浙江理工大学课程设计答辩记录表

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **日 期** | **学 院** | **专业 / 班级** | **学 生** | **指导教师** |
| 2025/1/7 | 计算机学院 | 23计算机科学与技术4班 | 李凯涛 | 孙树森/郭奕亿 |
| **课程设计题目** | | 数据结构与算法课程设计 | | |
| **答辩时间** | | 2025/1/7 10点 | **答辩地点** | 10-414 |
| 答辩记录：  1、问： 为什么迷宫生成会有一闪一闪的动画呢？  答： 迷宫生成采用了随机化深度优先搜索算法。首先为迷宫生成过程分配所需的数据结构内存，包括用于存储待探索节点的栈stack和用于标记节点是否已被访问的二维数组visited。核心是一个循环，只要栈stack不为空，就持续进行探索。每次从栈中取出一个节点current，并更新进度显示。在探索当前节点的相邻节点之前，会对四个方向进行随机化。检查相邻节点是否在迷宫范围内且未被访问。如果满足条件，则将当前节点与相邻节点之间的墙壁打通，将相邻节点标记为已访问，并将其压入栈stack。在打通墙壁和标记节点后，会调用display\_maze函数显示当前迷宫状态，并通过msleep函数进行短暂延时，以实现迷宫生成过程的动画效果。因此就会有一闪一闪的动画生成。  2、问： 讲解下这个迷宫问题中，关于最短路径的代码为什么这里写着一份for循环语法。  答：这段for循环代码在寻找最短路径的函数find\_shortest\_path中，主要用于为广度优先搜索（BFS）算法初始化两个重要的数据结构：parent和visited，并在内存分配失败时进行相应的清理和错误处理。内层for循环从0到i - 1遍历，释放之前已经成功分配的parent[j]和visited[j]的内存。这是因为之前的分配可能成功了，但当前行的分配失败，需要清理之前的分配以避免内存泄漏。  3、问：讲解下这份代码找到最短路径的逻辑  答：使用广度优先搜索算法来寻找迷宫的最短路径。初始化 BFS 相关变量，将起点maze->start加入队列queue，并将起点标记为已访问。同时创建一个Progress结构体来跟踪搜索进度。核心是一个循环，只要队列不为空且还未找到终点，就持续进行搜索。在每次循环中，取出队列头部的节点current，并检查其四个方向（上、下、左、右）的相邻节点。对于每个相邻节点nextX和nextY，通过is\_valid\_move函数检查其是否是合法的移动位置，并且尚未被访问。如果满足条件，则将该相邻节点标记为已访问如果相邻节点是终点maze->end，则表示找到了最短路径，将found\_path标记为true并跳出循环。最后调用reverse\_path函数将路径反转，使其成为从起点到终点的顺序。  4、问：讲解下代码中路径和迷宫存储在桌面上的实现  答：迷宫和路径存储到桌面的功能分别由save\_maze和save\_path\_to\_desktop函数实现。**save\_maze函数：将迷宫保存到桌面**。首先写入迷宫的逻辑大小maze->size。接着写入起点的坐标maze->start.x和maze->start.y。然后写入终点的坐标maze->end.x和maze->end.y。最后通过两层嵌套循环，将迷宫的二维网格数据逐行逐格地写入文件。**save\_path\_to\_desktop函数：将路径保存到桌面。首先**打开文件，使用fopen函数以写入模式打开指定路径的文件。接着写入路径的长度path->length。然后通过循环，将路径上每个点的坐标path->points[i].x和path->points[i].y按格式(x, y)逐行写入文件，每行结束后写入换行符\n。关闭文件并返回结果：使用fclose函数关闭文件。打印提示信息，告知用户路径已保存到桌面的具体路径。返回SUCCESS表示保存成功。 | | | | |