

**2022年春季学期  
计算学部《软件构造》课程**

**Lab 1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](#_Toc97648154)

[2 实验环境配置 1](#_Toc97648155)

[3 实验过程 1](#_Toc97648156)

[3.1 Magic Squares 1](#_Toc97648157)

[3.1.1 isLegalMagicSquare() 1](#_Toc97648158)

[3.1.2 generateMagicSquare() 1](#_Toc97648159)

[3.2 Turtle Graphics 1](#_Toc97648160)

[3.2.1 Problem 1: Clone and import 2](#_Toc97648161)

[3.2.2 Problem 3: Turtle graphics and drawSquare 2](#_Toc97648162)

[3.2.3 Problem 5: Drawing polygons 2](#_Toc97648163)

[3.2.4 Problem 6: Calculating Bearings 2](#_Toc97648164)

[3.2.5 Problem 7: Convex Hulls 2](#_Toc97648165)

[3.2.6 Problem 8: Personal art 2](#_Toc97648166)

[3.2.7 Submitting 2](#_Toc97648167)

[3.3 Social Network 2](#_Toc97648168)

[3.3.1 设计/实现FriendshipGraph类 2](#_Toc97648169)

[3.3.2 设计/实现Person类 2](#_Toc97648170)

[3.3.3 设计/实现客户端代码main() 2](#_Toc97648171)

[3.3.4 设计/实现测试用例 3](#_Toc97648172)

[4 实验进度记录 3](#_Toc97648173)

[5 实验过程中遇到的困难与解决途径 3](#_Toc97648174)

[6 实验过程中收获的经验、教训、感想 3](#_Toc97648175)

[6.1 实验过程中收获的经验和教训（必答） 3](#_Toc97648176)

[6.2 针对以下方面的感受（必答） 3](#_Toc97648177)

# 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

本次实验采用IDE为 IDEA 2021.3,Java环境的配置，暑假预习的时候就已经安装完成了，在此处不再过多陈述

在这里给出你的GitHub Lab1仓库的URL地址。

https://github.com/ComputerScienceHIT/HIT-Lab1-120L031815

# 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

此程序要实现的功能是读入所给的5个txt文档中的矩阵，判断其是否为幻方矩阵，并返回true / false。在完成上述功能后，进一步实现生成幻方矩阵并保存至本地、调用函数判断其是否为幻方的功能。同时还要求，在读入矩阵时时需先进行多个维度的输入检测，确保其符合基本要求后，再检测其是否满足幻方。具体流程为：

1. 判断矩阵格式是否正确，即是否出现行列数不等，非矩阵，含非正整数，数字间不以\t分割等错误情况
2. 确认矩阵格式正确后，检测其是否为幻方，即判断矩阵每一行元素和，每一列元素和，以及每一条对角线元素和是否都相等。

### isLegalMagicSquare()

函数设计：

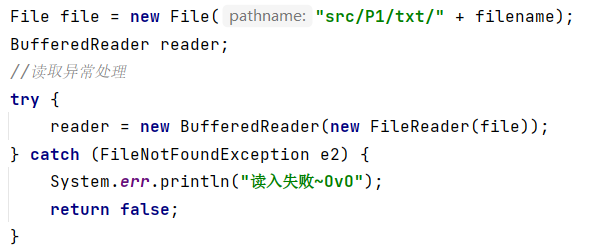
①txt文档中矩阵的读入

②判断矩阵格式正确性

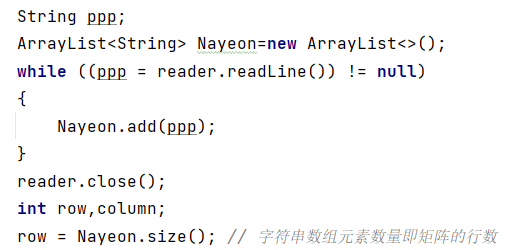
③判断是否满足幻方要求

函数实现思路：

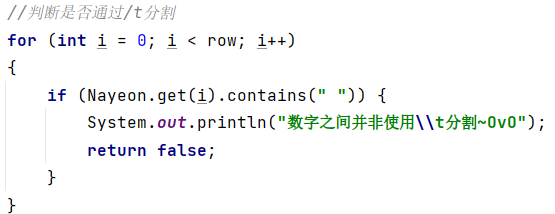
1.创建输入流，读入txt文档，同时使用try，catch结构判断是否读入成功，若失败则返回false及提示语句，代码如下：



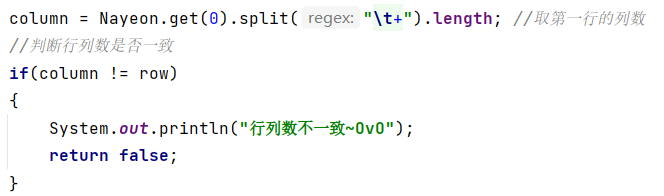
2.使用readline函数，将矩阵按行保存在字符串数组当中



3.矩阵格式检测部分，检测是否用/t分割，行列数不一致等，部分代码如下：

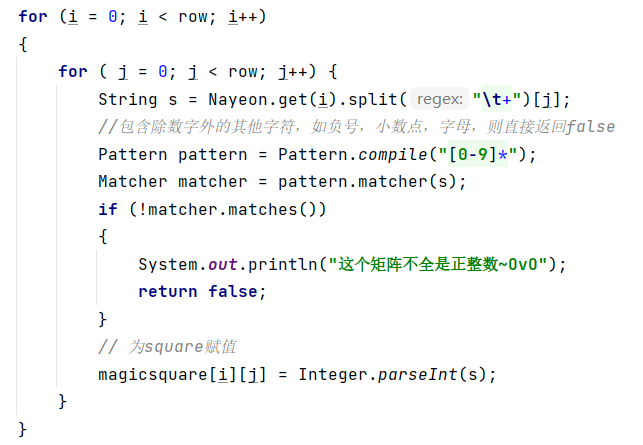


/t分割检测

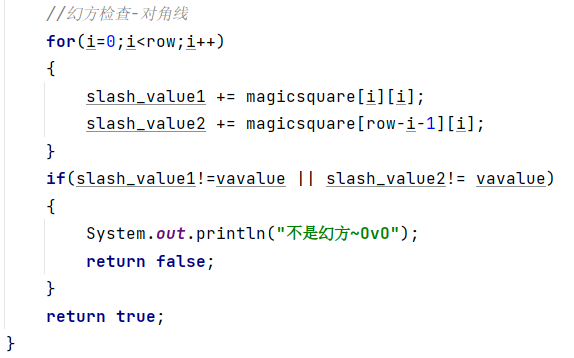


行列数一致检测

4.幻方判断部分，首先是将字符串数组中存储的数字信息转到矩阵里面



然后通过幻方的性质对矩阵进行检测，检测矩阵每一行，每一列的和以及对角线的和是否相等，若判断为幻方则返回true，部分代码如下：



### generateMagicSquare()

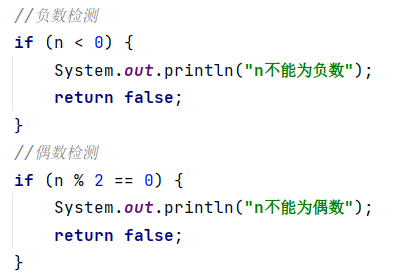
函数设计：

①生成一个矩阵并保存到txt文档中

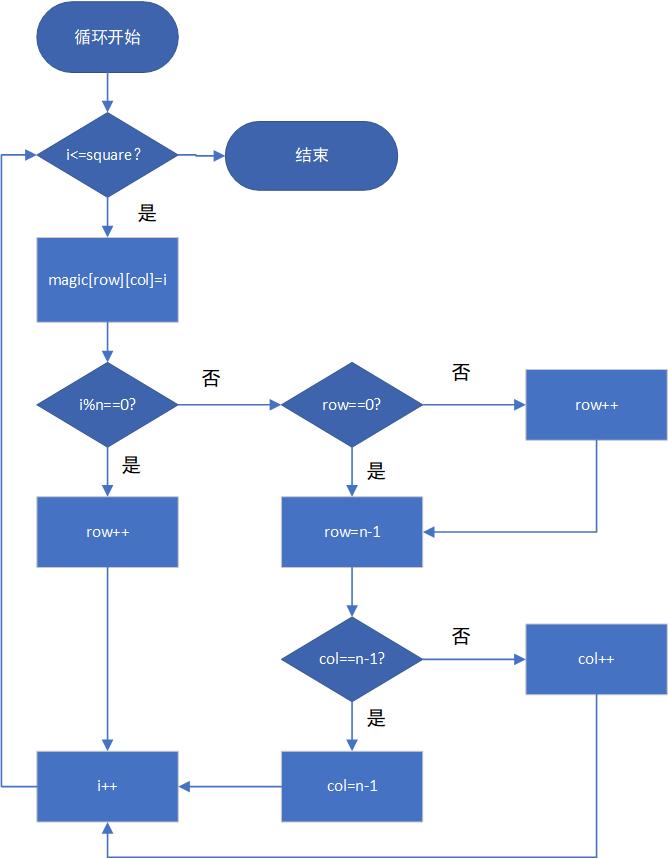
②判断其是否为幻方

函数实现思路：

1.矩阵长度n检测，若为偶数或者负数则直接退出，代码如下：



2.矩阵赋值部分的流程图



根据传入函数的参数n，创建二维数组magic[n][n]，根据循环次数控制变量i，依次向矩阵中填充的数字。在第一次循环时，起点为第一行中间，之后每次循环，依次向上一次循环填充位置的右上角移动一格，若遇到上边界就返回到下边界，遇到第一行，则返回最后一行，遇到最后一列则返回第一列。同时，每输入n个数填充位置就下移一行，直到整个矩阵填充完毕。

对整个函数进行注释后，研究其输入n为偶数或负数时产生的异常情况。

① n为偶数时，错误提示为数组下标越界。其原因如下：row初始在第一行，每一轮向上移动n次（遇到顶就移到最后一行），然后再下移一行，开始下一轮循环，即经过每一次循环，row的位置从0开始下移两行；当n为奇数时，row每一轮下移两行，可以正好滑到边界，但是当row为偶数时，这样会造成数组越界。

② n为负数时，错误提示为数组大小为负数，原因显然。

## Turtle Graphics

Turtle Graphics是一个广泛存在于不同编程语言中的图形库，在这一任务中，在java环境下完成一个简单Turtle类的搭建与使用，完成多个函数的设计以及使用turtle完成一个自己独特的作品。

### Problem 1: Clone and import

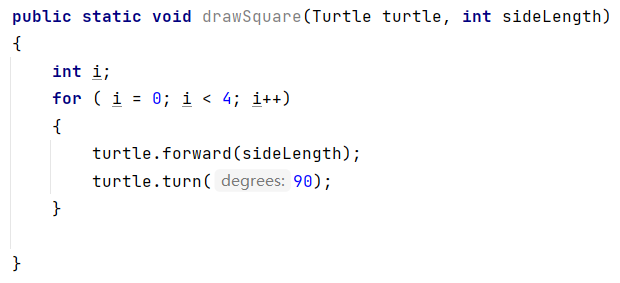
https://github.com/rainywang/Spring2022\_HITCS\_SC\_Lab1/tree/master/P2获取代码，git clone至本地，移至P2文件夹下，修改各个文件的路径，即可运行。

### Problem 3: Turtle graphics and drawSquare

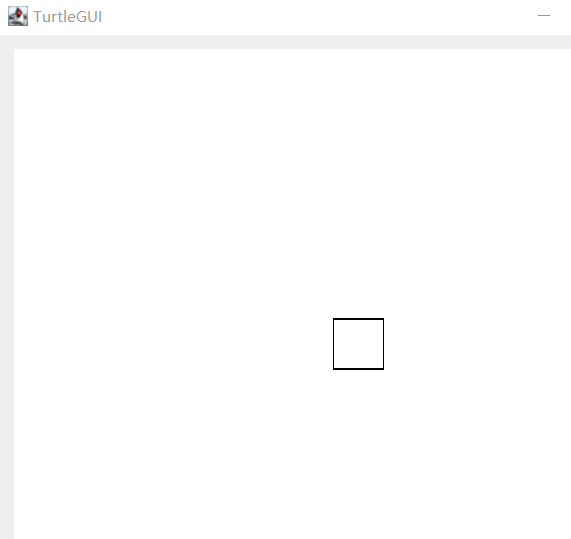
使用程序已经提供了的forward与turn两个方法，完成正方形绘制。

具体程序实现非常简单：利用for循环，四次执行画线，改变角度的代码即可。

程序代码及运行结果如下：



运行结果如下：

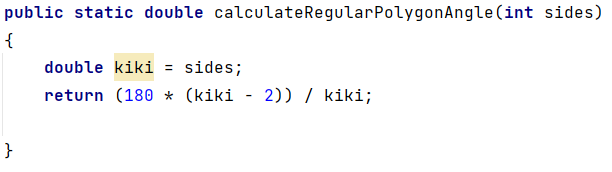


### Problem 5: Drawing polygons

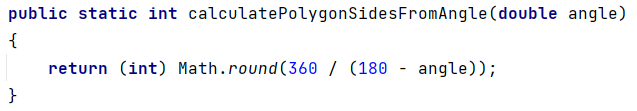
查询得正多边形内角度数公式：



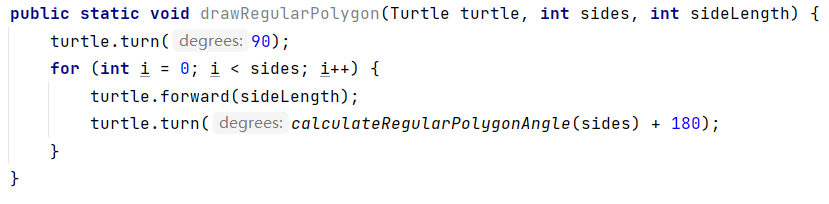
函数calculateRegularPolygonAngle的实现如下：



函数calculatePolygonSidesFromAngle的实现只需将公式反推，即可实现从角度反推边数，代码如下：



最后实现drawRegularPolygon ( )函数。保存正多边形内角为ang，需要注意每次旋转的角度为180-ang，代码如下：



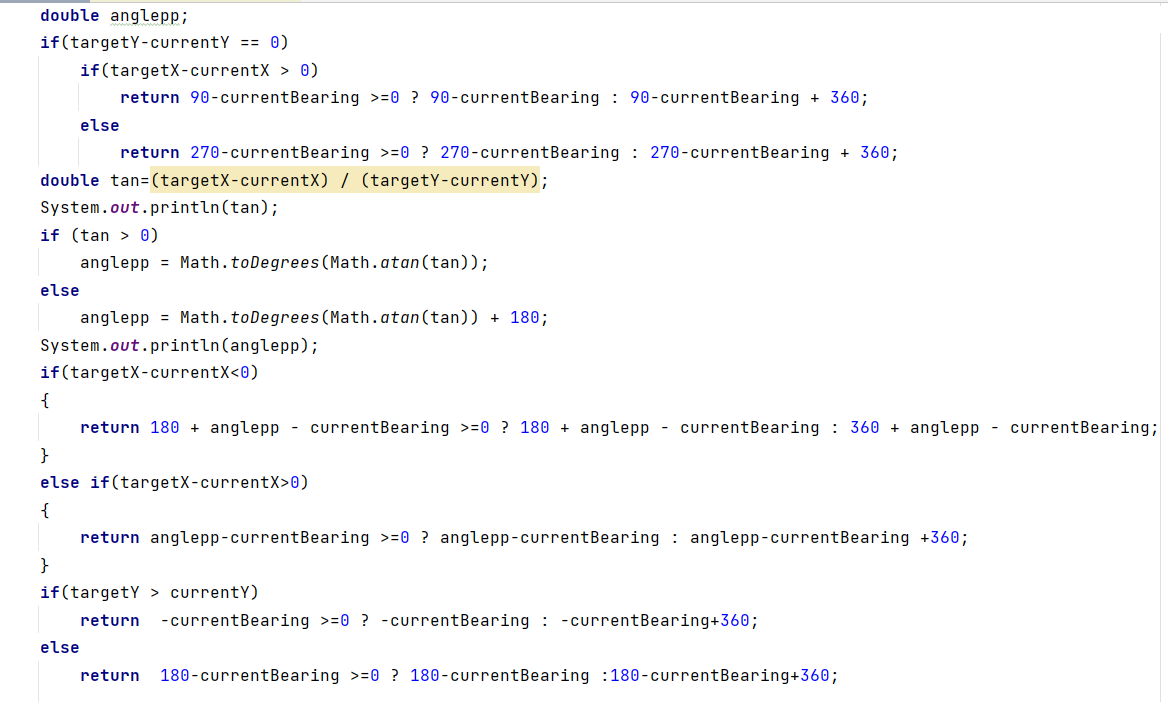
### Problem 6: Calculating Bearings

函数设计：

已知当前点的坐标，下一点坐标，以及当前点目前所指方向，求需要顺时针转动的角度为多少，才能使当前点指向下一点。

函数设计思路：

首先，可以根据两点间的与计算出两点连线与轴夹角的值，进而求出两点连线与轴的夹角角度，求出这一数据后，分情况处理即可，代码如下：



情况大致分为六类，分别是target在current的左上，右上，左下，右下，正左，正右，同时对于结果的输出方面，将其归一化到0-360度这个范围内。

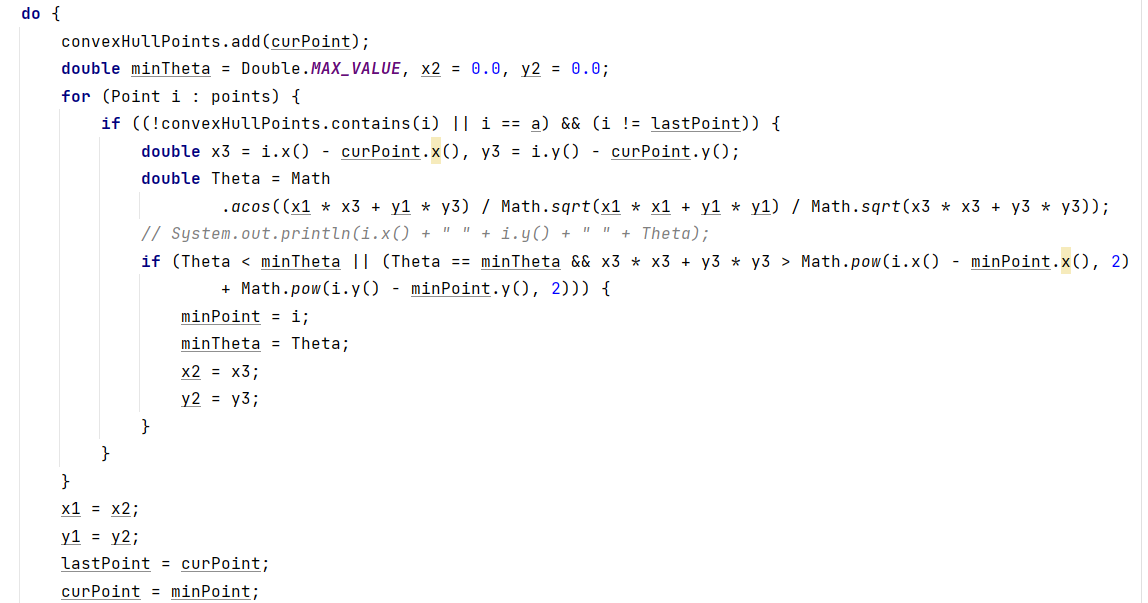
### Problem 7: Convex Hulls

在这一环节中，需要设计实现求凸包最小点集的函数。查阅Wiki，借鉴礼物包装算法的思路，设计convexHull函数如下：

首先对points中保存的点的个数进行判断，若点数小于等于3，则points中的所有点都构成了最小凸包，直接返回points；若点数大于3，建立保存最小凸包的点集minpoints，继续之后的步骤：

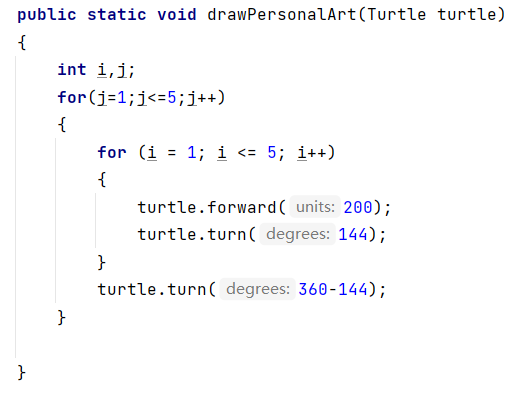
以y坐标最小的点作为起始点，起始转角bearing设为0，利用calculateBearingToPoint函数，计算当前点到其余点的转角，以转角最小对应的点作为下一点；若同时有多个点对应最小转角，则取距离当前点距离最大的点作为下一点；更新转角，对下一点重复以上步骤，直到计算出的下一点为起始点，结束循环。

代码如下：

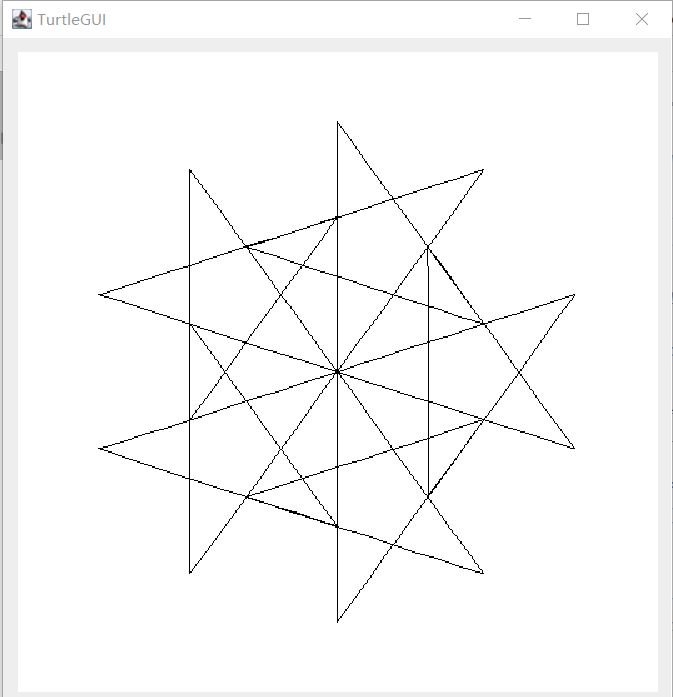


### Problem 8: Personal art

自己设计了一个由五个五角星组成的图形，代码部分如下：



结果如下：



### Submitting

如何通过Git提交当前版本到GitHub上你的Lab1仓库。

## Social Network

构建一个程序，实现人际关系图的表示。设计两个类：一个类表示人与人之间的关系网络，需要能够拓展为有向图；另一个类为网络中的节点，表示关系网络中的个体，另外，还需要为设计的人际关系图编写测试用例，以验证程序的正确性。

### 设计/实现FriendshipGraph类

FriendshipGraph类中共有三个方法，addVertex，addEdge，getDistance，下面具体介绍每个方法实现的功能及设计思路。

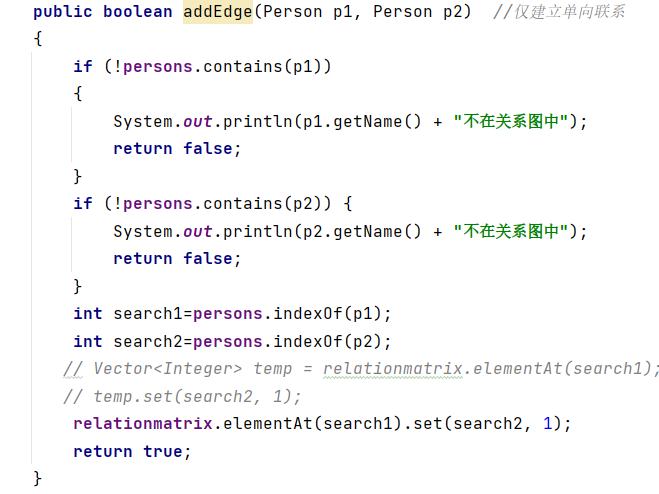
①addVertex

这个部分要实现的功能是添加元素，就是将一个人加入到人际关系网当中，因此需要实现的部分有：1.将人名添加到人际关系网的人名表中。2.初始化新人与其他关系网中的人的关系。代码如下：



②addEdge

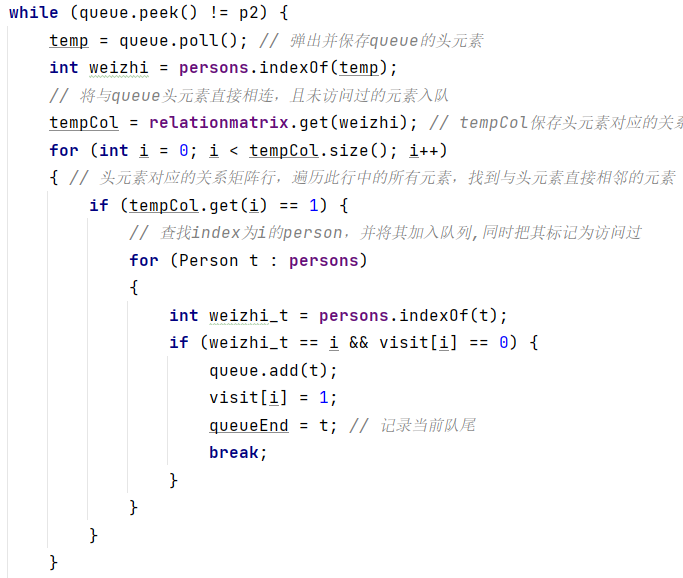
这个部分要实现的功能是在某两个人之间加入一条边（有向），即在人际关系矩阵当中建立一条两人之间的有向边，代码如下：



首先进行输入检测，若输入的两个人有某个人不在关系图中，返回false，然后通过indexof找到这两人在列表中的位置，并加一条边。

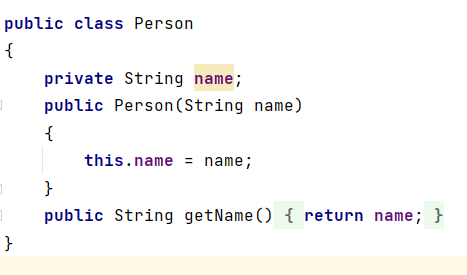
③getDistance

这部分要实现的功能是输入两个人，返回两个人的社交距离（即在图中的距离），距离的搜索通过BFS实现，部分代码如下：



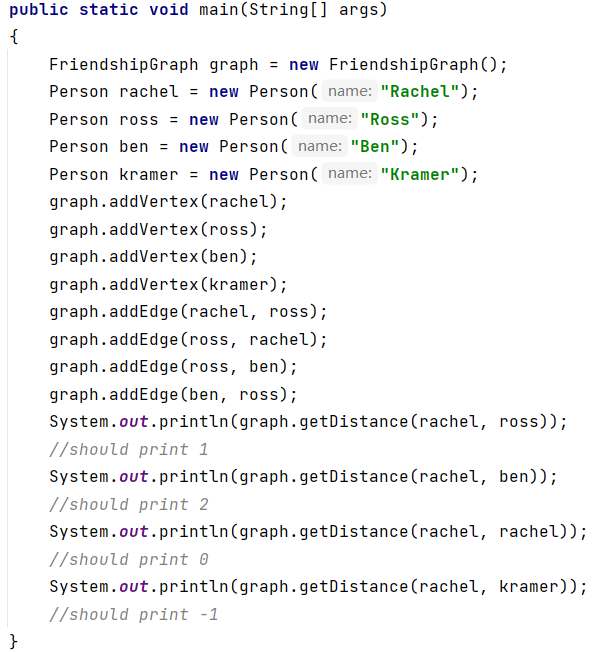
### 设计/实现Person类

Perosn类中包含一个私有成员，代表该成员的名字。还设置了一个getname函数，返回该成员的名字。



### 设计/实现客户端代码main()

将手册里面的代码复制一下就好，代码如下：

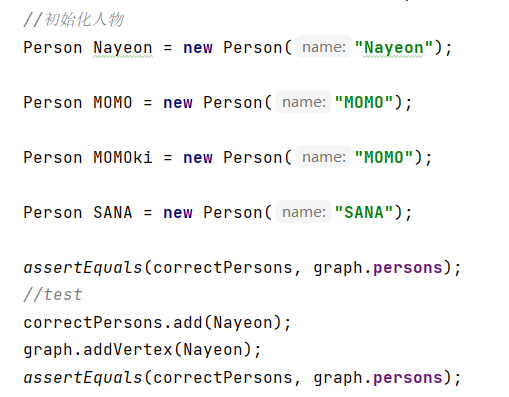


### 设计/实现测试用例

对FriendGraph中的三个函数，测试用例设计如下：

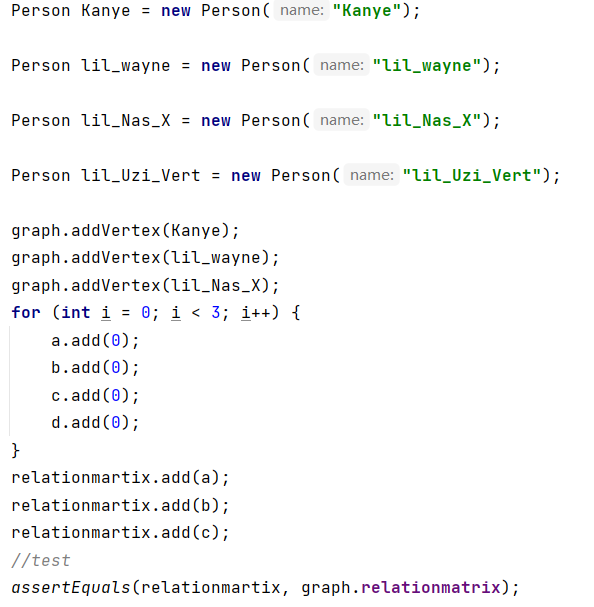
①addVertex的测试：

主要测试addVertex能否正确添加人物元素，设置一个同名人物，判断函数能否返回提示信息。



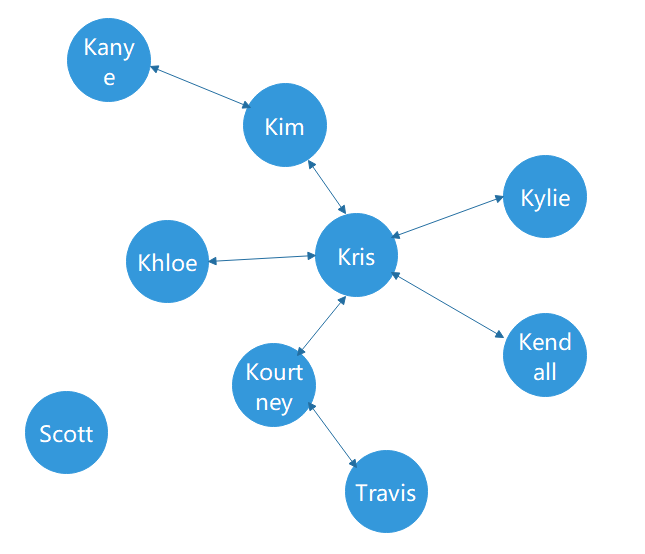
②addEdge的测试：

设计了addEdgeTest。验证添加边后图的邻接矩阵是否符合预期。并在向图中添加新的顶点后再次验证，以确保程序正确无误。



③getDistance的测试：

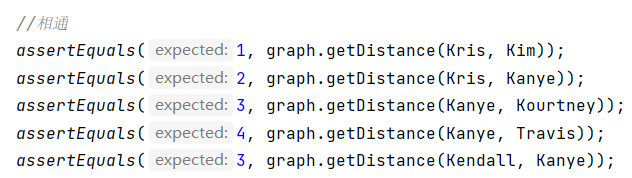
测试如下的关系网：



网络中共有9个人物，下面对以下几种情况的getDistance进行测试：

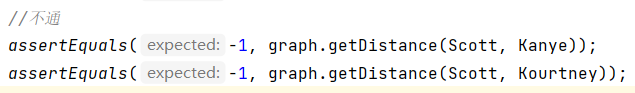
①相通的两点

对以下五组点进行测试，检查其是否能正确返回距离



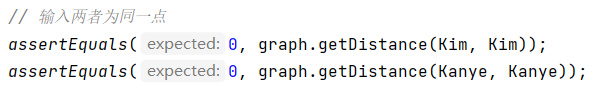
②不相通的两点

对以下两组点进行测试，检查其是否能返回-1



③输入同一点

测试两个Kim，两个Kanye的距离



④输入不存在的点



# 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 |

|  |  |  |  |
| --- | --- | --- | --- |
| 2022-5-1 | 13:45-17:30 | 编写问题1的isLegalMagicSquare函数并进行测试 | 按计划完成 |
| 2022-5-2 | 9:10-9:35 | 完成问题一的部分报告撰写 | 按计划完成 |
| 2022-5-2 | 9:10-9:45 | 分析问题1的generateMagicSquare函数，并撰写部分报告 | 按计划完成 |
| 2022-5-2 | 21:05-22:05 | 完成问题1的generateMagicSquare函数拓展与对应部分的报告撰写 | 按计划完成 |
| 2022-5-3 | 13:45-15:15 | 完成问题2的Problem 3,5及对应部分的报告撰写 | 按计划完成 |
| 2022-5-3 | 18:50-21:00 | 完成问题2的Problem 6及对应部分的报告撰写 | 按计划完成 |
| 2022-5-3 | 19:40-22:30 | 编写问题2的Problem 6代码 | 按计划完成 |
| 2022-5-4 | 8:00-10:30 | 解决问题2，Problem 6之前存在的问题 | 按计划完成 |
| 2022-5-4 | 14:50-15:40 | 完成问题2剩余部分的编程与报告撰写 | 按计划完成 |
| 2022-5-4 | 19:30-21:00 | 完成问题3的部分代码编写 | 按计划完成 |
| 2022-5-5 | 8:00-11:00 | 完成问题3的部分代码编写 | 按计划完成 |
| 2022-5-5 | 8:20-9:15 | 完成问题3的代码编写 | 按计划完成 |
| 2022-5-5 | 9:15-11:00 | 完成问题3及后续报告撰写，并优化代码 | 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 |
| 对于Java的很多函数不熟悉，无法很好的应用 | 参考网上的java教程，学会了很多函数的用法，并成功应用到这次的实验当中 |
| P3当中getDistance需要用到BFS算法，但是忘的差不多了 | 从CSDN上查找了BFS的实现方式，最终成功编写出了算法 |
| P3当中，对于重复人名加入网络的处理，一开始直接使用persons判断是否contain，但在test过程中无法通过 | 经过思考发现，当创建了两个类中名字相同，类名不同的类时，这两个类还是不相等的，因此不可以使用persons.contain来进行判断，最后选择了再创建一个姓名表，直接通过名字来进行检测。 |

# 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）

经验：Java中的很多函数可以简化设计流程，例如可以通过indexof直接返回元素的位置，就可以节省关于元素位置搜索的程序编写。

教训：在进行程序编写时，需要先对整体设计有一个大致的想法，然后再逐一完成各个部分，否则会导致思维混乱，设计出的程序结构性很差。

## 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？

比较对我的口味，java和其他语言相比，内置了很多函数，对于工程方面具有较为明显的优势。

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

本实验我使用的IDEA，个人觉得是一个非常好的IDE，有很多实用的功能，比如自动补写，格式调整等功能；不足方面暂时还没有特别在意的地方。

1. 关于Git和GitHub，是否感受到了它在版本控制方面的价值；

git pull 只会拉取远程仓库中最新的版本，这个最新的版本是和本地仓库中的版本进行比较的，因此在版本控制方面具有十分优越的特性。

1. 关于CMU和MIT的作业，你有何感受；

这两所学校的课程作业都很有特色，特别是MIT的turtle实验，有趣而且能够起到对知识的巩固作用。

1. 关于本实验的工作量、难度、deadline；

本实验的工作量较为合适，一周的实验完全足以完成，在难度方面也较为适中，没有特别难以设计的地方，经过思考基本都可以顺利完成。

1. 关于初接触“软件构造”课程；

个人对软件构造理论以及具体的编程实践感兴趣，能体会到这门课对未来进入计算机hangye1工作有很大的帮助。