

人工智能原理

课程项目 1

注意：

- 1) 请在网络学堂提交**电子版**；
- 2) 请在 **4月10日晚 23:59:59** 前提交作业，**不接受补交**；
- 3) 如有疑问，请联系助教：

李 震：lizhen22@mails.tsinghua.edu.cn

李可伊：lky23@mails.tsinghua.edu.cn

王子安：wangza24@mails.tsinghua.edu.cn

请在下面题目中选择一题，通过搜索算法求解。自己实现搜索算法以及必要的界面。

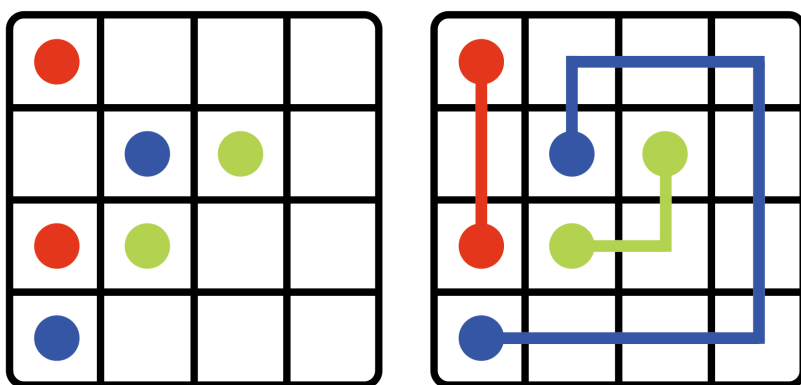
题目一：自选

欢迎大家根据自己的兴趣选择一个问题，使用搜索算法进行解答，求解方法不限于前面几次课所讲授的搜索算法，但需要征得助教的同意之后再作。

请大家不要做下面这些问题：梵塔问题、食人生番问题、八皇后问题、独立钻石问题、夹子棋（黑白棋）、一笔画问题、五子棋、计算24、2048、数独。课上给大家演示过的游戏也尽可能不要做。

题目二：交叉线

在“交叉线”游戏中，棋盘上放置了若干对不同颜色的棋子，玩家需要相同颜色的棋子连接起来，并确保不同颜色棋子的连接路径不会交叉且位于棋盘内，如下图所示。基于这个基本规则，请你设计合适的搜索算法，完成以下几个问题：

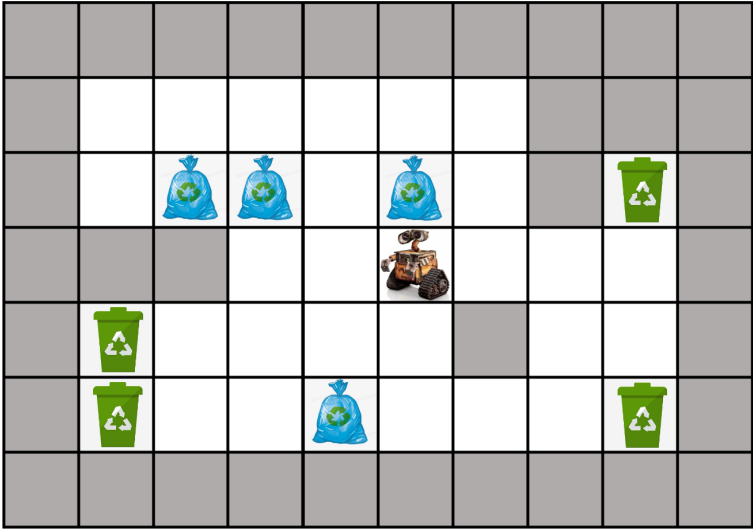


1. 请你实现问题场景的生成，包括地图大小，棋子对的数量与位置。要求（1）允许用户自定义棋盘大小，如 `size=8` 表示 8×8 棋盘；（2）允许用户自定义棋子对的数量与分布，如上图中有3对棋子， $(1,1) - (2,3)$ 代表蓝色棋子的分布。
2. 请你设计搜索算法完成棋子之间的连接，并使得连接路径最优（最优指棋子之间的连接路径长度最短，如上述红色棋子之间的路径长度为3）。如果无法完成所有棋子的连接，请返回“失败”的提示。
3. 在上述基本规则的基础上，现在需要进一步考虑转向的代价，在原本路径代价的基础上，每转向一次，代价+2。如上图绿色棋子之间原本路径代价为3，加上转向代价后综合代价

为5。请你设计搜索算法，完成棋子之间的最佳连接方式，使得综合代价最小。如果无法完成所有棋子的连接，请返回“失败”的提示。

题目三：智能清洁机器人

智能机器人在近年来发展迅猛，逐渐应用在我们的生活当中。在本题目中，你需要利用搜索算法帮助一个智能清洁机器人在复杂环境中执行垃圾清理任务。如图所示，场地中存在 k 个垃圾和 m 个回收站。机器人的目标是将垃圾全部送入回收站中，由于机器的特殊构造，只能推动垃圾而不能拉动。机器人可以向上下左右四个方向移动，移动的过程中不能穿越障碍物或其他垃圾堆。请你设计搜索算法，帮助机器人找到最优的垃圾清理路线，所谓最优，就是在完成任务的前提下机器人的移动距离是最短。



题目要求：

1. 请你实现场景的生成，包括地图形状，机器人的初始位置，垃圾和回收站的数量与分布。
2. 若场地上存在着等量的垃圾和回收站，为了响应垃圾分类的号召，垃圾与回收站之间存在一一对应关系，每个垃圾都需要送入对应的回收站当中；垃圾不能堆叠并且机器人同一时间只能推动一个垃圾。基于以上认知，请你设计一种搜索算法，帮助机器人找到最优路径，使得垃圾清理的总移动代价最小。
2. 若场地中存在着任意数量（可人为指定）的垃圾和回收站，机器人在推垃圾的过程中，两个垃圾可以堆叠成为一个垃圾。这个堆叠起来的垃圾仍然只占一个格子，且可以推送到任意一个回收站，在这个例子中，我们假设回收站的容量充足，每个回收站都能够接收多个垃圾。请设计一种搜索算法，帮助机器人寻找最优路径，使得垃圾清理的总移动代价最小。

本项目要求提交以下材料：

1. **EXE可执行文件和源代码。**
 - a) **EXE可执行文件。**助教没法为每位同学的代码都去搭建可行的运行环境，所以请大家务必导出EXE可执行文件（可以使用pyinstaller等工具进行打包），以便助教对你的程序进行验证。注意：如果不提供可执行文件将对分数产生较大影响。
 - b) 具备可读性的源代码。
2. **实验报告。**
 - a) 搜索算法的设计思路，例如启发函数的设计等等；
 - b) 搜索算法的结果演示，可以结合可视化结果进行解释；
 - c) 必要的讨论，例如调试过程中碰到的问题，对该问题的展望等。

注意事项:

- 1) 实验报告是对算法设计思路、探索调试过程和算法实现结果的描述总结，所以请**不要**在实验报告中粘贴大段代码，请更多地展示你的思考过程和算法的实现效果；
- 2) 同学们需要实现可视化界面来展示搜索算法的结果，但是力求简洁，便于助教测试即可，希望同学们将精力更多投入到算法设计，而不是把界面弄的很好看；
- 3) 编程语言原则上要求使用 Python，如果你使用其他编程语言，应详细描述运行的环境；

若同学们还有其他疑问，请通过邮箱或者课程群联系助教。