项目说明文档

数据结构课程设计

——勇闯迷宫

作 者 姓 名： 杨晶

学 号： 1854025

指 导 教 师： 张颖

学院、 专业： 软件学院 软件工程

同济大学

Tongji University

目 录

[1 分析](#_Toc495668153)

[1.1 背景分析](#_Toc495668154)

[1.2 功能分析](#_Toc495668155)

[2 设计](#_Toc495668156)

[2.1 数据结构设计](#_Toc495668157)

[2.2 类结构设计](#_Toc495668158)

[2.3 成员与操作设计](#_Toc495668159)

[2.4 系统设计](#_Toc495668160)

[3 实现](#_Toc495668161)

[3.1 Stack类的实现](#_Toc495668162)

[3.1.1 基础操作的实现](#_Toc495668163)

[3.1.2 栈顶元素分析PointAnalysis()](#_Toc495668164)实现

[3.1.3 输出函数Output()实现](#_Toc495668165)

[3.2 Judge()函数的实现](#_Toc495668166)

[3.3 Search()函数的实现](#_Toc495668170)

[3.4 TestVisit()函数的实现](#_Toc495668174)

[3.5 主函数的实现](#_Toc495668178)

[4 测试](#_Toc495668186)

[4.1 功能测试](#_Toc495668187)

[4.1.1 测试一](#_Toc495668188)

[4.1.2 测试二](#_Toc495668189)

[4.2 边界测试](#_Toc495668193)

[4.2.1 迷宫最小的情况](#_Toc495668194)

[4.2.2 迷宫没有通路的情况](#_Toc495668194)

[4.3 出错测试](#_Toc495668197)

[4.3.1 迷宫大小超过给定最大值](#_Toc495668198)

# 1 分析

## 1.1 背景分析

人们比较喜欢走迷宫的游戏，现在中小学课程中也有设置通过孩子们最喜欢的走迷宫游戏，培养、提高孩子运笔、观察、思考等各方面的能力以及集中注意能力。在愉快的游戏过程中，孩子的智力将得到快速提升。

走迷宫有助于培养运笔、观察、思考能力以及集中注意能力。走迷宫是在仔细观察局部和整体的基础上，边判断通往出口的正确路径边前进的游戏。它能培养孩子的观察能力和思考能力，而且朝着终点努力前进也有助于提高孩子的注意力。此外，走迷宫更是一种描画方向性线条的练列，能培养孩子的运笔能力，为写字、绘画打下良好的基础。

各种各样的走迷宫游戏有助于提高孩子的观察能力、思考能力和注意力在没有岔路的基本型迷宫的基础上，通过增加或找出通往出口的入口，粘贴贴纸连接或改变道路等各种不同的走迷宫训练，提高孩子的观察能力、思考能力和注意力。

国外专家研究表明，迷宫对幼儿的逻辑能力和方向感方面的开发，有相当大的促进效果。迷宫在国外有完整的应用体系，包括在心理（如，治疗抑郁症等）、建筑（如，核电站的建造）、工业（如，迷宫式的密封技术、音响共鸣箱等）、娱乐（如，迷宫游戏、各类迷宫实景等）、教育等方面，尤以教育方面应用的最为广泛：

 综上所述，“少年智则国智, 少年强则国强”，为了深入推进人才发展体制机制改革，切实解决发展人才培养问题，促进我国青少年智力发展，增强国家人才后备力量。我们可以试着写出一个勇闯迷宫的小程序。

## 1.2 功能分析

迷宫只有两个门，一个门叫入口，另一个门叫出口。一个骑士骑马从入口进入迷宫，迷宫设置很多障碍，骑士需要在迷宫中寻找通路以到达出口。

# 2 设计

## 2.1 数据结构设计

在求解迷宫的过程中，我们往往需要将问题的候选解逐一检验。回溯法又叫试探法。在回溯法中，放弃当前解，寻找下一个候选解的过程叫做回溯。扩大当前候选解的规模并且继续试探的过程叫向前试探。

考虑到用回溯法求解问题时，往往使用栈来帮助向前试探，因此对于本题我们采用栈的数据结构。

## 2.2 类结构设计

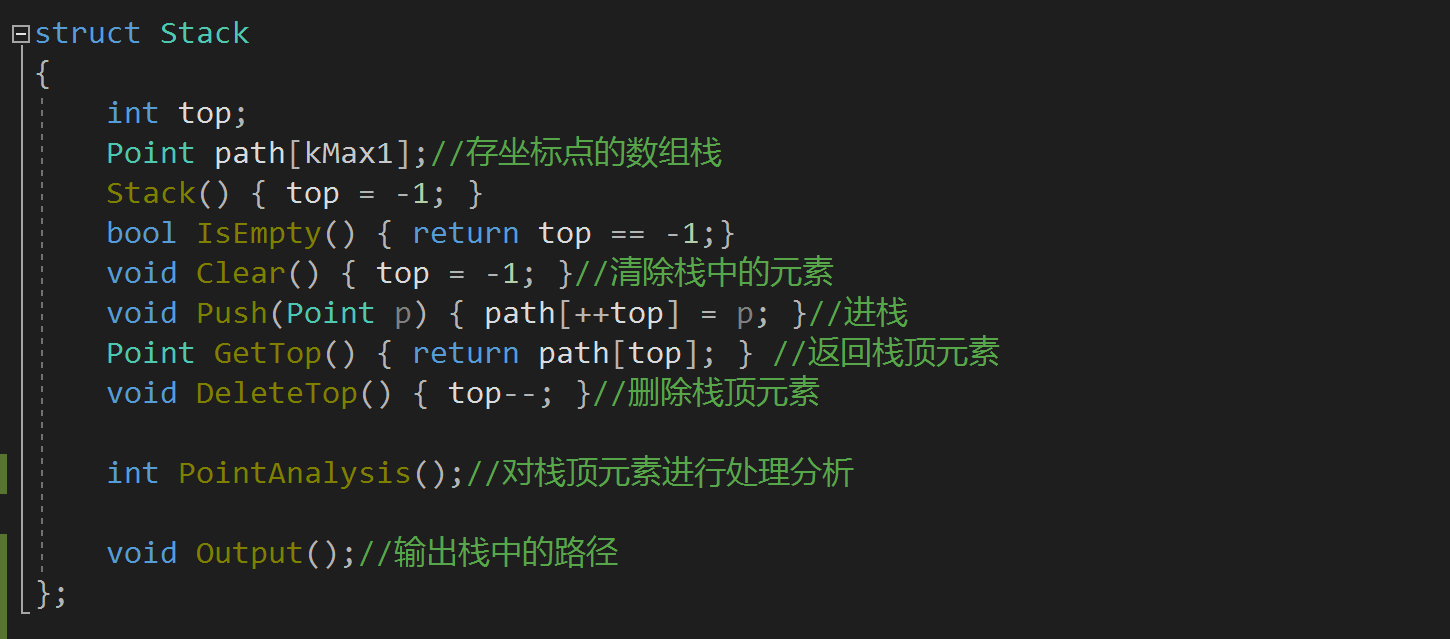
将每一次处理的坐标点放入栈中，依次试探四个前进方向直到可以前进。遇到墙则将栈中元素弹出，换个方向再次试探。同时为了防止重走原路，设置一个存储访问记录的二维bool数组visit来记录，false表示该点被访问过，已经不能访问。

## 2.3 成员与操作设计

Point结构体实现：



Stack类实现：



## 2.4 系统设计

Stack类中最重要的函数就是PointAnalysis函数，这个函数用于对坐标点的数据栈的栈顶元素进行分析，通过不同的返回值反映当前坐标栈的寻找状态——迷宫越界/找到通路/仍在迷宫内。

在类及其成员操作函数之外，我们还设定了Judge函数用来从当前的点找路，利用PointAnalysis函数对当前情况进行判断，并且决定这之后的进出栈；

Search函数则采用回溯法，利用Judge函数输出最终的通路结果，并且在Judge函数判断该点之后，无通路后在栈中弹出之前的点往下找。

这样一来主函数则只需要负责生成迷宫，调用Search函数即可输出迷宫以及通路结果，从而达到完美生成迷宫小游戏的目的。

# 3 实现

## 3.1 Stack类的实现

### 3.1.1 基础操作实现

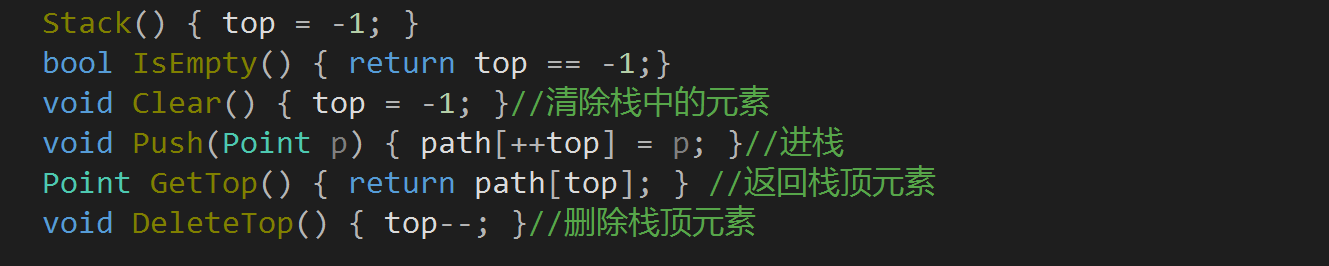
构造函数中默认top=-1；

IsEmpty判断是否为空；

Push入栈函数另path的首位置的值致为新坐标

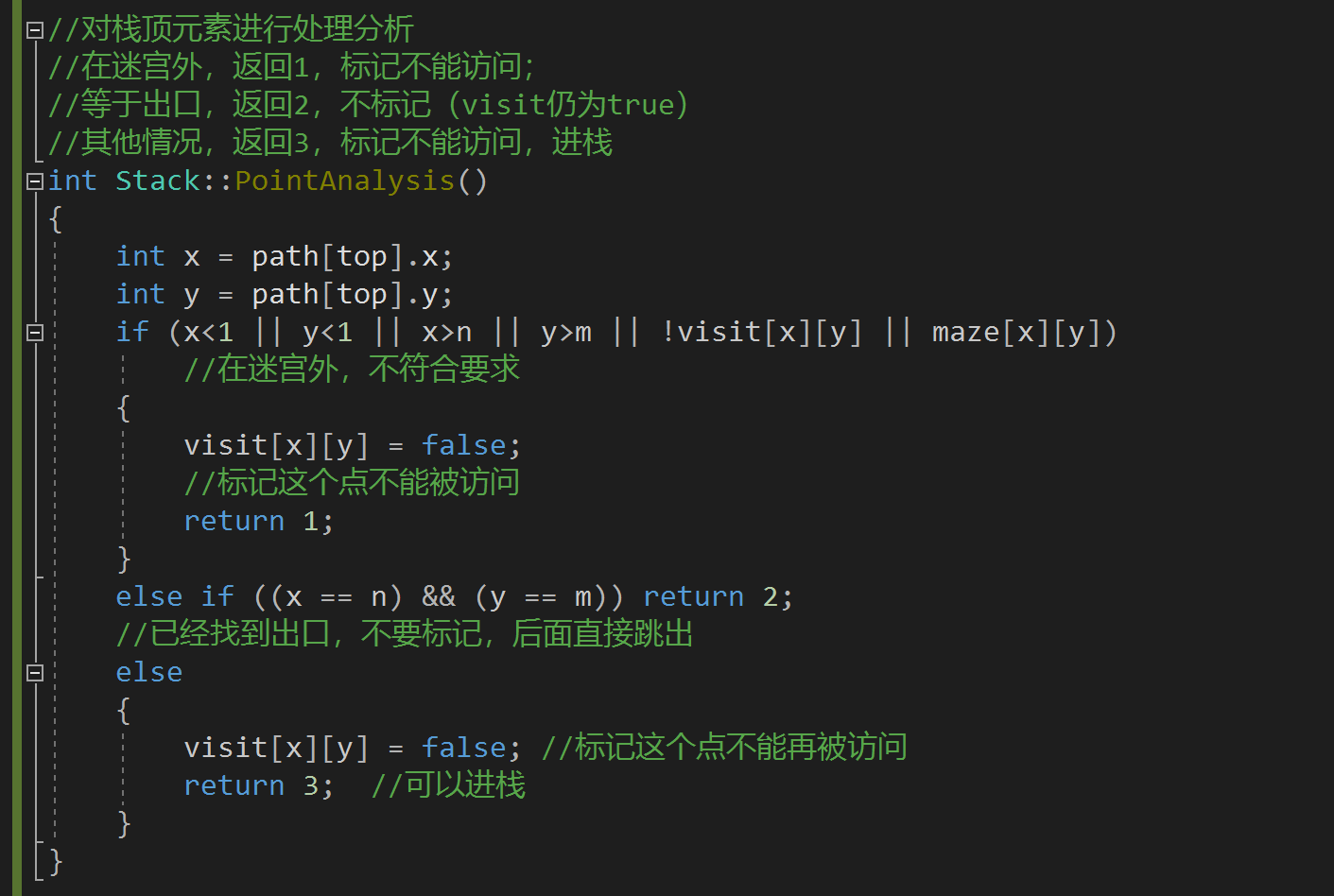
GetTop函数用于获取栈顶元素

DeleteTop元素用于删除栈顶元素



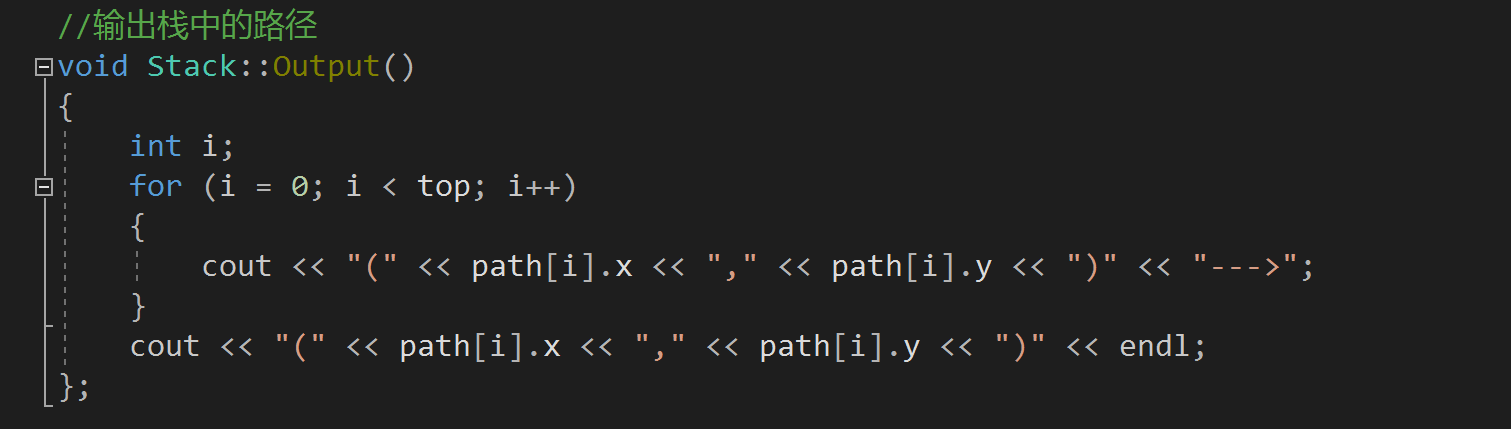
### 3.1.2 点分析PointAnalysis ( )函数实现

PointAnalysis ( )函数对栈顶单个元素进行处理分析，通过不同的返回值来对这个元素做不同的判断。如果在迷宫外，返回1，同时在Visit二维数组中标记不能访问；如果当前坐标点等于出口，返回2，不标记（visit仍为true）；其他情况，返回3，标记不能访问，进栈。



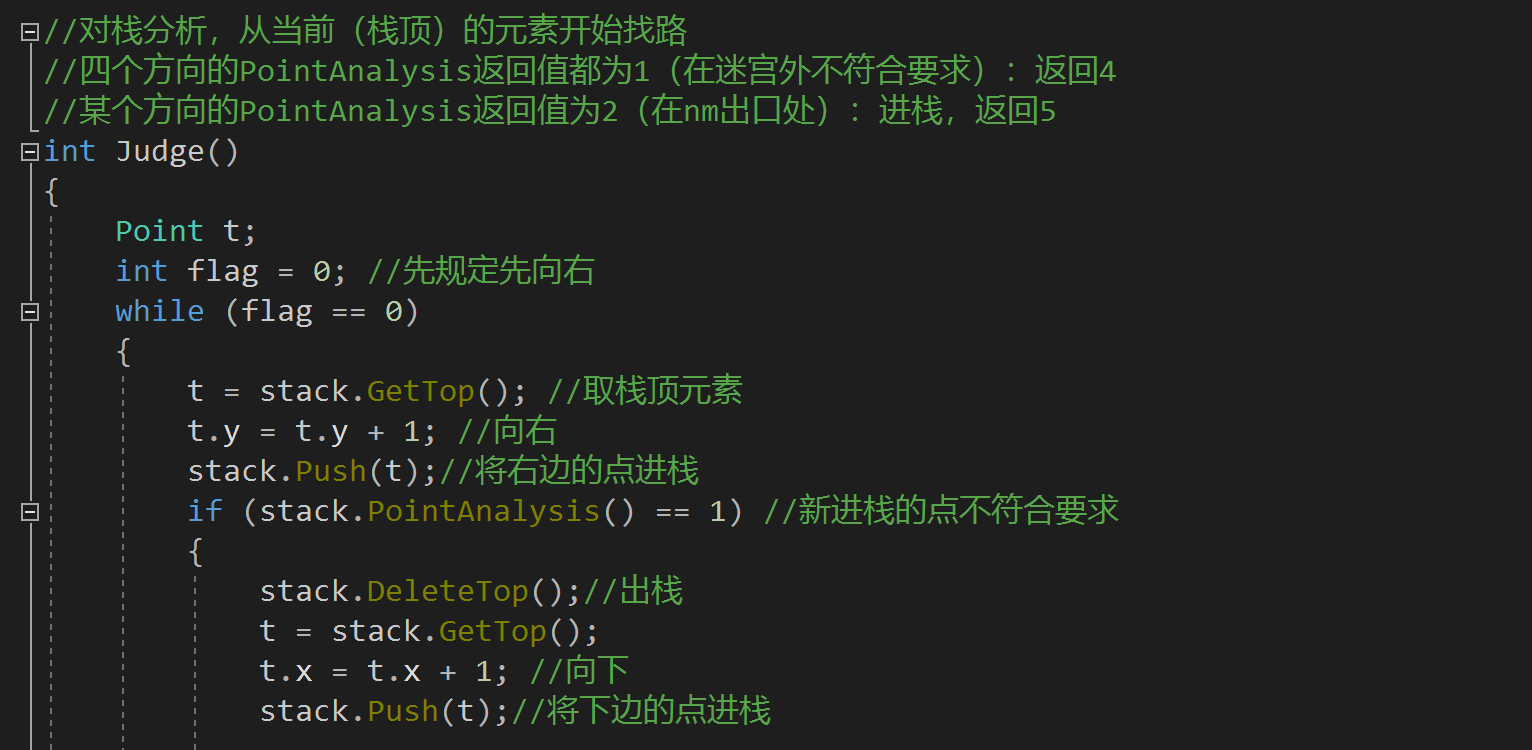
### 3.1.3 Output函数实现

Output()函数负责把栈中的路径输出



## 3.2 Judge( )函数实现

Judge（）函数负责对当前的栈进行判断，判断从当前栈的栈顶向四个方向走下去是否有出现通路的可能。





## 3.3 Search( )函数实现

Search()函数是回溯法的主要实现函数，在Judge()函数分析当前栈顶元素没有可能形成通路后，Search函数内部将这个坐标弹出，退回一个点继续往下寻找。

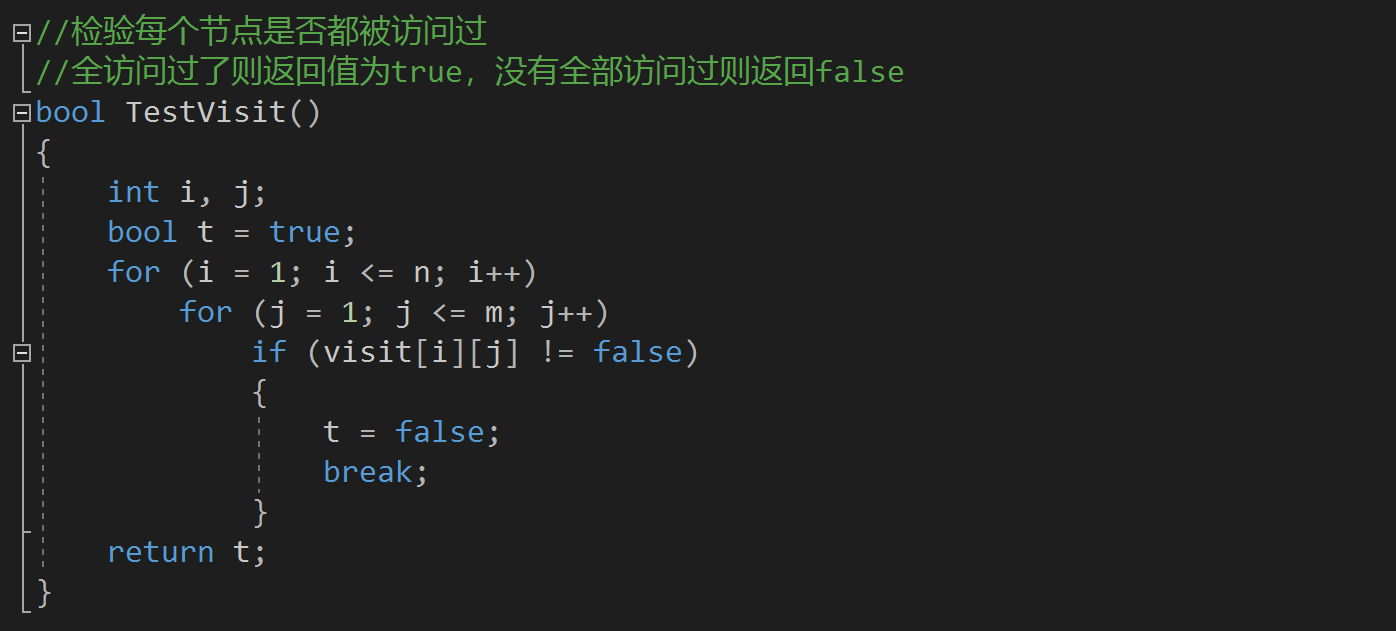
最终当某个点的Judge函数返回5时，表明找到了通路，此时就从栈中依次弹出坐标输出结果，若迷宫所有的点都被访问了一遍或者栈中元素个数为0，则代表该迷宫没有通路。





## 3.4 TestVisit实现

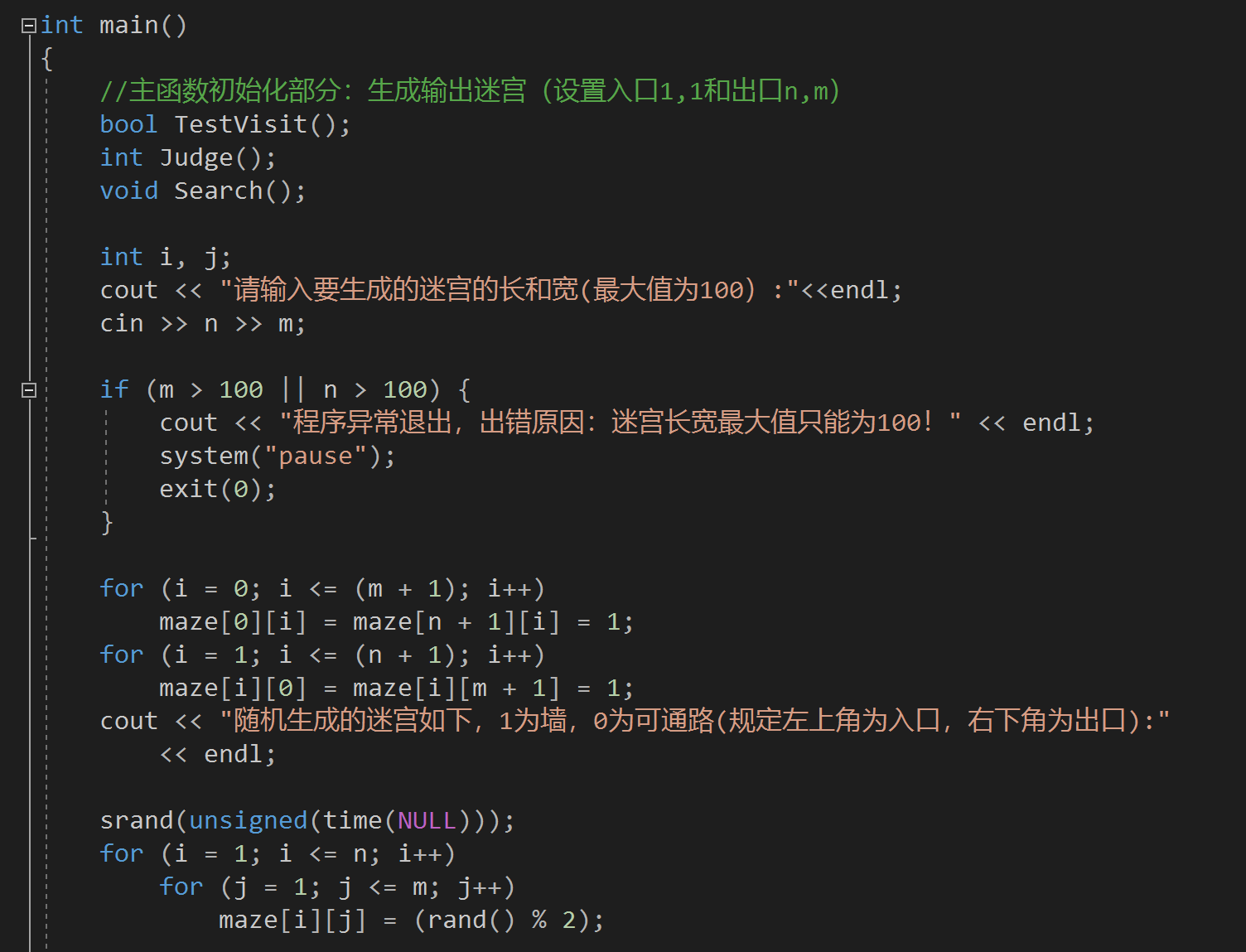
TestVisit()函数主要用于实现判断当前迷宫所有点是否都被访问过，如果都被访问过则标记为true，如果有点仍没有被访问，则标记为false。

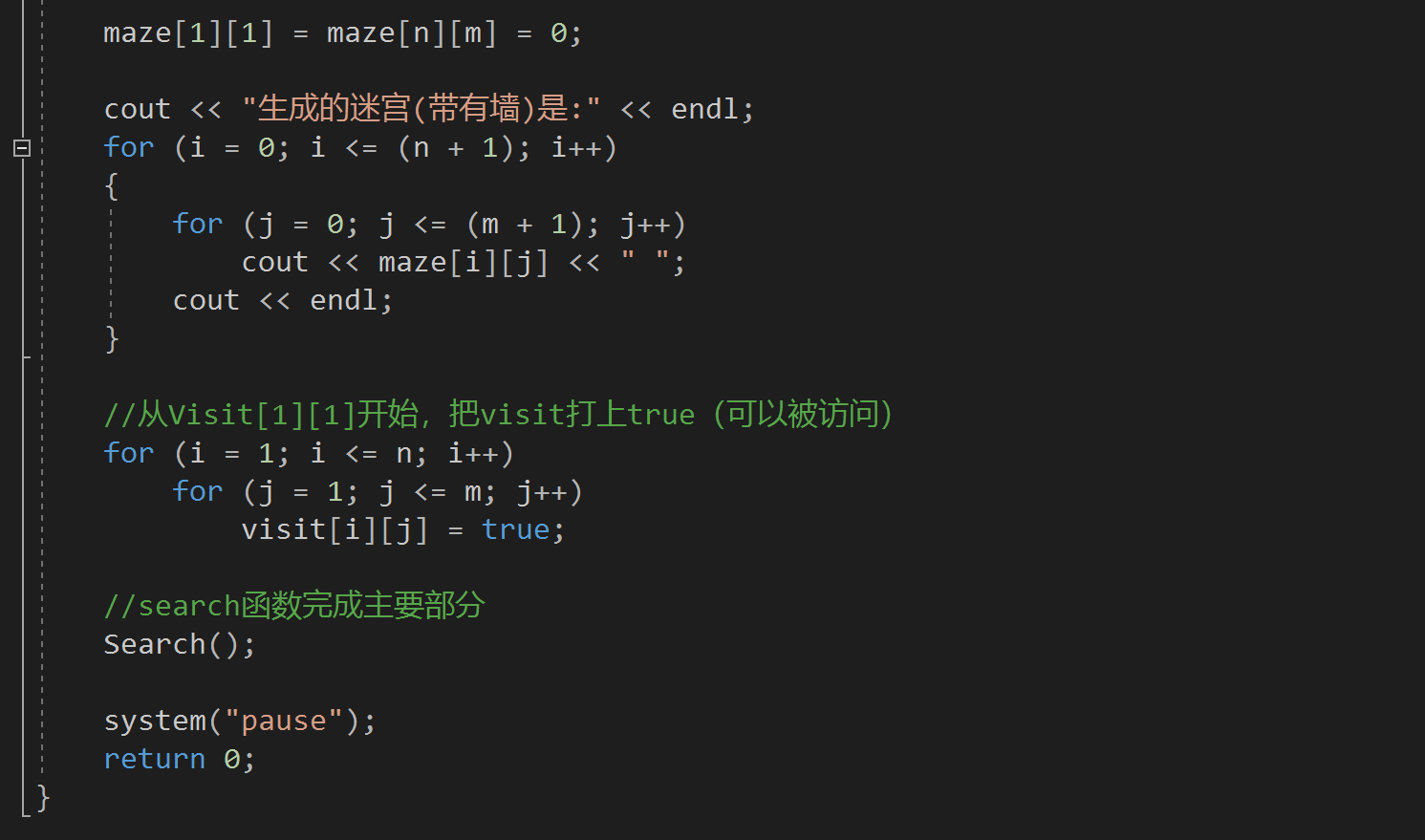


## 3.5 主函数的实现

主函数主要负责生成迷宫，这里利用了随机数rand()%2。并且对非法输入进行判断，报错处理。

迷宫生成完成后，调用Search函数即可输出通路。

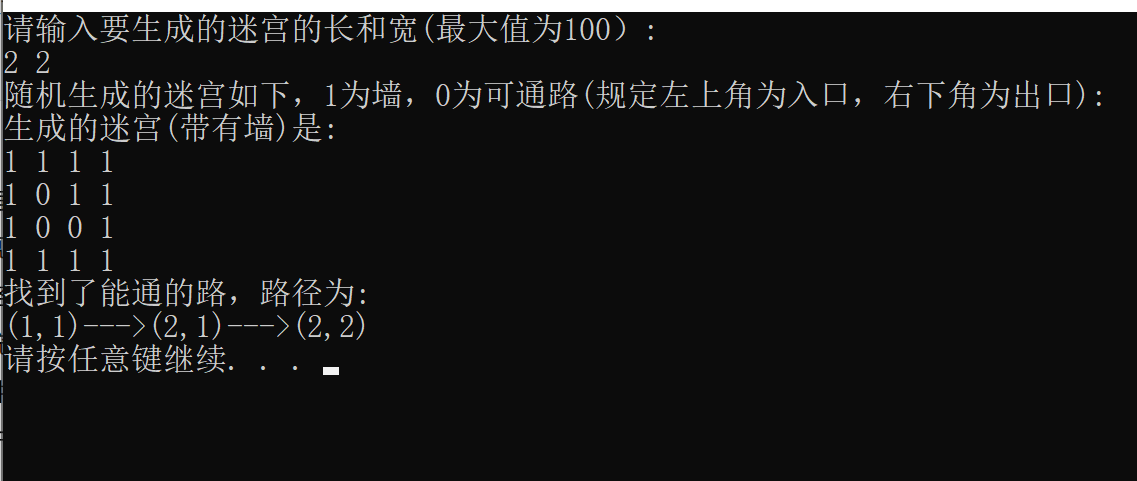




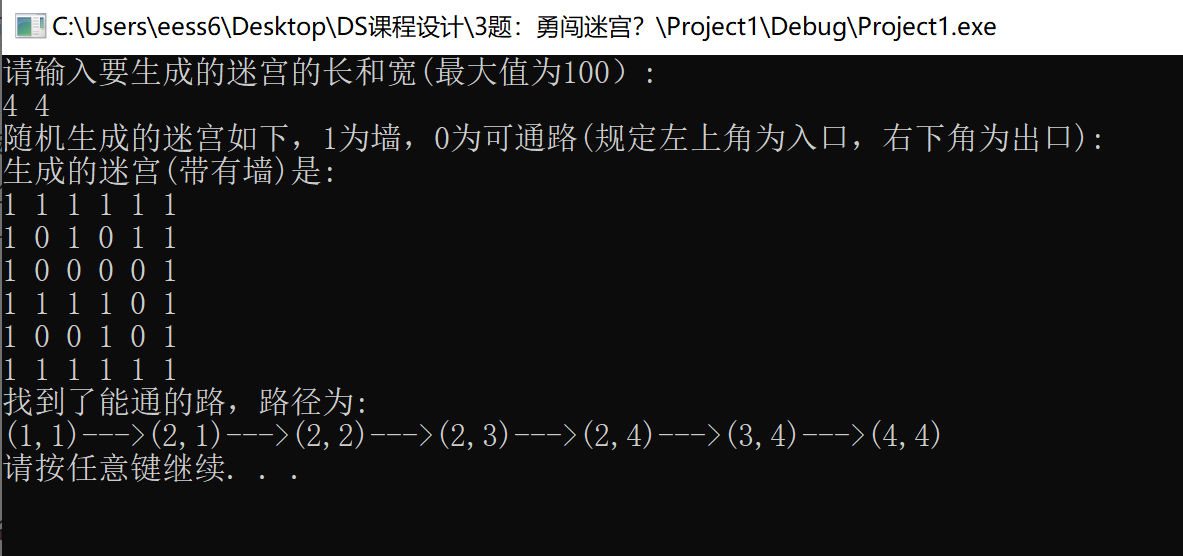
# 4 测试

## 4.1 功能测试

### 4.1.1 测试一

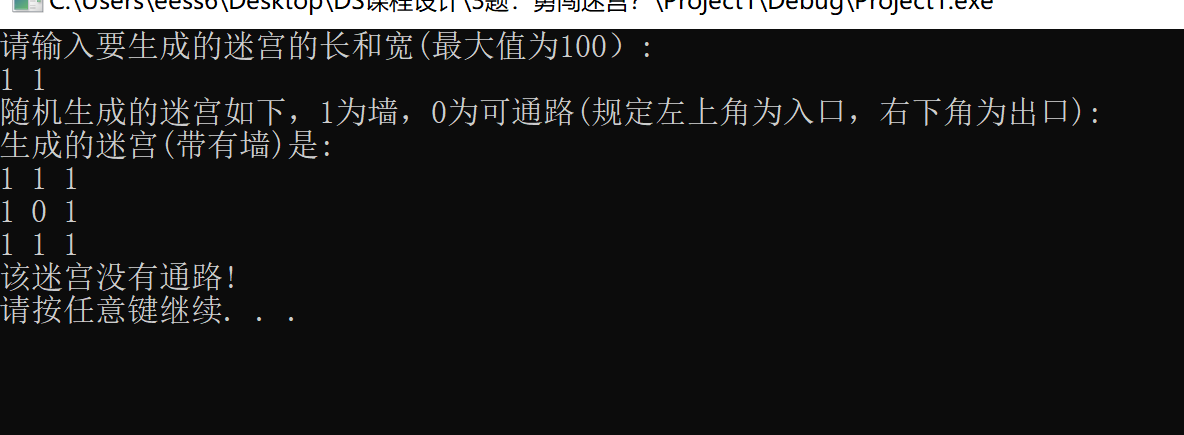


### 4.1.2 测试二

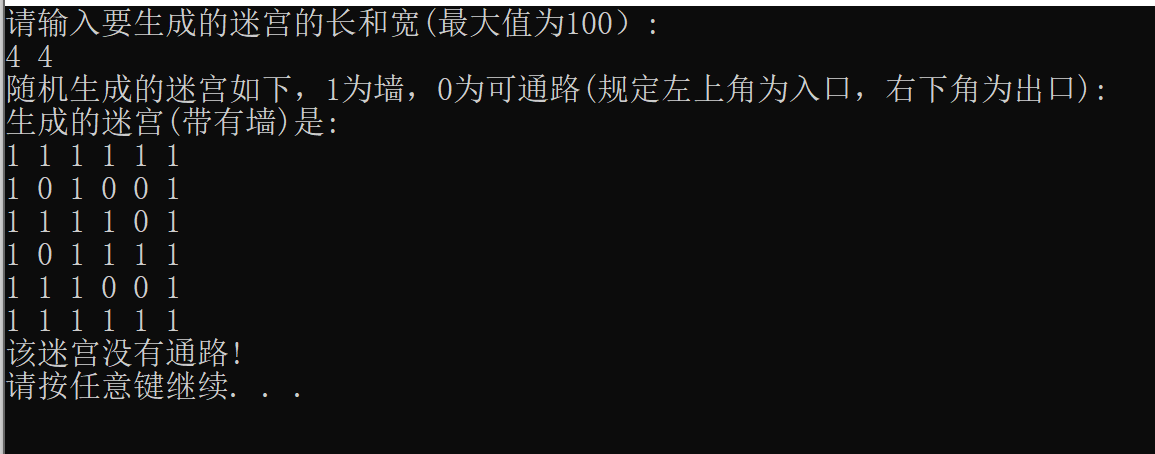


## 4.2 边界测试

### 4.2.1 迷宫最小的情况

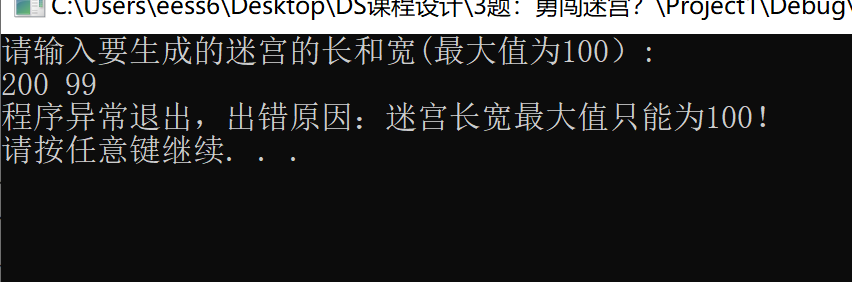


### 4.2.2 迷宫没有通路的情况

****

## 4.3 出错测试

### 4.3.1 迷宫大小超过给定最大值

****