|  |  |  |
| --- | --- | --- |
| 实验2 栈和队列 | | |
| **实验室**：  **因疫情在家中完成** | **实验日期:**  **2020.5.30** | **周次、节次：**  **填写本次实验上课的周次等**  **第（ 13 ）周/星期( 3 )第( 3.4 )节** |
| **班级**：**18计科四班** | **学号**：**20182714** | **姓名**：**孙宝才** |
| **评分标准：**  **难度系数**  **工作量**  **完成程度**  **报告质量**  **综合评定**  **老师评语：** | | |
| **一、开发环境及工具：**   1. **实验软件系统配置：（注意标注版本号）**   操作系统：**window 10**  开发工具：**visio stdio 2019**   1. **硬件环境：**   硬件基本配置主要参数： | | |
| 1. **实验目的：** 2. **通过栈掌握程序的回溯** 3. **熟悉栈的调用，掌握出栈入栈算法的使用** 4. **熟悉较长算法使用流程图使之更清晰易懂** 5. **掌握迷宫类可视化问题界面的建立，初始化** 6. **掌握枚举，结构函数的使用** | | |
| **三、你的选题：**  **选题说明：**  栈和队列实验选题如下：  **(书本p105)**  **【问题描述】以一个力MxN的长方阵表示迷宮,0和1分别表示迷宫中的通路和障碍。设计一个程序，对任意设定的迷宫，求出一条从入口到出口的通路，或得出没有通路的结论。**  **【实现提示】计算机解迷宫通常用的是“穷举求解”方法，即从入口出发，顺着某一个方向进行探索，若能走通，则继续往前进;否则沿着原路退回，换一个方向继续探索,直至出口位置，得一条通路。假如所有可能的通路都探索到而未能到达出口,则所设定的迷宫没有通路。可以二维数组存储迷宫数据，通常设定入口点的下标为(1,1),出口点的下标为(n.n).为处理方便起见，可在迷宫的四周加一圈障碍。对于迷宫中任一位置，均可约定有东、南、西、北四个方向可通。** | | |
| **四、存储结构说明和定义：**    **说明：采用顺序栈存储结构，存储通道块信息，通道块包含坐标位置，序号，访问方向，具体将在录屏中阐明。** | | |
| **五、主要算法:**   1. 程序的主要算法的程序流程图（Ctrl点击 查看，如果不行可在文档同一文件夹查找）      1. 核心函数的算法（ctrl点击查看）     （注：另有许多辅助算法，简单但不可或缺将一一在录屏中讲解，不在此重复）   1. 给出每个子函数的函数头   /\*   \* 使用穷举法，找到一条可行通路即返回   \*/  Status MazePath(MazeType maze, PosType start, PosType end);  /\*   \* 初始化一个规模为N×N迷宫   \* start和end分别为迷宫的入口坐标和出口坐标   \*/  void InitMaze(MazeType maze, PosType\* start, PosType\* end);  /\*   \* 判断当前位置是否可通过：要求该位置是从未曾探索的通道块   \*/  Status Pass(MazeType maze, PosType seat);  /\*   \* 获取下一个应当探索的位置   \* di指示当前所处位置的探索方向，包括East, South, West, North   \*/  PosType NextPos(PosType seat, int di);  /\*   \* 留下初始访问足迹，初始访问足迹即向东访问   \*/  void FootPrint(MazeType maze, PosType seat);  /\*   \* 在迷宫的seat处留下mark标记，此处的函数改进为可以留下任意标记，包括探索方向的标记   \*/  void MarkPrint(MazeType maze, PosType seat, int mark);  /\*   \* 构造一个通道块信息并返回   \*/  SElemType Construct(int ord, PosType seat, int di);  /\*   \* 判断两个坐标是否相等   \*/  Status Equals(PosType a, PosType b);  /\*   \* 绘制迷宫   \* 以图形的方式呈现迷宫当前的状态  void PaintMaze(MazeType maze); | | |
| **六、程序运行结果测试：**  **录屏演示，详细见文件**  **注：录屏讲解包含算法实现，算法分析，程序运行演示等** | | |
| 1. **本次实验小结：**   **（此栏不允许空白）**   1. 分析自己的主要算法   **采用顺序栈存储结构，存储通道块信息，通道块包含坐标位置，序号.**  **在进行必要的准备工作（如迷宫地图的绘制，初始化，栈的相关算法实现）**  **主要编写的算法是迷宫穷举算法。具体的算法分析将在视屏中讲解。**   1. 说明本次实验中遇到的问题和你的解决方案   **1.迷宫穷举算法较长，光看代码无法理解，根据代码画出流程图再进行理解就变得非常简单。**  **2.迷宫算法比较复杂，涉及多个分支，更好的理解方法是使用断点一步一步进行编译**   1. 对本次实验的完成情况自我评价   **本次栈和队列实验中，选择的是用栈解决迷宫问题。一般栈是用来解决回溯类，递归类问题，为了在树实验中可以很好的完成，这里选择较难的迷宫算法。**  **代码来自网友分享，经过自己的更改和优化，历时两星期理解转化为自己的知识。** | | |
| 1. **附录： （源程序代码）**   **代码详看附录** | | |
| **九、你使用或推荐的参考资料：**  **1.书本：《数据结构(C语言版)].严蔚敏\_吴伟民》**  **2.回溯算法：**  <https://blog.csdn.net/My_ben/article/details/82768047?ops_request_misc=%257B%2522request%255Fid%2522%253A%2522159067530819724835841523%2522%252C%2522scm%2522%253A%252220140713.130102334.pc%255Fall.%2522%257D&request_id=159067530819724835841523&biz_id=0&utm_medium=distribute.pc_search_result.none-task-blog-2~all~first_rank_v2~rank_v28-1-82768047.pc_insert_v7&utm_term=%E7%A8%8B%E5%BA%8F%E5%9B%9E%E6%BA%AF> | | |