

**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 | 路天帅 |
| 学号 | 1163710208 |
| 班号 | 1637102 |
| 电子邮件 | 1321743979@qq.com |
| 手机号码 | 18846178718 |

**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 2](#_Toc506282561)

[3.1.1 isLegalMagicSquare() 2](#_Toc506282562)

[3.1.2 generateMagicSquare() 3](#_Toc506282563)

[3.2 Turtle Graphics 4](#_Toc506282564)

[3.2.1 Problem 1: Clone and import 4](#_Toc506282565)

[3.2.2 Problem 3: Turtle graphics and drawSquare 4](#_Toc506282566)

[3.2.3 Problem 5: Drawing polygons 4](#_Toc506282567)

[3.2.4 Problem 6: Calculating headings 4](#_Toc506282568)

[3.2.5 Problem 7: Personal art 5](#_Toc506282569)

[3.2.6 Submitting 5](#_Toc506282570)

[3.3 Social Network 5](#_Toc506282571)

[3.3.1 设计/实现FriendshipGraph类 5](#_Toc506282572)

[3.3.2 设计/实现Person类 5](#_Toc506282573)

[3.3.3 设计/实现客户端代码main() 6](#_Toc506282574)

[3.3.4 设计/实现测试用例 6](#_Toc506282575)

[3.4 Tweet Tweet（选作，额外记分） 6](#_Toc506282576)

[4 实验进度记录 6](#_Toc506282577)

[5 实验过程中遇到的困难与解决途径 6](#_Toc506282578)

[6 实验过程中收获的经验、教训、感想 7](#_Toc506282579)

# 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

首先下载Java的JDK8，eclipse，还有git bash，然后按照MIT网站上的要求配置Java环境。之后在配置git时发现我的路径里有中文（我的名字的中文），因此在助教学长和百度的帮助下把中文改成了英文，才配置好git。但之后又发现eclipse打不开了，说是路径错误，因为eclipse存在了我的这个用户文件夹下，所以改完名字就无法打开了。经过百度搜索无解后又把eclipse卸载重装，最后才配置好环境。

Lab1-1163710208

# 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

设计Java程序按照MagicSqure的定义首先查看该矩阵是否符合其定义，即数据不能有小数负数，是否是矩阵等，然后来检查每一行每一列和对角线的数据的加和是否相等。最后返回判断结果。

### isLegalMagicSquare()

1. 定义一个一维数组来用来存储从文件中按行读出的字符串。
2. 然后通过语句

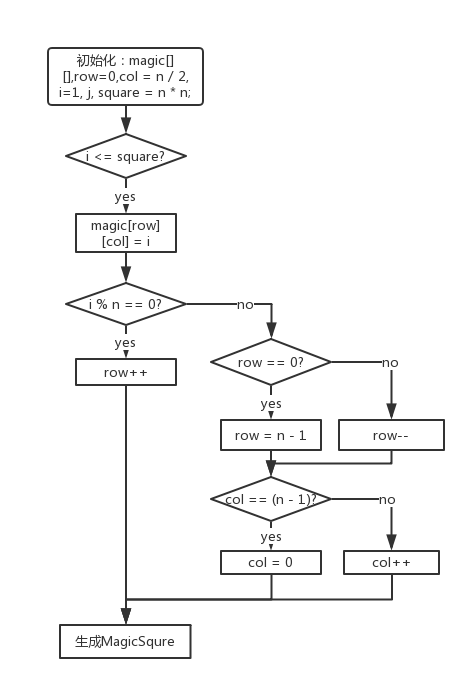
String[] m = str[j].split("\t");

ms[j][k] = Integer.valueOf(m[k]);

来按照制表符来切割字符串，并转换成int类型整数，再定义一个二维数组用来存储得到的整数。

3.得到矩阵后，先求出矩阵第一行数据的和。然后再与剩余的行和列以及对角线元素的和来比较，若不相等则返回false，最后如果都相等则返回true.

### generateMagicSquare()



1. 首先判断传的参数n是否是正数且是奇数，若不是则打印相应提示信息并退出。
2. 然后按照流程图的算法来生成MagicSqure
3. 最后通过BufferedWriter写入文件。

## Turtle Graphics

该问题要考察我们是否会调用turtle类里的方法来作图，同时也要使用一些Java标准类库的方法来实现编程。也考察了一些基本编程技巧。

### Problem 1: Clone and import

在 Problem 1: Clone and import 中无法连接 MIT 的 didit 服务器，所以要从 https://github.com/rainywang/Spring2018\_HITCS\_SC\_Lab1/tree/master/P2 获取代码。

### Problem 3: Turtle graphics and drawSquare

首先根据Turtle.java发现有forward方法，功能是向前移动指定距离，还有turn方法，功能是向右转指定的角度。通过尝试发现turtle最开始面向正北方向，所以先让乌龟向前走sideLength距离，然后转90度，在走sideLength距离再转。以此类推最后回到起点就画出了矩形。

### Problem 5: Drawing polygons

该问题要画正多边形，因此可调用写好的calculateRegularPolygonAngle方法，来得到目标正多边形的内角度数，然后让乌龟每走一段距离就转动180-内角度数，也就是内角的外角的度数。仿照画正方形的过程就可以得到正多边形。

### Problem 6: Calculating headings

首先定义一个double型的List数组，用来存储返回的角度。然后取出传过来的数组的第一个和第二个点，即乌龟的位置和第一个目标点，然后调用写好的calculateHeadingToPoint方法来求出第一个角度，然后用循环来依次求出其他点的转动角度。

### Problem 7: Personal art

首先调用乌龟的color算法来将笔的颜色变为红色，然后仿照画正方形的方法来画一个五角星，转动18度，再循环20次，最后构成一个抽象的美术图形。

### Submitting

1. 在git Bash用命令行打开实验所在的目录
2. 使用git add .和git commit等命令来将实验内容加入仓库
3. 最后使用git remote add origin [git@github.com:ComputerScienceHIT/Lab1-1163710208.git](mailto:git@github.com:ComputerScienceHIT/Lab1-1163710208.git)命令将项目加入到实验的Lab1-1163710208

## Social Network

该任务考察我们对有向图无向图存储的处理能力，以及Junit测试的方法和分模块编程的思想。

### 设计/实现FriendshipGraph类

1. 新建一个二维数组用来存储关系图，再定义一个Person类型的ArrayList用来存储用户。
2. addVertex函数直接调用Person的.add()方法。
3. 然后对于addEdge，直接用循环获取矩阵的横纵坐标并将其赋值为1.
4. 对于getDistance方法，使用floyd算法来算出点到个位置的最短路，然后取出需要得的数值。

### 设计/实现Person类

定义一个String类行型的name，然后在构造器内将传过来的string赋给name。

### 设计/实现客户端代码main()

首先定义一个FriendshipGraph类型的graph然后定义四个Person类型的人，然后调用graph的addVertex方法来将四个人加入graph，然后再调用graph的addEdge来将好友关系图加入graph，最后调用getDistance方法来求出所测试的人之间的距离。

### 设计/实现测试用例

首先定义一个FriendshipGraph类型的graph然后定义四个Person类型的人，然后通过语句assertEquals(rachel,graph.person.get(0)); 来测试人物是否加入到graph中。然后再调用graph的addEdge来将好友关系图加入graph，然后调用graph的addVertex方法来将四个人加入graph，然后通过语句assertEquals(1,graph.g[0][1]); 来测试图是否创建成功。最后调用getDistance方法来求出所测试的人之间的距离。再通过语句assertEquals(2,graph.getDistance(PersonA, PersonB)); 来测试A和B的距离。

## Tweet Tweet（选作，额外记分）

自行组织目录结构。

# 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 |
| 2018-03-02 | 14:00-15:30 | 编写P1 | 按计划完成 |
| 2018-03-05 | 18:00-22:00 | 编写P2 | 延期1小时完成 |
| 2018-03-08 | 18:00-22:00 | 编写P3的FriendshipGraph类和Person类 | 延期1小时完成 |
| 2018-03-10 | 18:00-22:00 | 编写P4的Junit测试 | 遇到困难，未完成 |
| 2018-03-12 | 13:45-3:30 | 编写P4的Junit测试 | 按计划完成 |

# 实验过程中遇到的困难与解决途径

实验环境的配置有很多意想不到的问题，电脑的路径有中文真的是很不方便，通过学长和百度的帮助才解决，所以以后所有文件名都不要起名为中文，然后MIT的实验要求阅读起来有困难。然后通过谷歌翻译等的帮助下才把实验要求弄懂。

# 实验过程中收获的经验、教训、感想

通过本次实验我对Java语言的理解更加深入，也熟悉了Java的一些操作。教训就是以后电脑的所有应户名，文件名都不要起中文名。我觉得Java编程语言还是比较适合我的胃口，Eclipse也很赞，Git有点不好搞，git Bash界面也不够友好，CMU和MIT的作业很有挑战性，很有针对性。对于Java没学好的我来说，作业完成起来还是很有难度的。