编号



面向对象程序设计I课程设计报告

二级学院 计算机科学与工程学院

专 业 计算机科学与技术

班 级 计算机五班

学生姓名 陈松 学号 11703990535

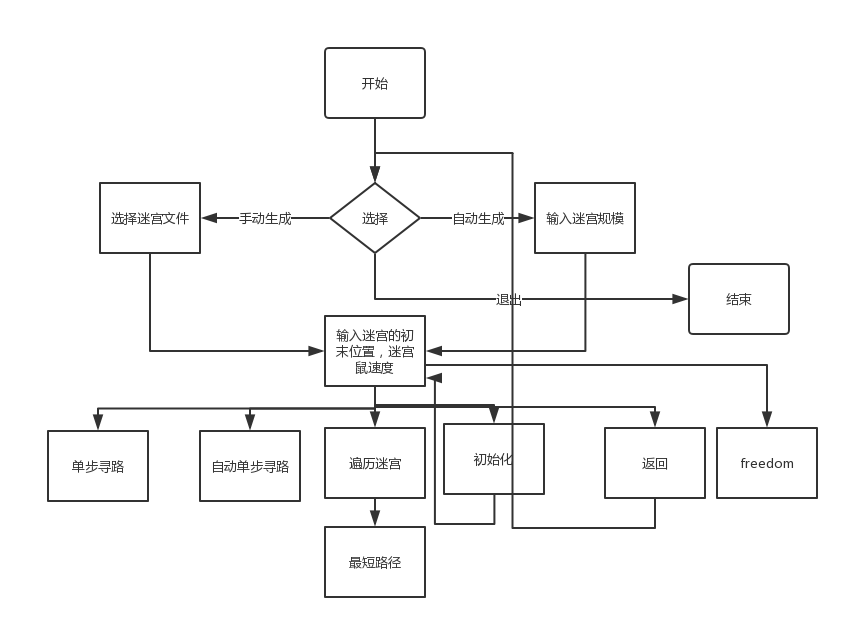
指导教师 刘恒洋

时 间 2018年8月

## 功能说明

概述:打开程序，会弹出初始窗口，让你选择进入（手动生成模块，自动生成模块），并且内置退出键可退出系统。手动生成模块会让使用者在文件中选择迷宫的TXT文件（打开会直接出现在迷宫TXT文件目录中，若未选择或选择了错误文件，系统将不会显示迷宫地图，你可以重新点击选择），自动模块可根据使用者需求生成迷宫（本迷宫采用方形制，即输入的为方形边长，输入必须为正整数，不然会弹出信息框提示重新输入），迷宫生成之后，会进行初始条件设置，迷宫鼠的初末位置（可自己输入 注：输入不为整数或超过最大值，会弹出信息框提示重新输入。也可以点击迷宫的一处，获取位置信息，再次点击输入框将会将位置信息复制在框内），移动速度（系统内置自动寻路按钮，可选择速度）。之后你可以点击按钮（freedom），可以通过上下左右来控制迷宫鼠移动，也可单步寻路，自动寻路找到通往终点的路径，也可以遍历得到最短路径。也有返回键可返回初始窗口（初始窗口：进行自动生成迷宫，手动生成迷宫，退出系统得选择）。

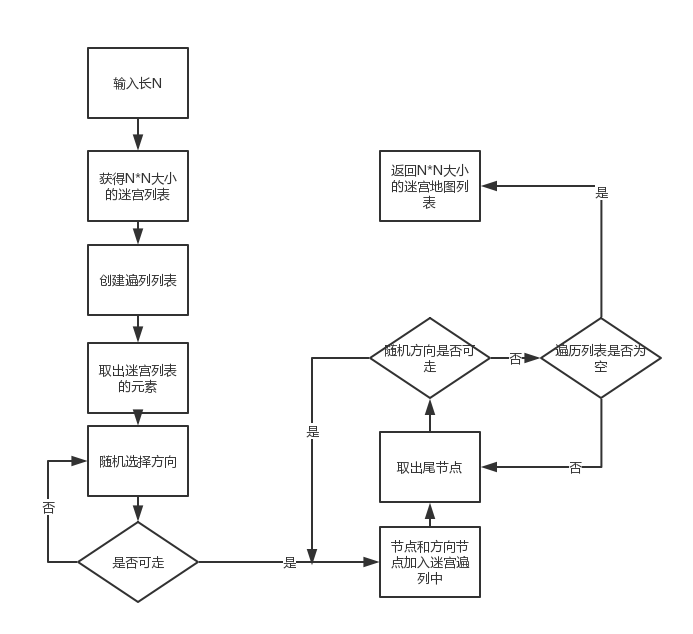
1.1.2 功能流程图



## 1.2 算法说明

1.2.1 Recursive division method（递归除法：深度优先） 自动生成迷宫算法

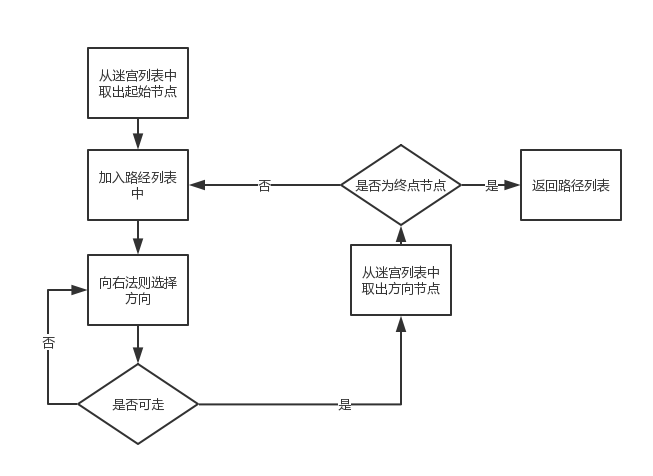
可随意输入迷宫的长和宽（本程序界面限定，只输入长(N)即可，宽默认和长相等），但数字必须为大于0的正整数，便会返回一个有N\*N个节点（节点为自写的Mpoint类，有上下左右属性，默认为1，即为墙，还有位置，是否被拜访属性）的列表，采用深度优先的方法，即从列表的0坐标节点开始，随机选择一个方向，确定可走后（判断依据：该方向的结点是否被拜访过，是否没有节点，如过是重新换个方向） 将0坐标和选择方向的第一个节点加入遍历列表，并将两结点相关属性改成0，即通路（如果向下，就将0的down和方向节点的up改成0）,然后取出遍历列表的尾节点（即方向节点），按上述方式判断，依次下去，若遇到节点的所有方向都不能走的情况（对这进行了改进，若遇到这种情况，便会有十分之一的机会随机打通一个方向的路，即可避免迷宫只有一条通路的情况），便移除尾节点，再取出新的尾节点判断，列表节点都被拜访过后，遍历列表便会一直移除尾节点，直至遍列列表为空。



1.2.2 深度优先-单步寻路算法

根据迷宫地图，取出起始节点加入路径列表中。

以到达终点结点，作为循环结束的标志。取出路径列表尾节点，采用向右法则，以右，上，左，下的顺序选择方向，若有方向可走，将方向节点加入列表中（作为列表新的尾节点），若无方向可走，移除该节点，结束此次循环，取出新的尾节点进行下一次循环。



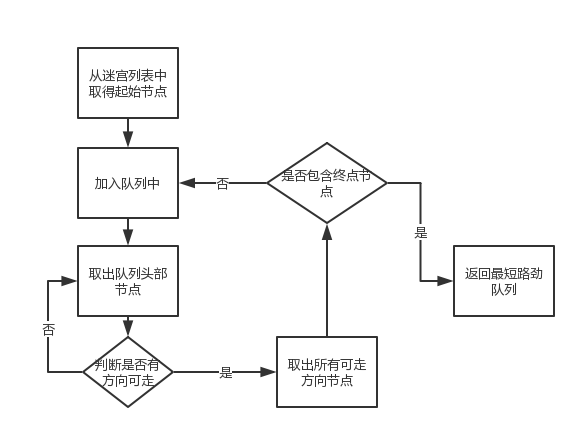
1.2.3 广度优先-迷宫遍历，最短路径算法

第一步：从迷宫列表中取出起始节点加入遍列队列中（先进的先出）中

第二步：以到达终点结点作为结束标志，做循环

第三步：从队列中取出节点，找出所有可走的方向节点加入队列中。

注：节点属于Mpoint类，有一个Mpoint pre成员，记录的是能得到该节点的节点，且最先得到终点节点的一定为最短路径，即随着循环完成后可逆向寻得最短路劲



## 功能设计

整个系统有三个类，其一为生成界面得Main主类，其二是生成迷宫地图列表的MazeMo类，其三是定义列表节点的Mpoint类。

首先根据对模式（自动和手动）的选择，弹出除迷宫生成外其余功能一致的界面

Txt文件选择生成模块:对选择得到得文件分析，形成一个个节点（Mpont类），然后用栈来储存这些节点，调用 setMazePane();方法画出迷宫。

自动生成模块：调用MazeMo类的getArraylist()方法得到迷宫栈，然后调用setMazePane()方法画出迷宫。

初始条件按钮模块:弹出初末位置，速度属性设置界面。属性正确设置后，其他功能键才能响应。（初始化按钮响应后此按钮才能再此被响应）

Freedom按钮模块：采用键盘响应事件（即键盘的上下左右键来控制小球的移动），且为防止按钮影响键盘事件的焦点集中在小球上，用监听事件（动画）让焦点在freeMouse按钮响应后，一直集中在小球上，再次响应时，取除再小球上的焦点，并解封单步，遍历功能

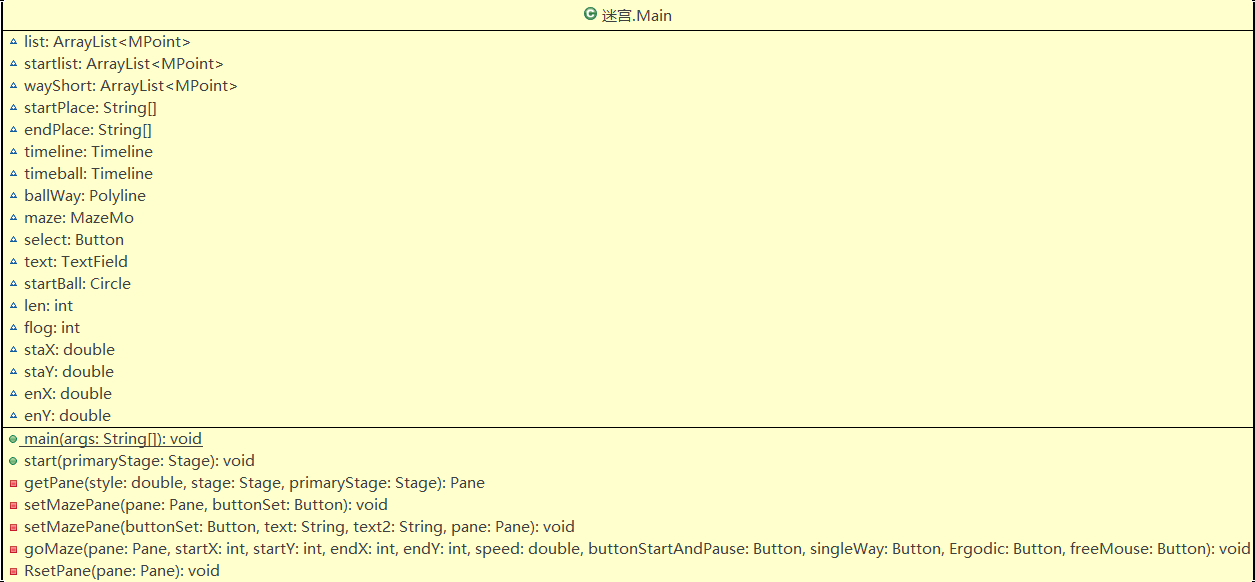
单步寻路模块：有手动单步功能和自动单步功能，均采用栈的出栈方式来达到下一个位置（节点）手动为按钮响应，即点击按钮一下，移动一个位置，自动为按钮控制动画事件的开始与暂停，两者可相互交换进行，即自动暂停后可手动。

遍历最短路径模块:遍历按钮响应后，采用广度优先法，遍历迷宫，找到最短路径，并将存有最短路径的栈传给Main类成员shortWay,并解封最短路径按钮

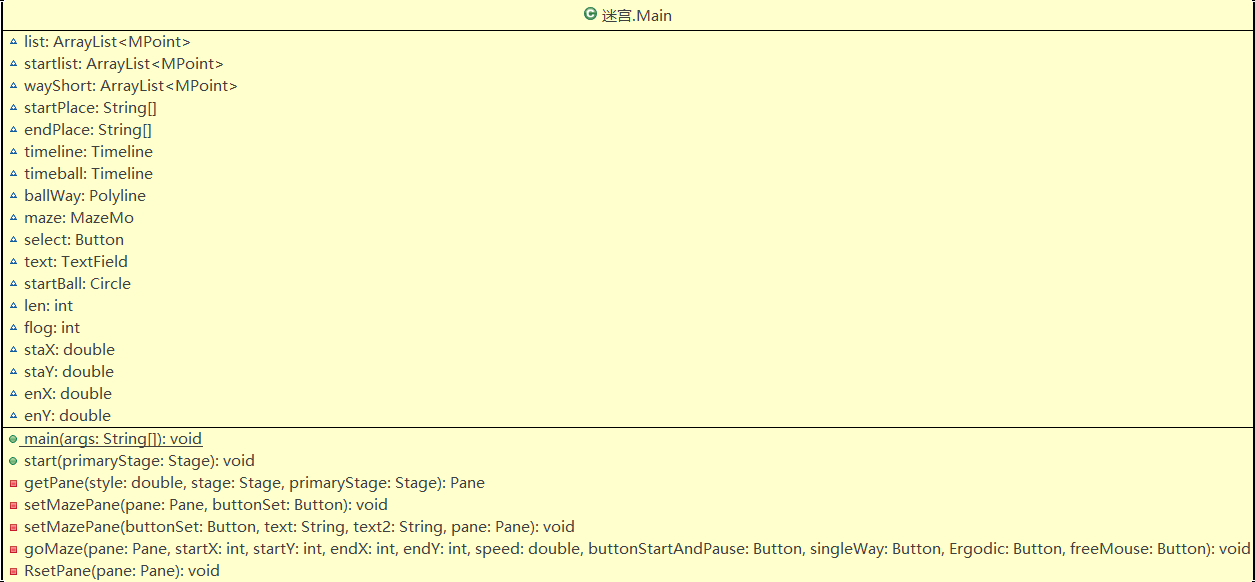
初始化返回模块：封印所有寻路功能，并采用覆盖法，即用一矩形将图覆盖，再将迷宫画上去，返回即返回上一模块。

## 1.3.1 系统静态模型

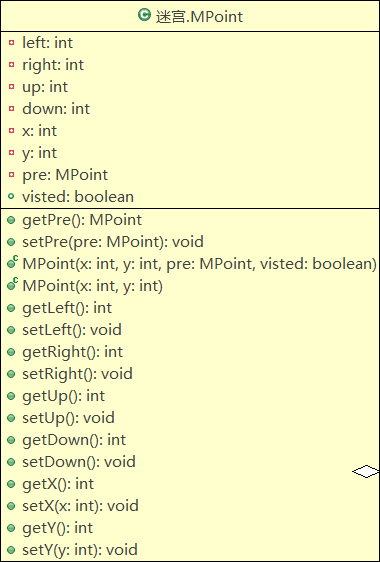
Main类:



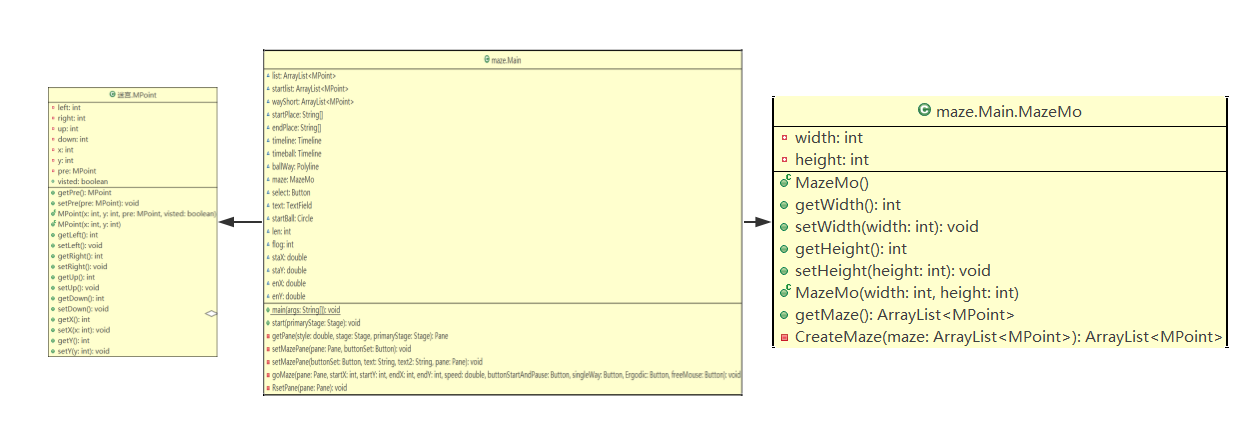
Main的内置类MazeMo类：



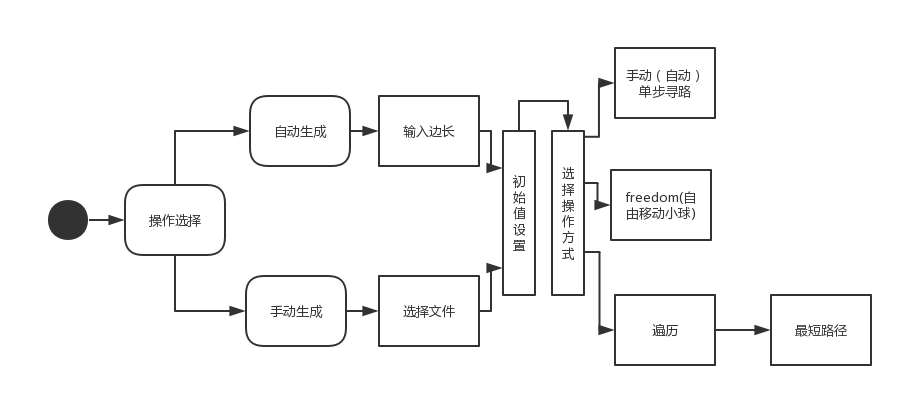
Mpoint类:



时序图:



1.3.2 **系**统动态模型



1.4 系统测试

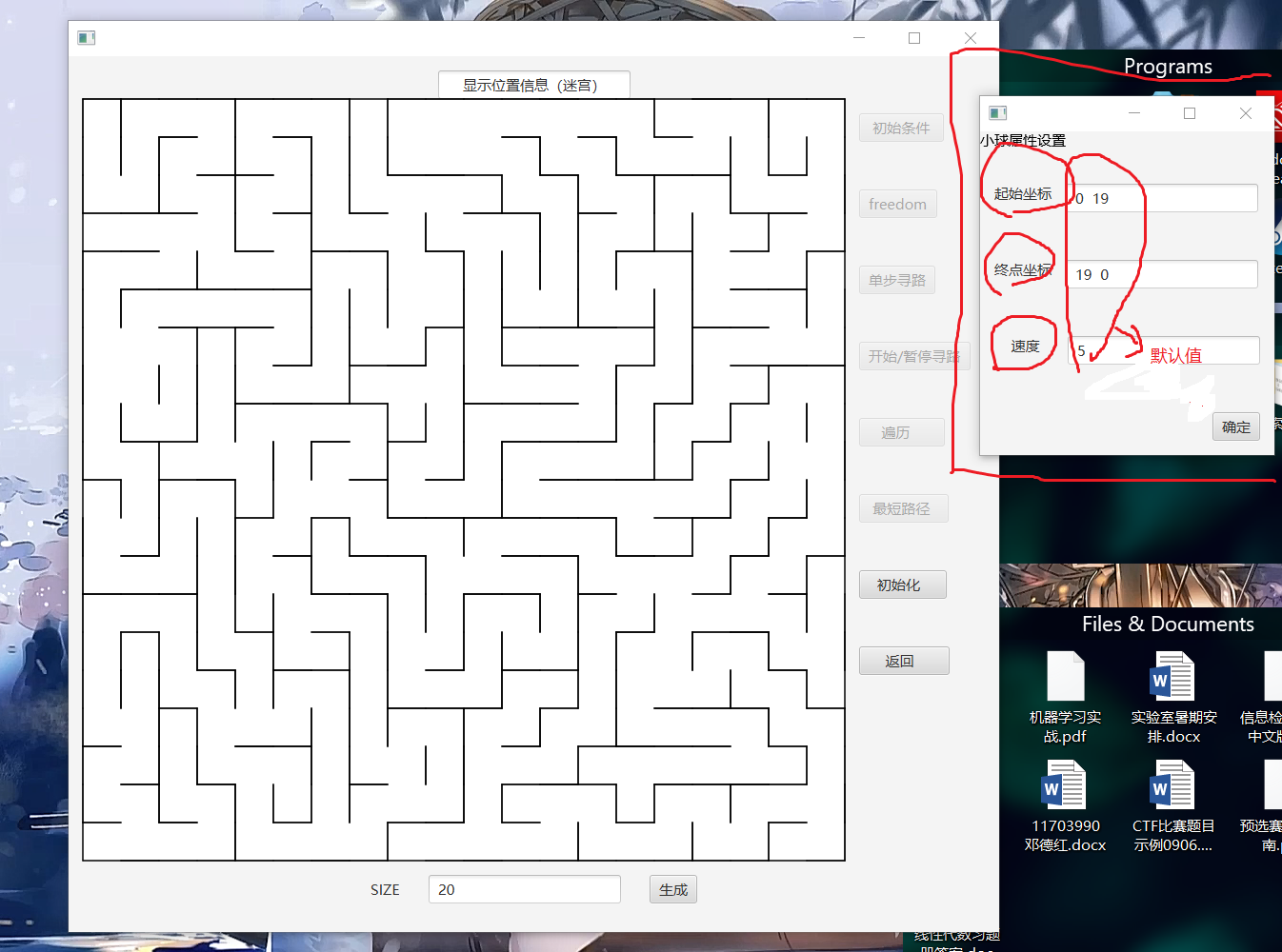
|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 测试用例编号 | 000 | | 版本号 | | 1.0 |
| 测试环境 | Window10 | | | | |
| 用例名称 | 迷宫 | | | | |
| 前提条件 | （手动）选择预定文件夹的，（自动）大于0的整数 | | | | |
| 测试步骤 | 逐一测试功能 | | | | |
| 输入数据 | 20 | | | | |
| 预期输出 | 20 | | | | |
| 实际输出 | 20 | | | | |
| 问题描述 | 无 | | | | |
| 设计人 | 陈松 | 设计日期 | |  | |
| 测试人 |  | 测试日期 | | |  |
| 再测试人 |  | 再测试日期 | | |  |
| 问题修改摘要 |  | | | | |
| 修改人 |  | | 修改日期 | |  |

**功能测试:**

**初始按钮**

首先弹出界面，进行初始条件设置，可对初始值（迷宫鼠初末位置，速度）设置

注:为方便用户已有默认值(可更改)

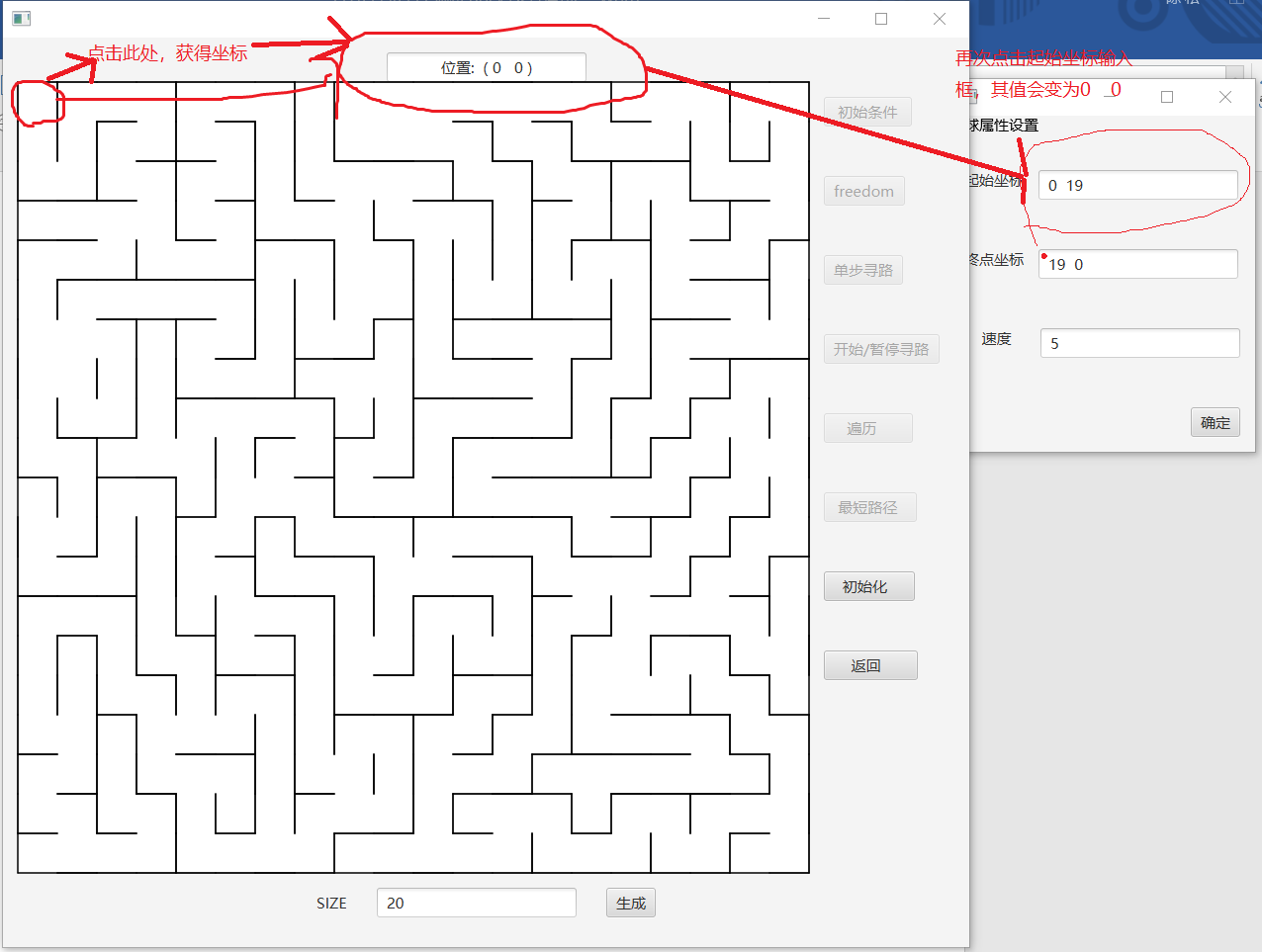


并有鼠标点击获取坐标功能

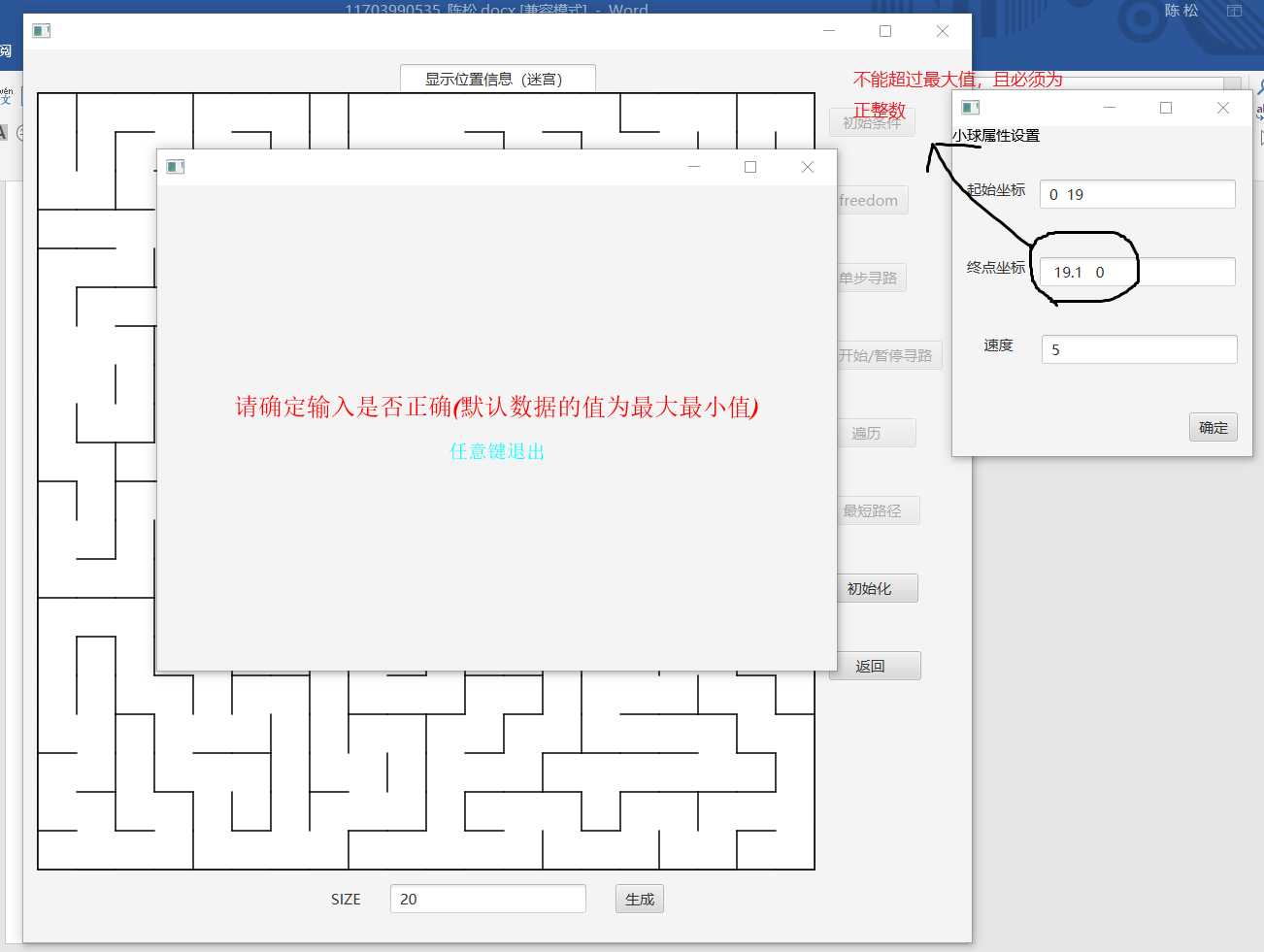
步骤一:点击迷宫中的一处

步骤二：点击初始位置输入框

完成后，初始位置会便成你选择的位置



也可选择键盘输入，但输错了，会弹出输入错误提示



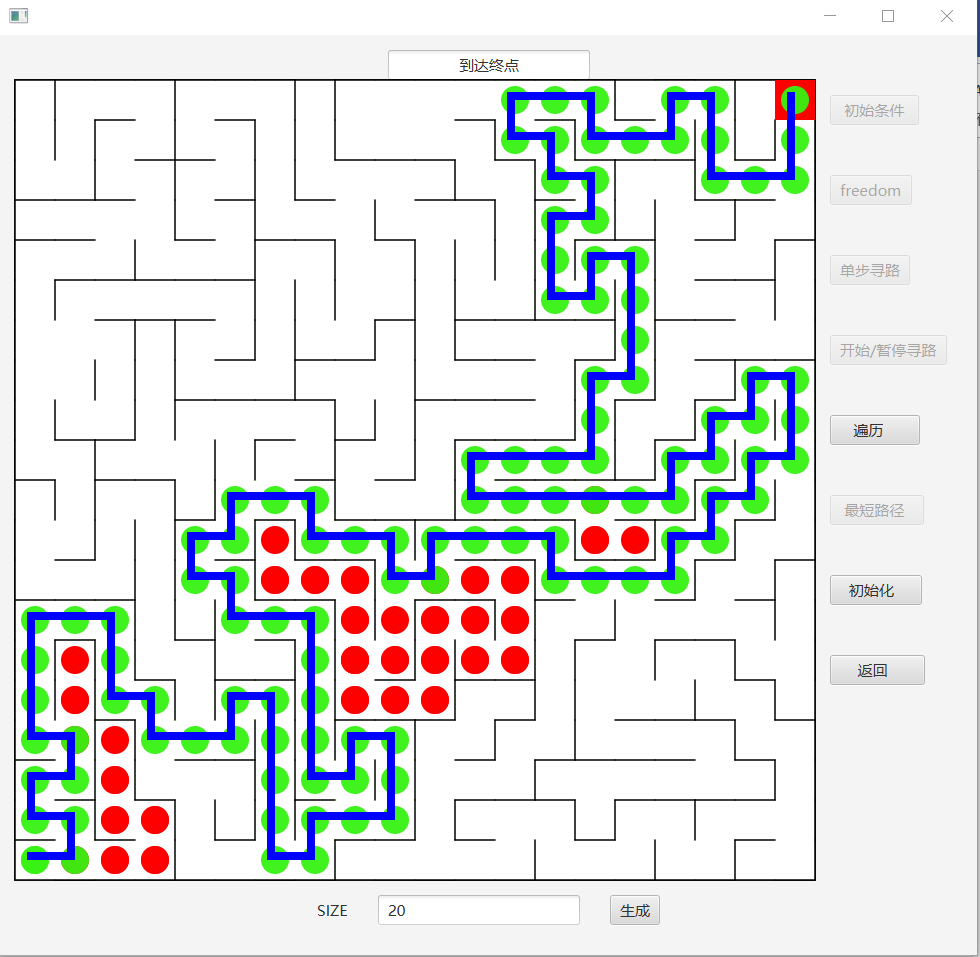
**Freedom按钮功能测试**

通过键盘上下左右控制，到达终点后会提示

如图：

**单步寻路（点击，自动）测试**

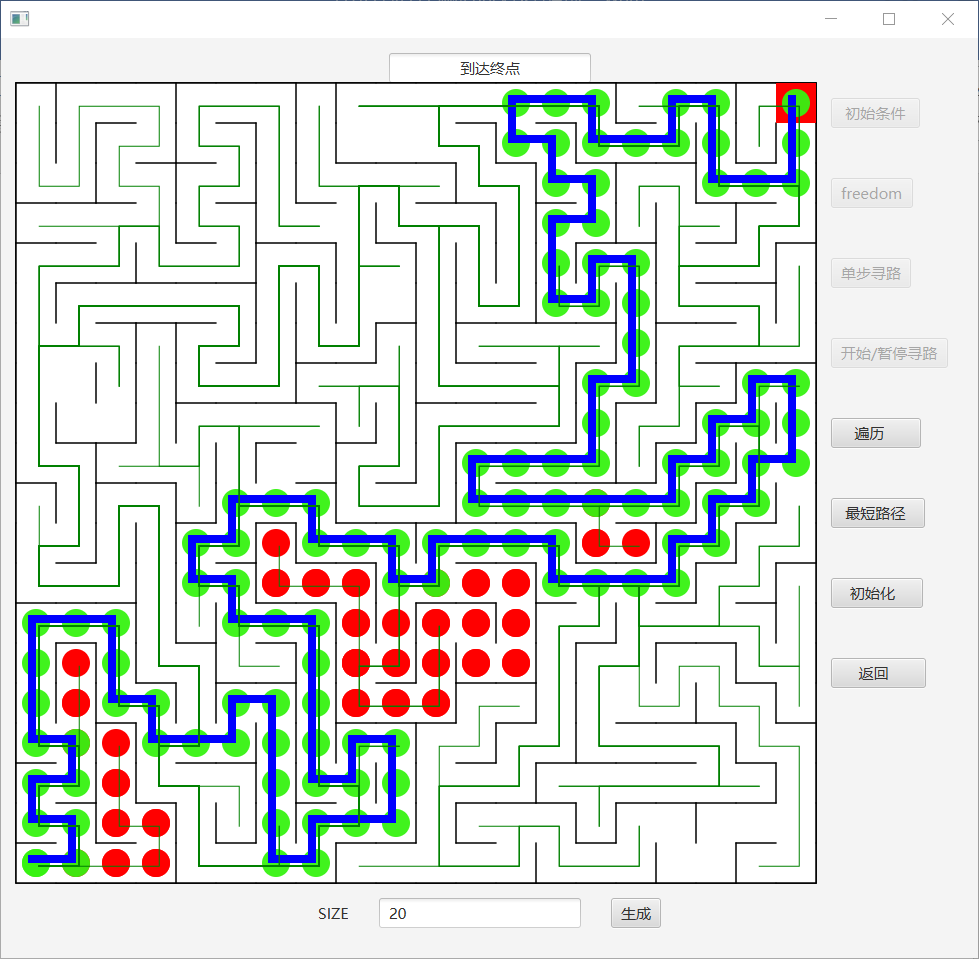
到达终点后会显示单步寻路的绝对路径



注：没在路径上的（红色）小球为返回路径

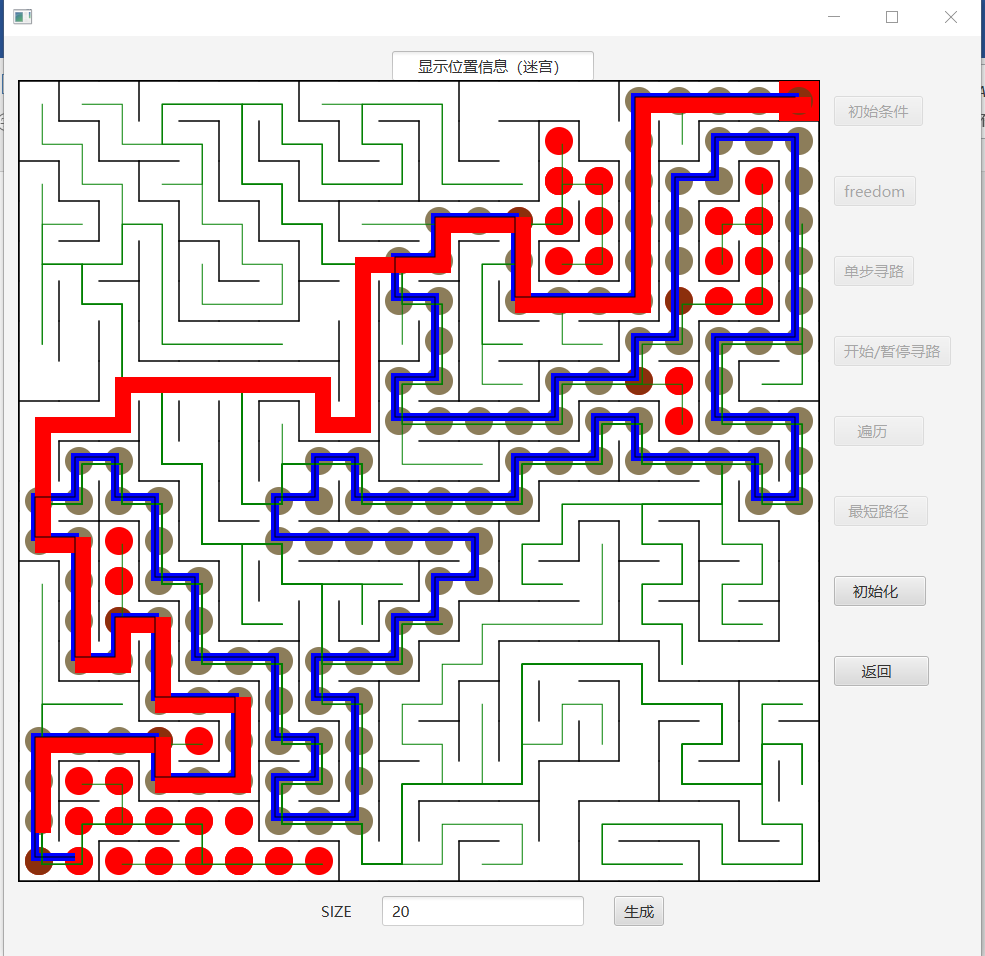
**遍历最短路径测试**

点击遍历按钮，遍历所有节点



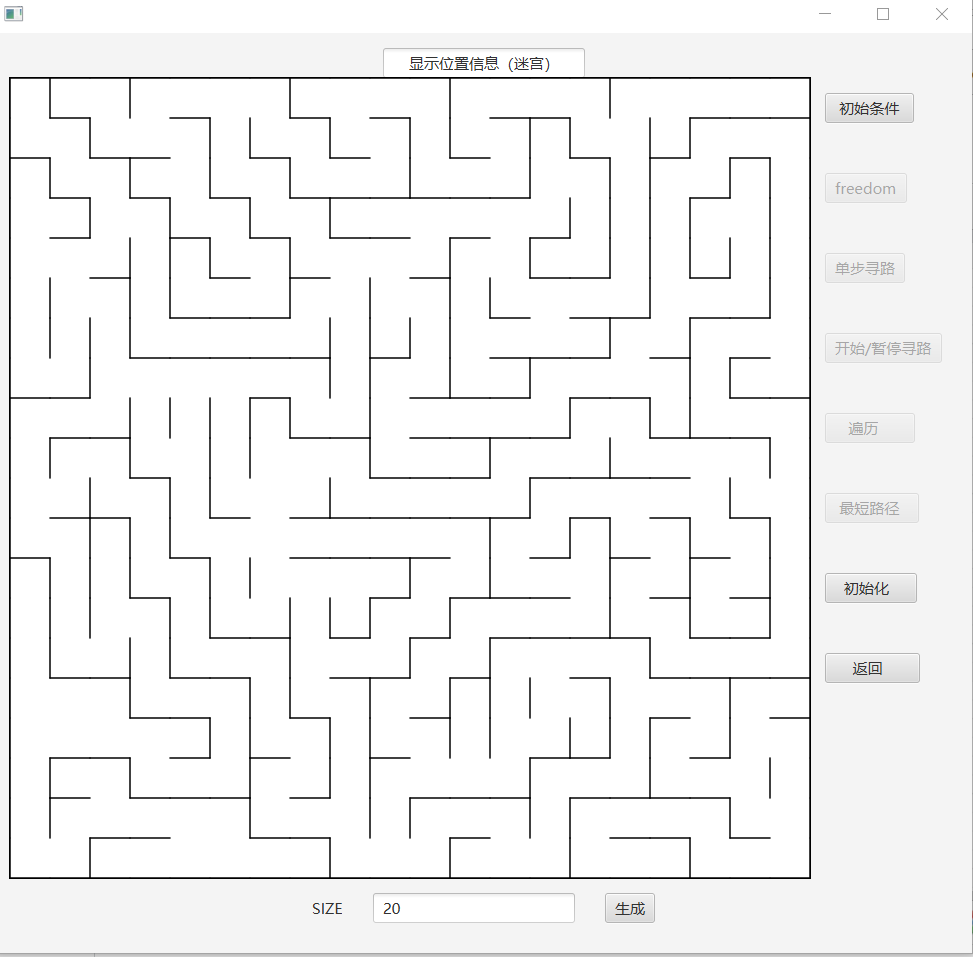
再点击最短路径按钮

便会显示最短路径和小球移动路径



**初始化按钮测试**

点击初始化按钮



恢复了原来模样

1.5 课程设计总结

这次课程设计让我收获很大，不仅仅是对深度，广度算法有了很好的理解，更多的是在程序设计中不断发生错误，不断解决错误带给我的愉悦感。同时也深刻地认识到了我编程能力的不足，还是需要继续努力。