

6. 选题之三：迷宫游戏（难度系数：★★★★）

6.1 问题描述

作为一名迷宫游戏爱好者，你经常在网络上寻找各种迷宫挑战。然而，你发现现有的迷宫游戏过于简单，无法满足你对挑战的需求。因此，你决定运用自己的编程技能，开发一个可以随机生成任意规模迷宫的程序，并允许用户亲自体验游戏。

此外，为了使不熟悉迷宫游戏的同学也能享受游戏乐趣，你计划在游戏中加入一个自动寻路系统，该系统能自动指示玩家下一步的行进方向。具体要求如下：

要求 1 迷宫的生成与展示

程序应从标准输入接收多个参数，**至少包括迷宫的长度和宽度**。你可以根据需要设计更多参数，如**墙壁密度和通道宽度**等。为确保游戏的可玩性，迷宫的**长度和宽度不应超过 50**。

程序应根据这些参数随机生成一个迷宫，迷宫中至少包括墙壁、通道、起点和终点，并确保每次生成的迷宫具有随机性，同时**起点和终点必须是连通的**。

迷宫生成后，应在字符界面上进行展示。例如：

```

迷宫布局：
# = 墙
S = 起点
E = 终点
空格 = 可行走的路径

迷宫示例：

#####
#S    #                #
#  ### # #####  #
#  #   #   #          # #
#  #  ###  ###  #  #  #
#    #          #  #  #
##### # #####  #  #
#   #  #          #   #
#  #  #####  #####
#  #   #          E   #
#  #  ### # #####
#  #   #  #  #          #
#### #  #  # #####
#   #  #  #  #          # #
# ##### #  # #####  #
#           #   #  #  #
#####

```

如果迷宫的参数设置合理，或字符界面实现美观清晰，可酌情加分 2 至 10 分。

要求 2 支持迷宫游玩

迷宫生成完毕后，玩家应能开始游玩。操作模式可以自由设计，例如通过方向键或输入 WASD 来移动玩家位置。游戏应实时维护并展示玩家在迷宫中的当前位置。

如果玩家试图移动至墙壁或离开迷宫边界，程序应给出提示并拒绝该操作。若玩家到达终点，程序应询问玩家是否希望重新开始，并重新生成迷宫地图及重置玩家位置。

要求 3 支持自动寻路

在游戏过程中，若玩家输入特定按键，程序应自动寻找并计算从当前位置到终点的最短路径，并在界面上指示下一步的方向。

如果存在多条最短路径，程序可以选择任意一条显示。

6.2 评分标准

本题的得分由 5 部分组成：

1. 基本功能实现（70 分）：

- a) 迷宫的生成与展示。(24 分)
- b) 支持迷宫游玩。(23 分)
- c) 支持自动寻路。(23 分)

对于每个基本功能，如果实现不完全或未满足上述具体要求（如起点和终点不连通），每缺失一项扣除 2 分，直至扣完。

2. 实验报告（15 分）：

在文档中，需要包含以下内容：

- a) 功能的代码位置(文件名或起始行号、函数名等)
- b) 描述功能的实现思路，如果涉及到一些复杂算法，需要对你使用的算法进行描述
- c) 其他任何你想让助教知道的内容

注意：判断你是否选择某个功能的唯一依据是你文档中的声明。如果你实现了某个功能，但没有在文档中声明，助教将不会测试这一功能，你将不会得到这部分分数。如果你声明实现了某项功能，但没有给出对应的代码位置和实现思路，或没有描述使用的算法，缺少一项将会被扣除 2 分，封顶 10 分，扣完为止。

3. 代码风格（5 分）：

代码应该拥有较好的代码风格、包括但不限于：合理的缩进、规范的函数和变量、合理的封装、适当的注释等，如果代码过于杂乱，会被扣除代码风格分，每项 2 分，扣完为止。

4. 程序鲁棒性（7 分）：

你的程序需要处理异常的输入，并对异常的输入输出提示信息，尽量不要使程序崩溃。助教会使用一些常见的异常输入测试你的程序，如果程序崩溃或没有给出相应提示信息，则每个未处理的异常输入扣 2 分，扣完为止。

5. 提交正确性（3 分）：

提交的内容要完整。提交的二进制可以直接在 Windows 系统上运行，或者其对应源文件可以在 Visual Studio，32 位 Debug 模式下正确编译。

6.3 参考指南

每次移动后,如果需要刷新屏幕,在 Windows 平台下可以使用 `system("cls")` 命令(需包含 `stdlib.h` 头文件)。此方法可能导致屏幕闪烁。若使用其他方法避免此问题,可酌情加分 2 至 10 分。