的编程,用了大概两周的时间完成junit,和java本身语言的学习,虽然时间有些紧张,但最后完成还是很有成就感的