

**2023年春季学期  
计算学部《软件构造》课程**

**Lab 1实验报告**

|  |  |
| --- | --- |
| 姓名 | 董禹松 |
| 学号 | 2021112213 |
| 班号 | 2137101 |
| 电子邮件 | [826482017@qq.com](mailto:826482017@qq.com) |
| 手机号码 | 18745029836 |

**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

# 实验环境配置

本次实验运行环境为IntelliJ IDEA Community Edition 2022.3.2，测试环境为junit-4.13.1.jar

在配置junit时遇到了idea无法下载junit5的问题，最后改为使用junit4问题得以解决

GitHub仓库的地址为：https://github.com/ComputerScienceHIT/HIT-Lab1-2021112213.git

# 实验过程

## Magic Squares

判断给出的文件中的矩阵是否为幻方，并生成一个幻方。

### isLegalMagicSquare()

1. 读取文件中的内容，并将其转换为二维数组并用正则表达式判断数组中的元素是否为正整数；
2. 检查数组是否为空或不是方阵；
3. 计算数组中每行、每列和两条对角线上的数字之和，并比较它们是否相等；最后，返回比较的结果。

### generateMagicSquare()

1.定义一个N×N的二维数组，用来存储幻方的数字；

2.初始化数组中的所有元素为0，表示空白格子；

3.定义两个变量row和col，分别表示当前填写数字所在的行和列，初始值为0和N/2（向下取整）；

4.定义一个变量i，表示当前要填写的数字，初始值为1；

5.使用一个循环结构，从1到N×N遍历i的值；

6.在循环中，先将i填入数组[row][col]中，然后更新row和col的值；

7.首先判断(i%n)是否等于0，如果是则row加1，即向下移动一格。否则，按照右上斜行的规则移动位置：row减1，col加1；如果row小于0，则将row设为N-1；如果col大于等于N，则将col设为0；

8.重复步骤7直到循环结束。

## Turtle Graphics

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

### Problem 1: Clone and import

进入项目目录，右键选择 Git Bash Here1。

输入命令 git init 在当前目录下创建一个空的Git仓库1。

输入命令 git clone https://github.com/... (替换为要获取的GitHub项目的URL) 将GitHub项目拉取到本地2。

输入命令 git add . 将当前路径下的所有文件/文件夹添加到Git仓库中13。

输入命令 git commit -m "提交时的备注" 将代码提交到本地仓库中13。

在GitHub上新建一个远程仓库，并设置好SSH密钥3。

输入命令 git remote add origin https://github.com/... (替换为新建的远程仓库的URL) 关联本地和远程仓库3。

输入命令 git push -u origin master 将本地代码推送到远程仓库3。

### Problem 3: Turtle graphics and drawSquare

使turtle向前走一段距离后转向90度，转四次即可生成一个正方形

### Problem 5: Drawing polygons

1.根据边数计算内角的度数

return 180.0 - 360.0 / sides;

2. 使turtle向前走一段距离后转向多边形外角的度数，转向次数为边数

for(int i = 0; i < sides; i++) {  
 turtle.forward(sideLength);  
 turtle.turn(180 - angle);  
 }

### Problem 6: Calculating Bearings

1. 计算弧度并转换为角度

double angle = Math.*toDegrees*(Math.*atan2*(dy, dx));

2. 将角度转换为方位角（以y轴为0度，顺时针方向）

double bearing = (90 - angle + 360) % 360;

3.计算相对方位角

double relativeBearing = (bearing - currentBearing + 360) % 360;

### Problem 7: Convex Hulls

1.先确定边界上的点v1和与下一个点v2

2.在点集里去寻找下一个点v3，使得v1 v3 v2满足CCW

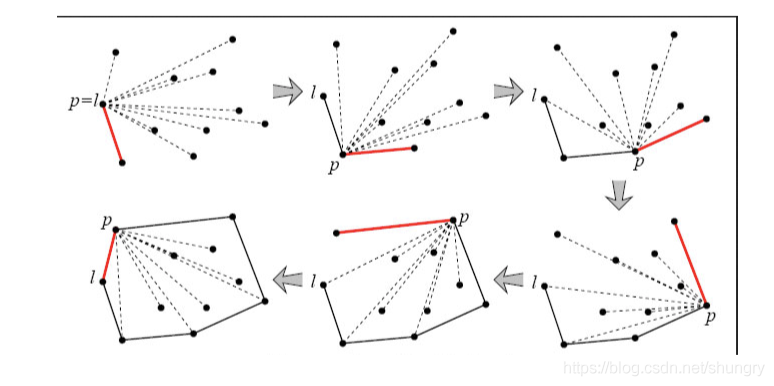
3.如果满足，这就说明v3是更外围的点

4.把v3的值覆盖v2，即v2 = v3

5.重复的3步，直到所有点都访问过，即为找到了下一个最外边的点

6.将v2的值覆盖v1，即v1 = v2

7.重复第2步，直到下一个点回到起点



### Problem 8: Personal art

1.画一个正方形

for (int i = 0; i < 4; i++) {  
 turtle.forward(50);  
 turtle.turn(90);  
}

2. 画一个圆

for (int i = 0; i < 360; i++) {  
 turtle.forward(1);  
 turtle.turn(1);  
}

1. 画一个三角形

turtle.forward(100);  
for (int i = 0; i < 3; i++) {  
 turtle.forward(80);  
 turtle.turn(240);  
}

### Submitting

git add src

git commit -m“latest”

git push “url地址”

## Social Network

绘制一个无向图，并将顶点都改为person类，就得到了一个社交网络图

### 设计/实现FriendshipGraph类

首先定义数据类型：用列表来存储节点，用邻接矩阵来存储边

public FriendshipGraph(){  
 vertex = new ArrayList<>();  
 matrix = new int[100][100];  
 vCount = 0;  
 edgeCount = 0;  
}

然后构造函数实现添加节点、添加边、计算两点之间的距离等操作

public void addVertex(Person p)

public void addEdge(Person p1, Person p2)

public int getDistance(Person p1, Person p2)

### 设计/实现Person类

给出你的设计和实现思路/过程/结果。

用String类型保存name

构建函数获取用户name，判断两个名字是否相等

### 设计/实现客户端代码main()

客户端代码与给出的代码保持一致

### 设计/实现测试用例

对添加点、添加边、求两点之间的距离进行测试

# 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。

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 |
| 2023-03-07 | 18:30-19:30 | 编写问题1的isLegalMagicSquare函数并进行测试 | 按计划完成 |
| 2023-03-07 | 19：40-22.00 | 学习git及GitHub的使用，并将P1上传 | 按计划完成 |
| 2023-3-10 | 8.00-9.00 | 编写问题三的Person类 | 按计划完成 |
| 2023-3-10 | 9.00-11.00 | 编写问题三的friendGroup类 | 延期两小时完成 |
| 2023-3-13 | 1.00-5.00 | 编写实验二convexHull函数并进行测试 | 按计划完成 |
| 2023-3-13 | 6.00-8.00 | 编写实验二calculateBearings函数并进行测试 | 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 |
| 在配置junit时遇到了idea无法下载junit5的问题 | 改为使用junit4问题得以解决 |
| 编写实验二convexHull函数，复现代码遭到困难 | 通过在论坛查询解决问题 |
|  |  |

# 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）

1.学习了抽象数据类型（ADT）的设计、规约、测试和面向对象编程（OOP）技术的实现。

2.掌握了Java的泛型相关语法以及泛型设计思想。

3.熟悉了EclEmma等工具的使用，提高了代码覆盖度和质量。

4.增强了调试能力和解决问题能力，体会到了软件工程的重要性。

## 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？

Java很适合我。

优势：

1.语言简单易用，脱胎于C语言，但更容易学习和编写。

2.面向对象的程度高，支持封装、继承、多态等特性。

3.安全性高，没有指针操作，有自动垃圾回收机制

不足：

1.效率低，运行速度慢于C/C++等语言。

2.内存消耗大，需要类库的支持。

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

IntelliJ IDEA优势：

拥有光滑、精致、流畅的用户界面。

强大的整合能力，支持Git、Maven、Spring等。

提示功能快速、便捷、范围广。

好用的快捷键和代码模板。

精准搜索和重构功能。

不足：

创建索引时可能会卡顿，对大型项目不友好。

不支持多个项目同时打开，没有workspace的概念。

1. 关于Git和GitHub，是否感受到了它在版本控制方面的价值；

Git是一种分布式的版本控制系统，能够有效地管理和追踪代码的变化。

GitHub是一个基于Git的在线代码托管平台，能够让开发者共享、协作和发布自己的项目。

1. 关于CMU和MIT的作业，你有何感受；

学会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

1. 关于本实验的工作量、难度、deadline；

本实验工作量较大、难度适中、完成时间较为充足

1. 关于初接触“软件构造”课程；

是一门很有用的课，受益匪浅