

人工智能导论课程实验报告

|  |  |
| --- | --- |
| 实验名称： | 实验一  K启发式搜索 |
| 实验日期： | 2024-3-1 |
| 实验地点： | 西部片区4号楼301 |
|  | |
| 学号： | 33920212204567 |
| 姓名： | 任宇 |
| 专业年级： | 软工2021级 |
| 学年学期： | 2023-2024学年第二学期 |

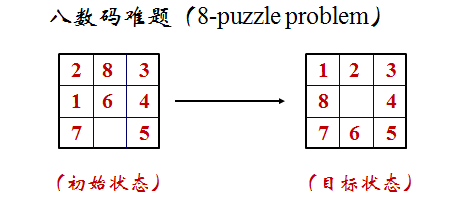
1. **实验目的**

启发式搜索就是利用知识来引导搜索，达到减少搜索范围，降低问题复杂度的目的。本实验通过解决八数码问题，帮助学生更好的熟悉和掌握启发式搜索的定义、估价函数和算法过程。

1. **实验内容**

利用启发式搜索（A\*算法）解决8数码问题

8数码规则：在 3× 3 的方格棋盘上，分别放置了表有数字 1、2、3、4、5、6、7、8 的八张牌，初始状态 S0，目标状态 Sg，如下图所示。

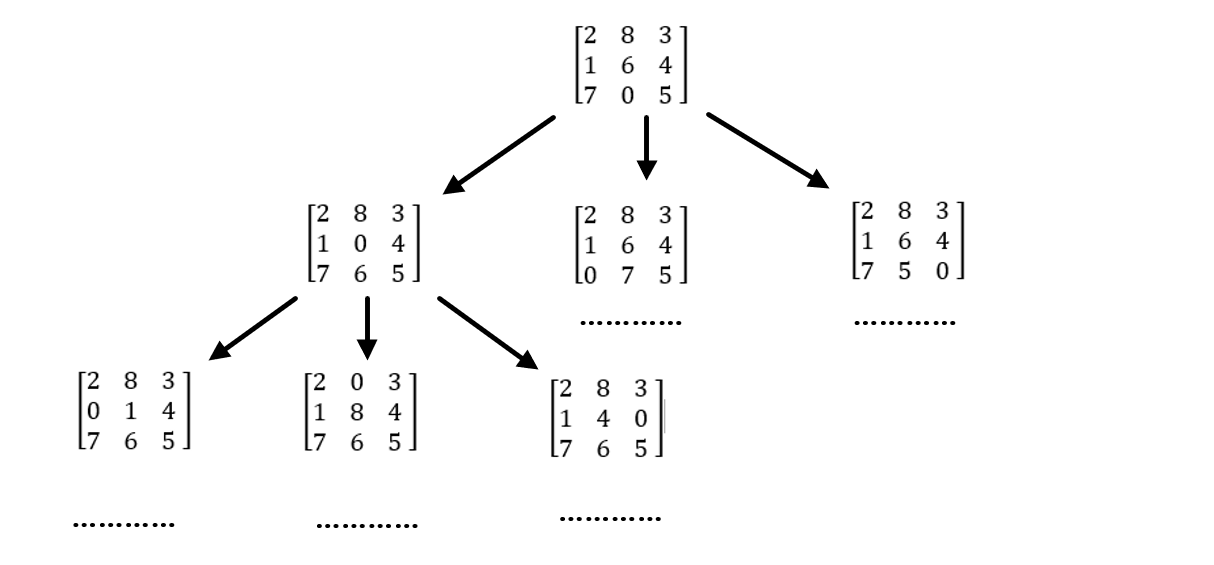


可以使用的操作有空格左移，空格上移，空格右移，空格下移。

即只允许把位于空格左、上、右、下方的牌移入空格。请编程实现该问题的解决。

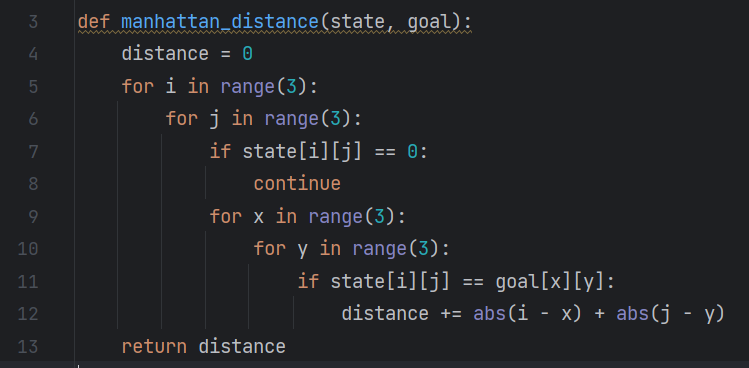
1. **实验过程**
2. 八数码解过程的节点状态图：

八数码问题的节点状态图展示了问题的每个状态以及它们之间的转移关系。在八数码问题中，每个状态代表了一个拼图的布局，转移关系则表示了通过合法的拼图移动从一个状态到另一个状态的过程。通过一系列的状态转移，可以从初始状态逐步搜索到目标状态，如下所示：

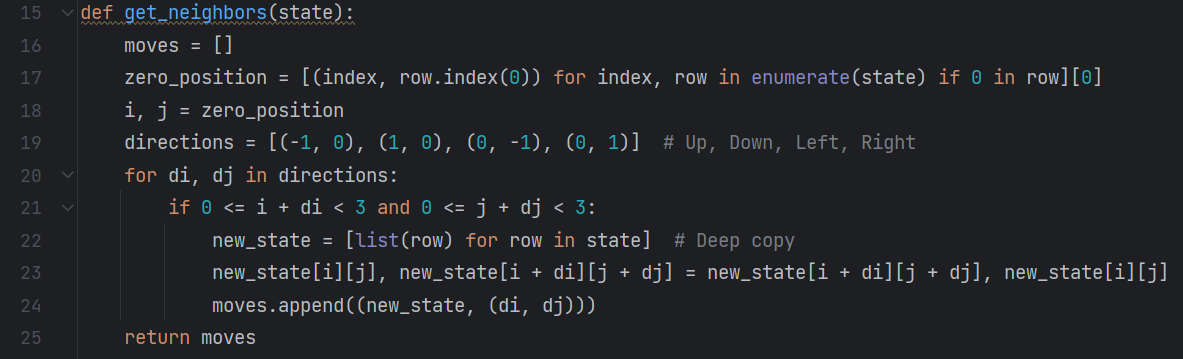


1. 编程解决八数码问题：

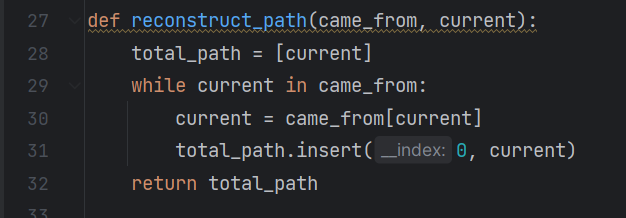
* manhattan\_distance(state, goal):此函数计算曼哈顿距离启发式，它是当前状态中的数字与目标状态中相应位置的数字之间的垂直和水平距离之和。函数遍历3x3网格中的数字，找到它在目标状态中的正确位置，并将垂直和水平距离的总和加到总距离上。



* get\_neighbors(state):此函数用于获取当前状态下所有可能的邻居状态。它首先找到数字0（空格）的位置，然后尝试将0上下左右移动，以生成所有合法的移动后状态。



* reconstruct\_path(came\_from, current):当A\*算法找到目标状态后，此函数用于重建从初始状态到目标状态的路径。它从目标状态开始，逐步追溯到初始状态，过程中使用came\_from字典来找到每个状态的前驱状态。

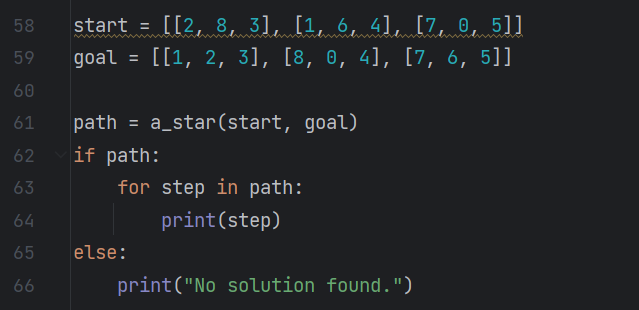


* a\_star(start, goal):这是A\*搜索算法的主函数。它初始化开放列表（优先队列），并开始从初始状态探索。对于每个状态，算法计算其所有邻居的g\_score（从初始状态到当前状态的实际成本）和f\_score（g\_score加上从当前状态到目标状态的启发式估计成本）。算法选择f\_score最小的状态进行扩展，直到找到目标状态或开放列表为空。如果找到目标状态，它使用reconstruct\_path函数来输出路径。

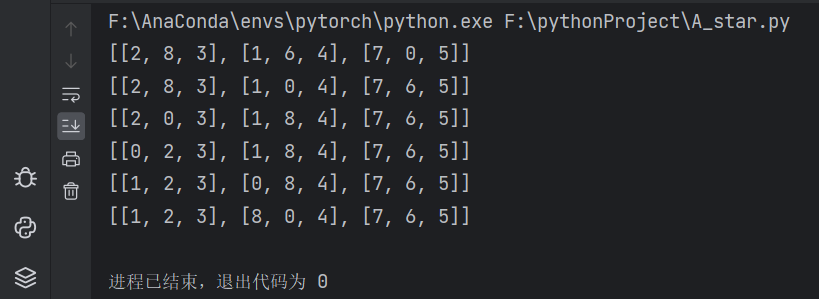


1. 运行结果：

测试用例：



运行结果：



1. **实验思考及心得**

通过这次关于启发式搜索和A\*算法的实验，我对解决复杂问题的算法设计与优化有了更深入的理解。在实际编写代码和调试过程中，我更加熟悉了如何利用启发式信息来指导搜索路径，从而有效减少搜索空间和计算时间。特别是曼哈顿距离的应用，不仅加快了搜索速度，也让我体会到合适的估价函数对于搜索效率的重要性。此外，通过逐步跟踪算法的每一个决策点，我加深了对A算法工作原理的理解，以及它如何在每一步选择最有希望的路径。这次实验不仅增强了我的问题解决能力，也激发了我进一步探索更多启发式搜索算法的兴趣。