计算机学院实验报告

|  |  |  |
| --- | --- | --- |
| 实验题目：八数码问题 | | 学号：201600181073 |
| 日期： | 班级： 智能16 | 姓名： 唐超 |
| **Email：**[**1984386166@qq.com**](mailto:1984386166@qq.com) | | |
| 实验目的：  熟悉和掌握启发式搜索的定义、估价函数和算法过程，并利用启发式搜索算法求解八数码难题，理解求解流程和搜索顺序。 | | |
| 实验环境介绍：  Python3.6 | | |
| 解决问题的主要思路：  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。算法的理论意义在于给出了求解最佳解的条件h(n)≤h\*(n)。对给定的问题，函数h\*(n)（n是变量）在问题有解的条件下客观上是存在的，但在问题求解过程中不可能明确知道，因此对实际问题，能不能使所定义的启发函数满足下界范围条件？如果困难很大，那么算法的实际应用就会受到限制。  对于八数码问题：在3×3组成的九宫格棋盘上，摆有八个将牌，每一个将牌都刻有1-8八个数码中的某一个数码。棋盘中留有一个空格，允许其周围的某一个将牌向空格移动，这样通过移动将牌就可以不断改变将牌的布局。这种游戏求解的问题是：给定一种初始的将牌布局或结构（称初始状态）和一个目标的布局（称目标状态），问如何移动将牌，实现从初始状态到目标状态的转变。  如果启发函数考虑任意节点与目标之间的距离信息，例如取h(n)=P(n)，P(n)定义为每一个将牌与其目标位置之间距离（不考虑夹在其间的将牌）的总和，那么能断定至少要移动P(n)步才能达到目标，因此有P(n)< =h\*(n)。 | | |
| 实验步骤：   1. 判断从初始状态到目标状态是否有解：八数码问题的一个状态实际上是0~9的一个排列，空格用0表示，对于任意给定的初始状态和目标状态，不一定有解。因为排列有奇排列和偶排列两类，排列只能在同类排列之间转化，而从奇排列不能转化成偶排列或相反。      1. 把初始节点放入Open表中，计算其f值； 2. 如果Open表为空，则问题无解，失败退出； 3. 把Open表的第一个节点取出放入Close表，并标记该节点为n； 4. 考察n节点是否为目标节点。如果是，则得到问题的解，输出搜索过程；      1. 如果节点n不可扩展，则转第3步；      1. 扩展节点n，计算每一个子节点的f值，并为每个子节点设置指向节点n的指针，将这些子节点放入Open表中；      1. 根据各节点的f值，对Open表中给的全部节点按照从小到大的顺序排序； | | |
| 实验结果展示及分析：  实验结果举例（初始状态："013425786"，目标状态："647850321"）：    启发式方法则是在有限的搜索空间内，大大减少尝试的数量，能迅速地达到问题的解决。但由于这种方法具有尝试错误的特点，所以也有失败的可能性。 | | |