

**2022年春季学期  
计算学部《软件构造》课程**

**Lab 1实验报告**

|  |  |
| --- | --- |
| 姓名 | 瞿久尧 |
| 学号 | 120L022314 |
| 班号 | 2003012 |
| 电子邮件 | 2668136485@qq.com |
| 手机号码 | 15390311738 |

**目录**

[1 实验目标概述 1](#_Toc17950)

[2 实验环境配置 1](#_Toc25442)

[3 实验过程 1](#_Toc6511)

[3.1 Magic Squares 1](#_Toc8424)

[3.1.1 isLegalMagicSquare() 2](#_Toc21521)

[3.1.2 generateMagicSquare() 3](#_Toc19082)

[3.2 Turtle Graphics 3](#_Toc12591)

[3.2.1 Problem 1: Clone and import 4](#_Toc28997)

[3.2.2 Problem 3: Turtle graphics and drawSquare 4](#_Toc7784)

[3.2.3 Problem 5: Drawing polygons 5](#_Toc12926)

[3.2.4 Problem 6: Calculating Bearings 6](#_Toc14507)

[3.2.5 Problem 7: Convex Hulls 6](#_Toc24584)

[3.2.6 Problem 8: Personal art 6](#_Toc30900)

[3.2.7 Submitting 7](#_Toc3352)

[3.3 Social Network 7](#_Toc21669)

[3.3.1 设计/实现FriendshipGraph类 8](#_Toc20479)

[3.3.2 设计/实现Person类 8](#_Toc1885)

[3.3.3 设计/实现客户端代码main() 8](#_Toc18093)

[3.3.4 设计/实现测试用例 9](#_Toc29551)

[4 实验进度记录 9](#_Toc7235)

[5 实验过程中遇到的困难与解决途径 9](#_Toc1769)

[6 实验过程中收获的经验、教训、感想 9](#_Toc64)

[6.1 实验过程中收获的经验和教训（必答） 9](#_Toc28241)

[6.2 针对以下方面的感受（必答） 10](#_Toc17422)

# 实验目标概述

本次实验通过求解三个问题，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

安装jdk1.8>>安装IDEA>>配置环境变量

安装git

IDEA与之前使用以及实验手册所描述的Eclipse有所区别，所有的几乎都得自己上网查询摸索，git的使用与cmd类似，但与平时的可视化界面不同，还得多加练习

本次实验在GitHub Classroom中的URL地址为：

<https://classroom.github.com/a/K_B5NllB>

# 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（但无需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

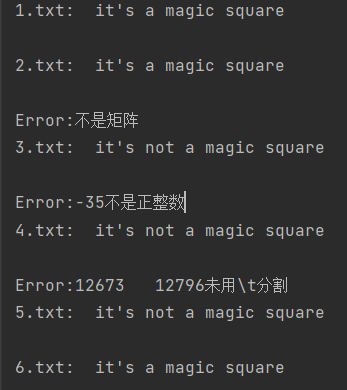
根据对幻方的描述，即每行每列以及两条对角线分别相加均为同一个数的正方形矩阵，设计程序来判断文件输入是否为幻方。

### isLegalMagicSquare()

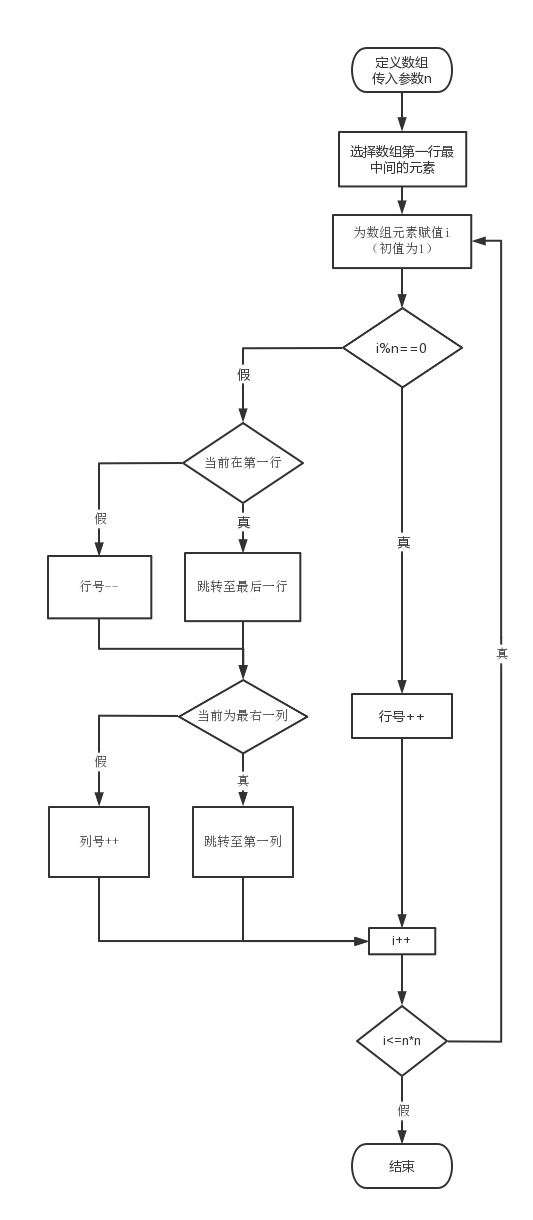
按步骤给出你的设计和实现思路/过程/结果。

首先需要判断文件读入的矩阵是否合法，即一些基本的要求：每个数必须是正整数、数之间用\t分隔，且这是一个n\*n的矩阵。若在这步出现问题，则会报错并且终止程序。

再根据幻方的定义，首先计算每行分别相加的值是否相同，并且得出值，方便后续比较。将上一步得到的值与每列之和以及对角线之和比较。若和均相等，则返回true，即为幻方；若和不相等，则返回false，不是幻方。



### generateMagicSquare()



先创建一个n\*n的矩阵，然后从1开始为二位数组赋值，直到n\*n

n是奇数，先选中第一行最中间的元素进行赋值，然后向右移一列，向上移一行，继续为数组元素赋值。若在过程中碰到最右一列，则下一步移到最左一列；若在过程中碰到第一行，则下一步移到最下面一行。赋值n个元素后会回到第一行，这时候下移一行，重复上述操作。即可完成对n\*n个元素的赋值。

若n为偶数，则会因为row++的时候导致数组越界

若n为负数，会因为数组magic[row][col]使用的时候col为负而出现问题

## Turtle Graphics

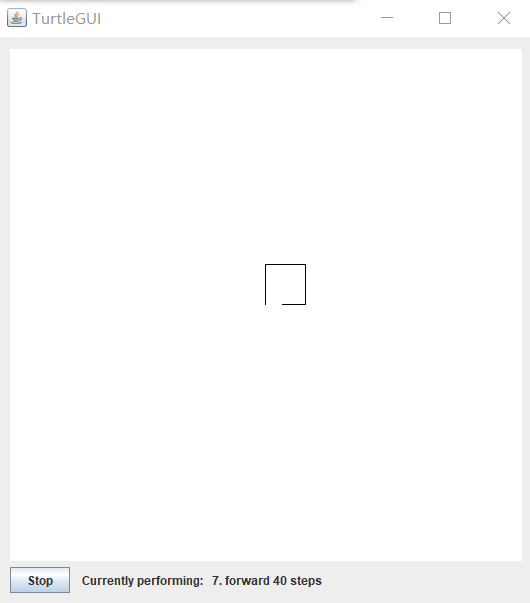
根据已经设计好的Action、DrawableTurtle、PenColor、Point、TurtleGUI、Turtle来完善TurtleSoup中的函数，使之能完成各功能，主要是实现绘画和寻找凸包的功能。

### 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。

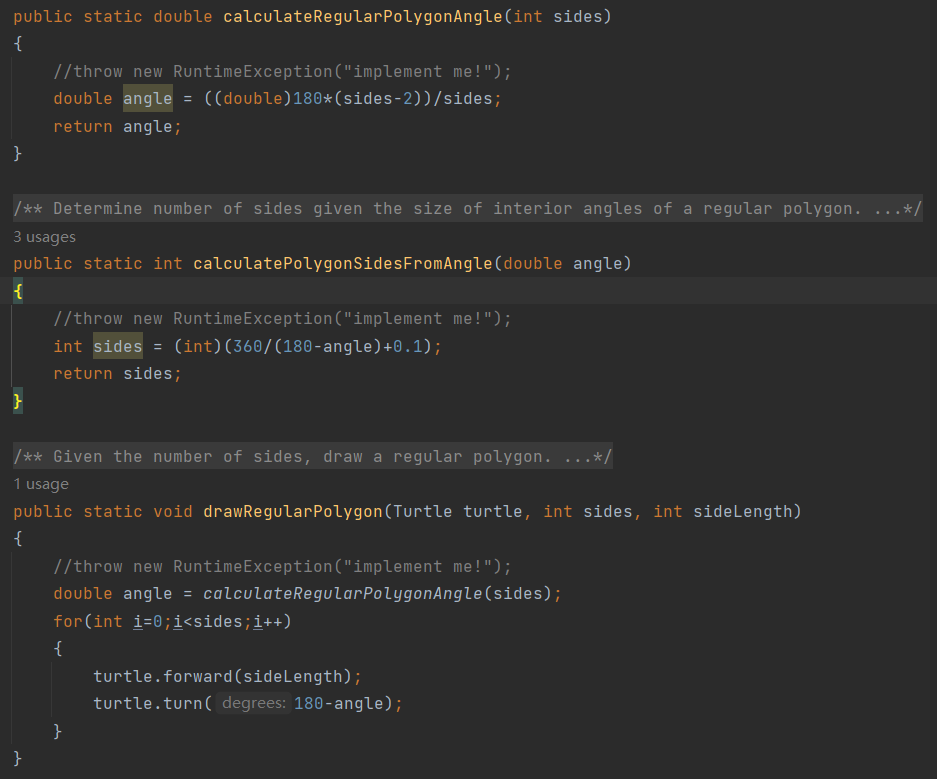
### Problem 3: Turtle graphics and drawSquare

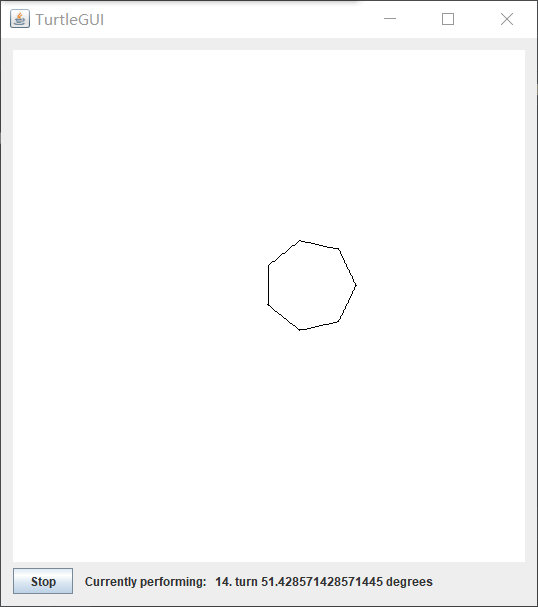
认识TurtleSoup是如何调用其中的各类以及函数来实现绘画，即可根据四边形的角度和长度关系来画出Square



### Problem 5: Drawing polygons

很容易得出正多边形内角公式，则同problem3，通过角度和长度来进行绘画

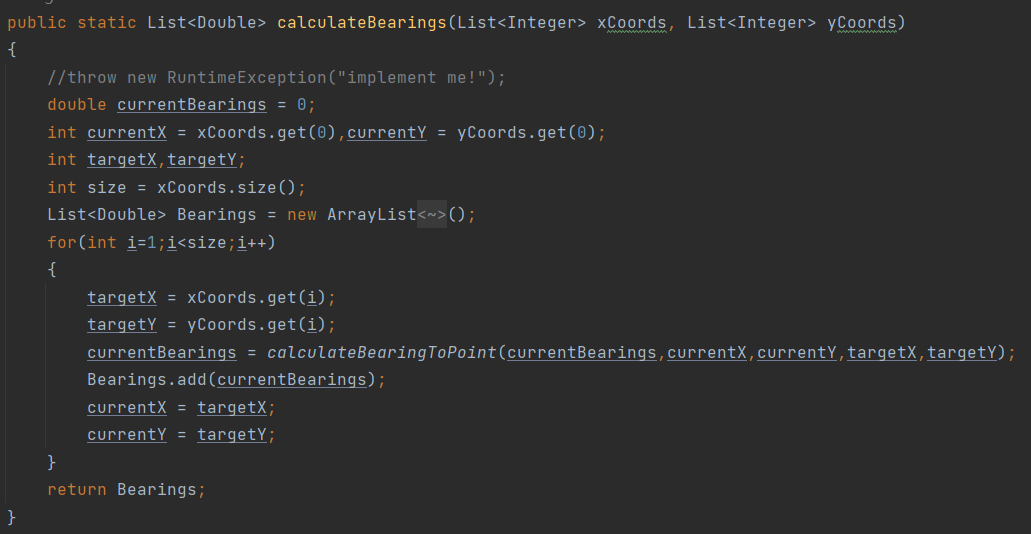




### Problem 6: Calculating Bearings

利用数学的方式，在点和点之间构造关系，加上对向量角度的判断，求出角度。

在此问题中需要学会对列表的使用

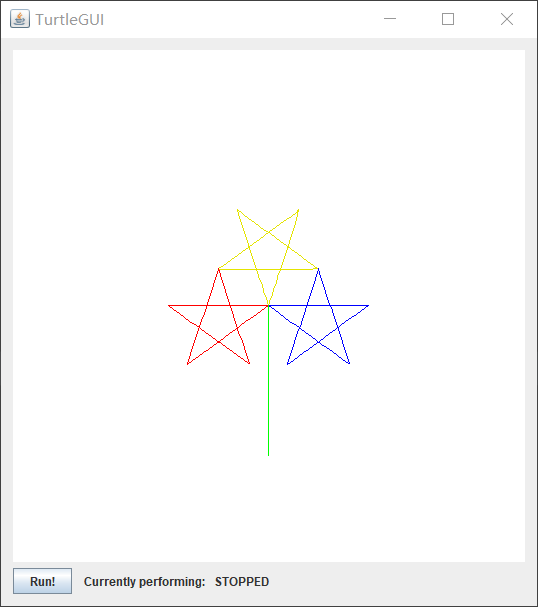


### Problem 7: Convex Hulls

凸包问题用gift-warpping算法即可解决，但仍需用到problem6中的函数对角度进行计算，进而寻找的偏转最小的角度。需要注意的是，在凸包边界上的点不能算做凸包集合中的点

### Problem 8: Personal art

目的在于激发学生的创造力，利用循环等方法，熟练使用turtlesoup进行创作



### Submitting

git status 查看当前仓库状态信息

git add . 将仓库下的所有文件加入暂存区

git commit -m "message" 提交文件到本地仓库

git push 将本地库推送至远端

## Social Network

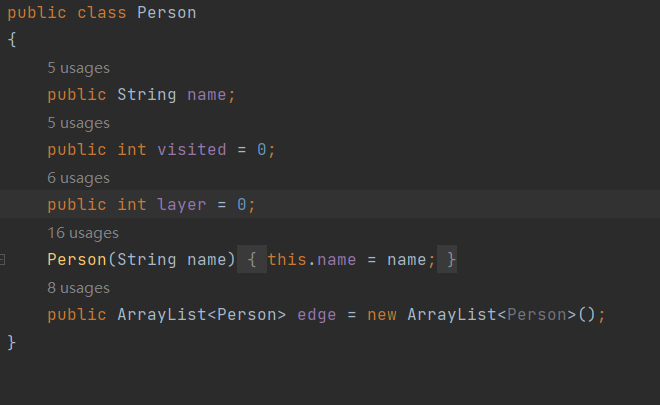
该任务旨在创造一个朋友关系网，描述一个圈子中哪些人互为朋友，哪些人和哪些人并不是朋友。并且对每个人都有较为详细的描述，包括他的名字以及他的朋友有哪些。在此任务中可以通过增加人数以及增加朋友间的关系来构建这个图，同样，还能寻找到两个人相互联系的最短距离。

### 设计/实现FriendshipGraph类

由于不能设置静态函数，所以，在编写方法的过程中需要尽量考虑到适合类对象使用的方法。配合Person类使用后，可以方便的调用Person来对朋友之间的关系进行存储，因此在此类中只需要定义一个总的图，用于存储这个圈子中有哪些人。因此，addVertex和addEdge函数变得相对简单，只是需要判断是否符合条件，即避免出现同名的两个人，避免出现不在圈子中的人想添加朋友关系的情况。至于计算距离时，可以考虑使用广度优先搜索，一定能找出最短距离。类似邻接链表的图结构，让BFS能较好执行

### 设计/实现Person类

Person类可以理解为用户自定义的结构体，其中包含用户的名字以及身边的朋友，对此，我们可以用一个列表来表示一个人的朋友集合，因此朋友可以一直增加。为了寻找朋友间相互联系的最短距离，我还为他们设置了可访问的标志符，来判断在寻找联系的过程中是否被访问过；并且，根据广度优先算法的树状图，我还设置了层数，代表这个人在这次BFS中所在位置，进而方便确定两人间的距离



### 设计/实现客户端代码main()

由于FriendshipGraph中不存在除main函数以外的静态方法，因此需要定义一个对象来实现相关功能。然后创建一些person来构建这个关系网，并且为他们之间建立朋友关系，最后根据需要来计算他们之间的距离，来验证我们的实现结果。

### 设计/实现测试用例

测试用例尽量考虑到多种情况，但由于我们已经在设计的时候对错误的情况进行报错，所以测试用例就可稍微减少工作量，对于基本情况的需求满足的测试完成即可。

而在进行复杂方法的时候，我们需要尽可能考虑到多种情况来对计算距离的函数进行检验，满足函数对于是朋友，朋友远近，不是朋友等多方面问题进行测试。

# 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。

|  |  |  |  |
| --- | --- | --- | --- |
| 日期 | 时间段 | 任务 | 实际完成情况 |
| 2021-05-01 | 18:30-19:30 | 编写问题1 | 按计划完成 |
| 2022-05-03 | 19:00-19:40 | 编写问题2的problem1-5 | 提前20min完成 |
| 2022-05-03 | 20:00-21:00 | 编写问题2的6 | 按计划完成 |
| 2022-05-04 | 15:00-17:00 | 编写问题2的7 | 延期30min完成 |
| 2022-05-05 | 19:00-20:00 | 编写问题2的8 | 按计划完成 |
| 2022-05-05 | 20:00-23:00 | 编写问题3 | 延期1小时完成 |
| 2022-05-06 | 8:00-9:00 | 写实验报告 | 按计划完成 |

# 实验过程中遇到的困难与解决途径

|  |  |
| --- | --- |
| 遇到的困难 | 解决途径 |
| Git push时出现问题无法连接至github | 上CSDN进行搜索，git config --global --unset https.proxy |
| 不太了解关于列表、集合、队列的使用 | 上网搜索关于这些相关的定义，以及他们所定义的函数的相关用法，并且在debug时对这些的执行和返回有了深刻认识 |
| 实验过程中的算法知识有遗忘的现象，导致不知道如何编写程序 | 复习凸包相关的知识，了解了gift-warpping算法的原理和使用；复习了BFS的用法，以及如何用邻接表的方式来实现 |

# 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（必答）

实验过程中会遇到很多未知的，在知识范围以外的问题，但问题发现的过程就是提高的过程，往往需要通过自己查资料、询问等多种方式来解决问题，快速加深记忆，提升自己。

## 针对以下方面的感受（必答）

1. Java编程语言是否对你的口味？与你熟悉的其他编程语言相比，Java有何优势和不足？

是的，java是面向对象开发，与面向过程不同；而且java少了对于指针的调用，很多列表集合以及方法的设计都能很大程度上减少编程人员的工作量。但有时候java必须定义对象才能使用的方法会造成一些困扰。

1. 关于Eclipse或IntelliJ IDEA，它们作为IDE的优势和不足；

Eclipse较为稳定，大家广泛使用，但界面较为生硬，不符合现代人的追求。IDEA对于很多问题都提供了解决的方法以及自动解决的快捷通道，并且在debug时会很智能且合理地给出各参数的值，方便进行bug修改。

1. 关于Git和GitHub，是否感受到了它在版本控制方面的价值；

是的，对于版本的控制，git和github无疑为开发人员提供了极大的便利，减少了小组成员共同开发的无用时间消耗，并且快速合并复制等操作能够帮助编程人员减少失误。

1. 关于CMU和MIT的作业，你有何感受；

很好，非常详细，但是对于中国的学生来说，可能更需要一些由自己学校设计的更符合学校学生学习的作业，难度循序渐进较为合适

1. 关于本实验的工作量、难度、deadline；

难度不大，但对于未学过java的同学来说，入门需要一定时间，工作量比其他课程的实验量大

1. 关于初接触“软件构造”课程；

软件构造好啊，对于软件开发的过程以及细节都做了较为详细的说明，能够为将来同学们的软件开发生涯打下了坚实的基础