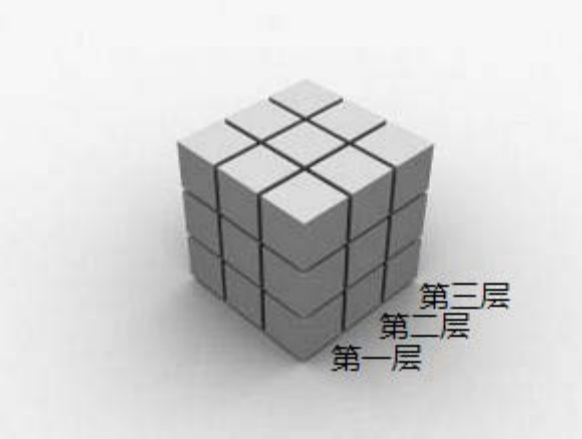
编程作业1

三立方数码问题

N皇后问题

P1:三立方数码问题 (50%)

三立方数码问题—本问题包括一个3x3x3的立方体，包括24个写有数字(1-24)的单位立方体以及一个空位（由0表示），两个障碍位（涂黑的部分，可以由-1表示）。与空位上、下、左、右、前、后相邻的棋子可以滑动到空位中，任何棋子都不能移动到障碍位中，且障碍位不可移动。游戏的目的是要达到一个特定的目标状态。



三立方数码问题

1. 实现A*搜索，as
2. 实现迭代深入A*搜索，idas
   - h1(n) = number of misplaced tiles（错位的棋子数）
   - h2(n) = total Manhattan distance（所有棋子到其目标位置的三个方向距离和）

Iterative deepening A*

Problem: A* is space inefficient

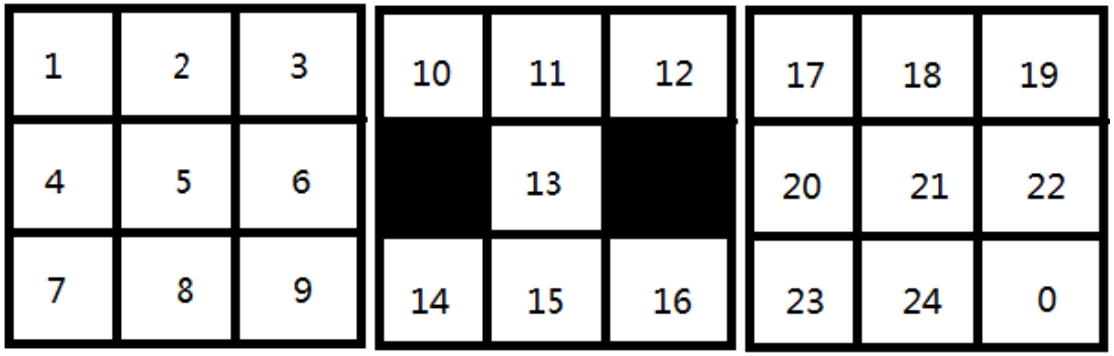
IDA*: Iterative deepening on f bound.

Algorithm 3 Iterative deepening A* search (IDA*)

```
1: d̂_limit ← d̂(s_0)
2: while d̂_limit < ∞ do
3:    next_d̂_limit ← ∞
4:    list ← {s_0}
5:    while list is not empty do
6:       s ← head(list)
7:       list ← rest(list)
8:       if d̂(s) > d̂_limit then
9:          next_d̂_limit ← min(next_d̂_limit, d̂(s))
10:      else
11:         if s is a goal then
12:            return s
13:         end if
14:         newstates ← apply actions to s
15:         list ← prepend(newstates, list)
16:      end if
17:   end while
18:   d̂_limit ← next_d̂_limit
19: end while
20: return fail
```

问题表示

状态由一个3维矩阵表示，0表示空位置，1-24表示棋子，-1表示障碍物。如图所示，在矩阵的(2, 1, 2)和(2, 3, 2)位置固定为-1。其余0-24数字任意放置。



问题表示

定义6个动作

- U代表up，即对空位0棋子上移
- D代表down，即对空位0棋子下移
- L代表left，即对空位0棋子左移
- R代表right，即对空位0棋子右移
- F代表forward，即对空位0棋子前移（靠近自己的方向）
- B代表back，即对空位0棋子后移（远离自己的方向）
- 所有动作均要合法

问题的目标是找到从初始状态到目标状态需要的**最短**移动序列

输出为动作序列，如 LDBRRB

作业要求

- 实现4个算法，即，使用启发函数h1(n)的A*算法，使用启发函数h2(n)的A*算法，使用启发函数h1(n)的IDA*算法，使用启发函数h2(n)的IDA*算法。

- 提交源代码和可执行文件（4个算法所以有4个代码和可执行文件），用readme文件写明如何运行你的程序以及对每个程序的说明。

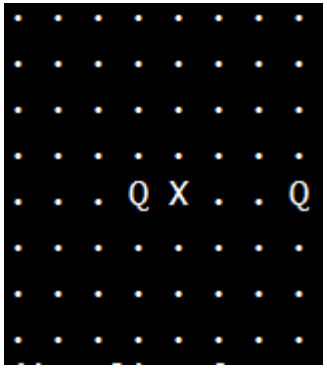
- 大致说明算法（A*和迭代A*）的时间复杂度和空间复杂度。比较使用不同的启发函数h1,h2的A*搜索及迭代深入A*搜索的性能，并分析性能差异的原因。

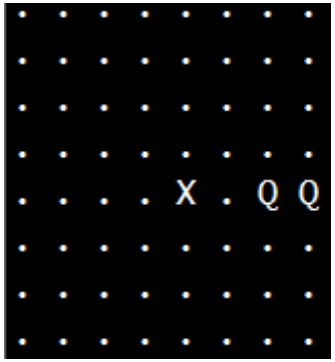
P2: N-Queen—N皇后问题 (50%)

本次实验要求：棋盘中某个位置有一个障碍。分别处于障碍相反两侧，并和障碍在一条直线上的两个皇后不会相互攻击。障碍不可以放置皇后。

问题描述

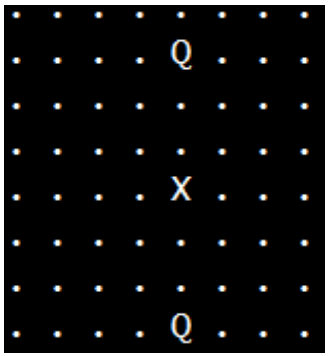
- 有一个N*N的棋盘，棋盘中第i行（i从1开始）第j列（j从1开始）的位置，记为(i,j)，i,j分别为行坐标和列坐标。点x(m, n)处是障碍。
- 现在有N个皇后，对任意两个皇后p,q，需要满足
  - p,q不在同一行上，除非p,x,q在同一行上，并且p,q分别在x的左右两边

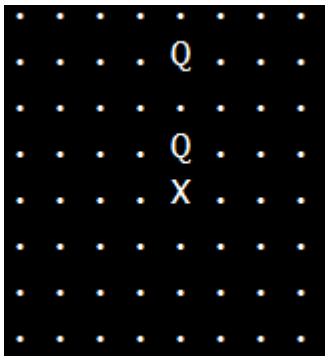




问题描述

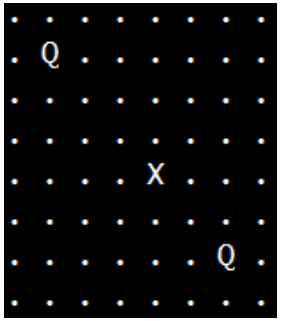
- 有一个N*N的棋盘，棋盘中第i行（i从1开始）第j列（j从1开始）的位置，记为(i,j)，i,j分别为行坐标和列坐标。点x(m, n)处是障碍。
- 现在有N个皇后，对任意两个皇后p,q，需要满足
  - p,q不在同一列上，除非p,x,q在同一列上，并且p,q分别在x的上下两边

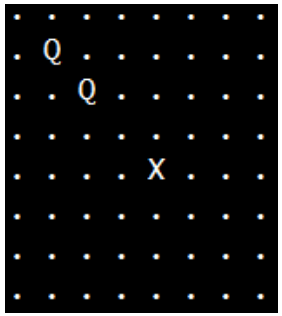


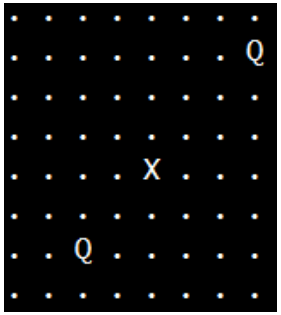


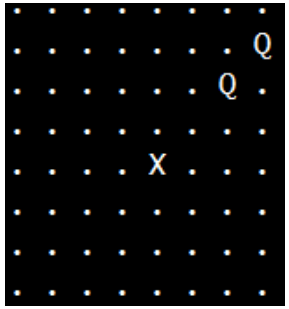
问题描述

- 有一个N*N的棋盘，棋盘中第i行（i从1开始）第j列（j从1开始）的位置，记为(i,j)，i,j分别为行坐标和列坐标。点x(m, n)处是障碍。
- 现在有N个皇后，对任意两个皇后p,q，需要满足
  - p,q不在同一条斜线上，除非p,x,q在同一条斜线上，并且p,q分别在x的两边。









作业要求

从四个算法中选两个算法实现：爬山算法、遗传算法、模拟退火算法以及CSP问题的局部搜索算法。如果可以找到其他更好的基于搜索的算法并实现，可以考虑适当加分。

说明文档要求：

- 算法思想
- 算法如何节省存储空间，分析空间复杂度
- 算法效率，分析时间复杂度，速度越快给分相应提高
- 实验结果说明
- 文档保存为pdf格式

Caution

Academic Integrity