

**2021年春季学期  
计算学部《软件构造》课程**

**Lab 1实验报告**

|  |  |
| --- | --- |
| 姓名 | 任焕臣 |
| 学号 | 1190200827 |
| 班号 | 1903003 |
| 电子邮件 | 405926869@qq.com |
| 手机号码 | 18346504604 |

**目录**

[1 实验目标概述 1](#_Toc29325521)

[2 实验环境配置 1](#_Toc29325522)

[3 实验过程 1](#_Toc29325523)

[3.1 Magic Squares 1](#_Toc29325524)

[3.1.1 isLegalMagicSquare() 1](#_Toc29325525)

[3.1.2 generateMagicSquare() 1](#_Toc29325526)

[3.2 Turtle Graphics 1](#_Toc29325527)

[3.2.1 Problem 1: Clone and import 2](#_Toc29325528)

[3.2.2 Problem 3: Turtle graphics and drawSquare 2](#_Toc29325529)

[3.2.3 Problem 5: Drawing polygons 2](#_Toc29325530)

[3.2.4 Problem 6: Calculating Bearings 2](#_Toc29325531)

[3.2.5 Problem 7: Convex Hulls 2](#_Toc29325532)

[3.2.6 Problem 8: Personal art 2](#_Toc29325533)

[3.2.7 Submitting 2](#_Toc29325534)

[3.3 Social Network 2](#_Toc29325535)

[3.3.1 设计/实现FriendshipGraph类 2](#_Toc29325536)

[3.3.2 设计/实现Person类 2](#_Toc29325537)

[3.3.3 设计/实现客户端代码main() 2](#_Toc29325538)

[3.3.4 设计/实现测试用例 3](#_Toc29325539)

[4 实验进度记录 3](#_Toc29325540)

[5 实验过程中遇到的困难与解决途径 3](#_Toc29325541)

[6 实验过程中收获的经验、教训、感想 3](#_Toc29325542)

[6.1 实验过程中收获的经验和教训 3](#_Toc29325543)

[6.2 针对以下方面的感受 3](#_Toc29325544)

# 实验目标概述

这次实验通过解决求解三个问题，训练基本的Java编程技能，并且培养阅读已有的代码框架并且根据需求补全代码的能力，能够为所开发的代码编写基本的测试程序并完成测试。学会使用GIT。

# 实验环境配置

到java官网以及eclipse官网下载了jdk以及eclipse

https://github.com/1190200827?tab=repositories

# 实验过程

## Magic Squares

主要模块分为两个，一个是将文件读取，并存储成矩阵，在这个过程中需要进行合法性的判断，第二个就是判断是否为幻方矩阵，先按照行比较，再按照列来比较，最后再和两个对角进行比较。

### isLegalMagicSquare()

1读取文件，用二维数组储存读入的数据

       首先利用FileReader对象读取文件，然后对每一行判断是否含有非法字符不是0~9或者分隔符‘\t’或者空格，若非法，返回false，否则利用取出矩阵的数字用二维数组保存。

2判断是否满足幻方定义

对于存好的矩阵，行列不相等返回false，否则对每列每行对角线分别求和，都一致才返回true，否则返回false。

### generateMagicSquare()

17   24   1     8     15

23   5     7     14   16

4     6     13   20   22

10   12   19   21   3

11   18   25   2     9



每个线上的数字沿着斜对向上角递增。这种构造方法名叫德拉鲁布(De laloubere)算法，可以构造任何一个正奇数阶幻方

## Turtle Graphics

完善TurtleSoup.java文件，这个实验是为了训练使用java自带的画图库。

### Problem 1: Clone and import

使用gitclonehttps://github.com/rainywang/Spring2021\_HITCS\_SC\_Lab1/tree/master/P2.git指令获取该任务代码



使用git init命令在一个空文件夹建立仓库，接下来用git add xxx、git commit -m “xxx”等命令可以在本地仓库添加本地的开发项目

### Problem 3: Turtle graphics and drawSquare

这个问题要我们补全drawSquare函数，turtle.forward方法表示前进一定长度，然后每次转90度，循环4次即可画出一个正方形

### Problem 5: Drawing polygons

 double calculateRegularPolygonAngle(int sides)；

这个方法是根据正多边形的边数计算出多边形的内角

依照公式double angle = 360 – 180/sides;计算即可

int calculatePolygonSidesFromAngle(double angle)；

  这个方法是根据正多边形的内角计算出边数， 角度范围为0~360

按照公式int sides = (int)Math.rint(180/(360 – angle));计算即可

void drawRegularPolygon(Turtle turtle, int sides, int sideLength)；

 这个方法是根据给定的多边形边数与边长画出一个正多边形，边数大于2，边长大于0

 调用calculateRegularPolygonAngle(int sides)函数得到角度，循环画边即可

### Problem 6: Calculating Bearings

double calculateBearingToPoint(double currentBearing, int currentX, int currentY, int targetX, int targetY) ;

这个方法要求用给定的参数求出从当前方向转到target点与current点连线顺时针需要旋转的角度。

       将角度都化为[0,360)可以更简便地进行计算

以x轴为极轴，current点为极点建立极坐标轴，然后利用Math.atan2(y,x)函数计算出target点与current点连线与x轴的夹角（极角），并将currenBearing与求得的极角转化为>=0且＜360°，从而得到它们的差，即为所求

calculateBearings(List<Integer> xCoords, List<Integer> yCoords);

根据参数计算出每次要旋转的角度

且X列表元素个数要与y列表元素个数相等

以第一次向北，接下来每一次都根据前一次的角度以及当前点与前一点的连线极角（极点为前一点）来计算，返回角度列表即可

### Problem 7: Convex Hulls

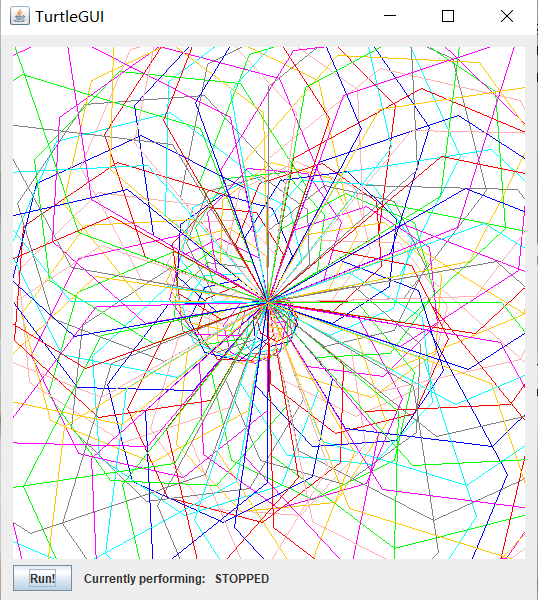
1.首先在points中找出一个对照点，放入到栈中，这个对照点就是在点集的最下，最左的那个点，这个点为极点，极轴平行于x轴建立极坐标轴。

2.对剩下的点按相对于对照点的极角从小到大排序，

3.把输入的点集的前两个点放入到栈中，然后依次扫描剩余的点，把不是凸包顶点的点从栈中剔除，沿逆时针方向通过凸包时，在每个顶点处应该左转，因此，如果发现在顶点处没有左转，就可以从栈里弹出了，把向左转的点加入到栈中

最后栈里面保存的就是凸包的顶点，将这个栈的内容全都添加到convexHull集合中即可

### Problem 8: Personal art



### Submitting

如何通过Git提交当前版本到GitHub上你的Lab1仓库。

## Social Network

在人群中通过关系将他们连接起来，且能在人群中添加新的人，能在人群中指定的某两个人之间建立关系，并能实现指定某两个人的关系长度

### 设计/实现FriendshipGraph类

用一个二维数组存储人们之间的关系，并为每个人编上代号，按照代号在数组中寻找两个人的关系，每个人的名字与代号一一对应，所以不应该有同名的人，否则会报错。

### 设计/实现Person类

**public class** Person {  
 String **name**;  
 **int id**;  
 Person **next**;  
 **public** Person(String name){  
 **this**.**name**=name;  
 **this**.**next**=**null**;  
  
 }  
  
}

### 设计/实现客户端代码main()

参照手册

### 设计/实现测试用例

addvertex函数测试正常添加一个人与重复添加一个人；

addEdge函数测试两个人正常建立关系、一个人与自己建立关系,与不存在的人名建立关系；

getDistance函数要测试不可达关系、自身到达自身的距离、一个可达关系的距离、与不存在的人的距离。

# 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。

每次结束编程时，请向该表格中增加一行。不要事后胡乱填写。

不要嫌烦，该表格可帮助你汇总你在每个任务上付出的时间和精力，发现自己不擅长的任务，后续有意识的弥补。

|  |  |  |  |
| --- | --- | --- | --- |
| 日期 | 时间段 | 任务 | 实际完成情况 |
| 2021-05-16 | 19:00-20:30 | 编写问题1函数并进行测试 | 按计划完成 |
| 2021-05-17 | 19:00-21:30 | 研究P2 | 遇到困难 |
| 2021-05-18 | 19:00-20:30 | 解决P2 | 完成 |
| 2021-5-20 | 19:00-20:30 | 解决P3 | 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 |
| 不会java | 看mooc |
| idea不会 | 上bi站和csdn找资料 |
| Git不会 | 问室友 |

# 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

## 针对以下方面的感受

1. Java编程语言是否对你的口味？是
2. 关于Eclipse IDE；不好用，idea天下第一
3. 关于Git和GitHub；太难用了
4. 关于CMU和MIT的作业；难
5. 关于本实验的工作量、难度、deadline；适中，如果没有自动机考试的话（呜呜呜）
6. 关于初接触“软件构造”课程；

很有趣