

搜索技术课外作业

- [1] 利用 A* 搜索算法求解 8 数码问题，问最少移动多少次就可达到目标状态？画出搜索树，并在搜索树上标注出各状态的估价函数值。估价函数定义为 $f(n) = g(n) + h(n)$ ，其中 $g(n)$ 为节点 n 的深度，如 $g(S_0) = 0$ 。 $h(n)$ 为节点 n 与目标棋局不相同的位数（不包括空格），简称“不在位数”，如 $h(S_0) = 4$ 。

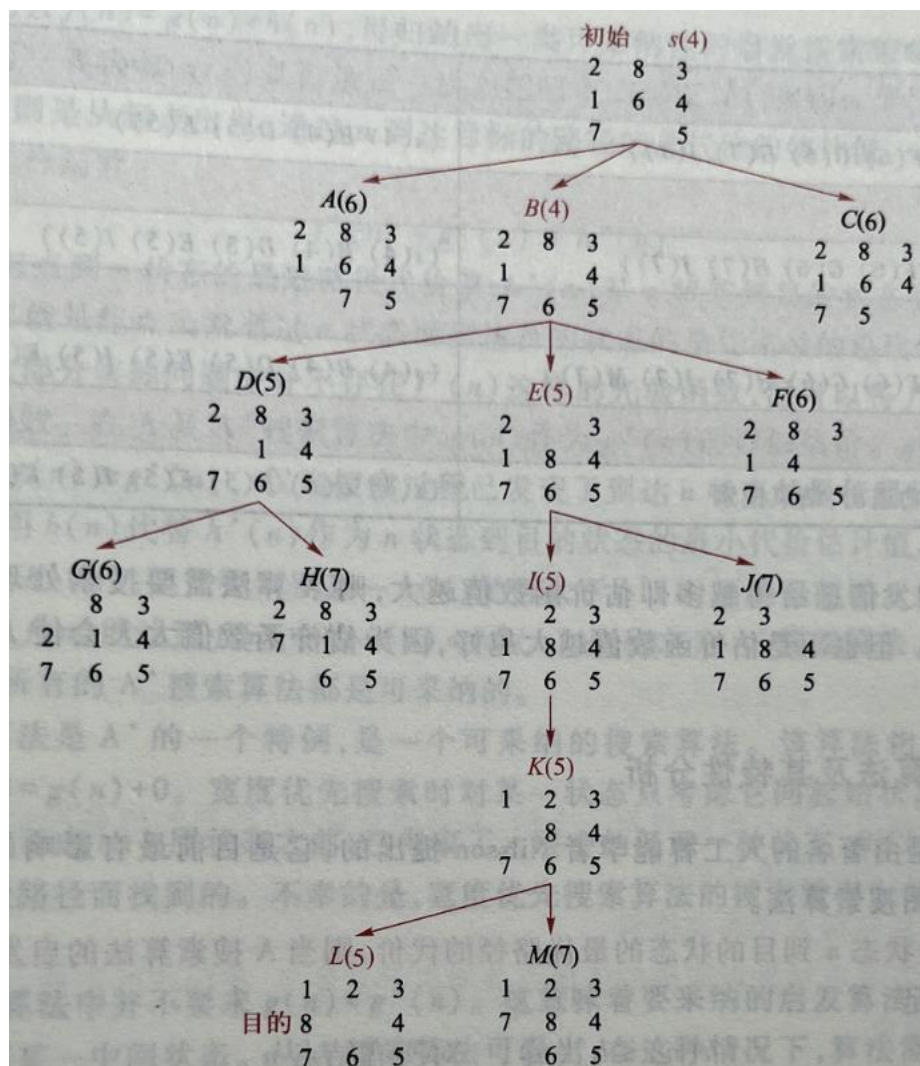
2	8	3
1	6	4
7		5

初始状态 S_0

1	2	3
8		4
7	6	5

目标状态

解答：搜索树如下：



因为不在位的将牌数小于实际归位的移动步骤数，即 $h(n) \leq h^*(n)$ 。所得的解路 (s, B, E, I, K, L) 为最优解路，其步数为状态 $L(5)$ 上所标注的 5。

[2] 对于8数码问题，令启发式函数 $h(n)$ 为所有数码的当前位置与其目标位置的曼哈顿距离之和。基于上述 $h(n)$ ，用A*搜索算法求解初始状态和目标状态如下图所示的8数码问题。对于空白格，规定其按照向上、向下、向左、向右的顺序进行移动。画出搜索图，并在图中标明所有状态的 f, g, h 值。

初始:

1	2	3
	8	4
7	6	5

目标:

2	8	3
1	6	4
7		5

注：每次行动的成本为1，左右（或上下）相邻数码的曼哈顿距离为1。可使用环检测。

解答：搜索图如下：

→

	2	3
1	8	4
7	6	5

$f: 4^*$

$g: 1$

$h: 3$

1	2	3
7	8	4
	6	5

$f: 6$

$g: 1$

$h: 5$

1	2	3
8		4
7	6	5

$f: 6$

$g: 1$

$h: 5$

→

2		3
1	8	4
7	6	5

$f: 4^*$

$g: 2$

$h: 2$

→

2	8	3
1		4
7	6	5

$f: 4^*$

$g: 3$

$h: 1$

2	3	
1	8	4
7	6	5

$f: 6$

$g: 3$

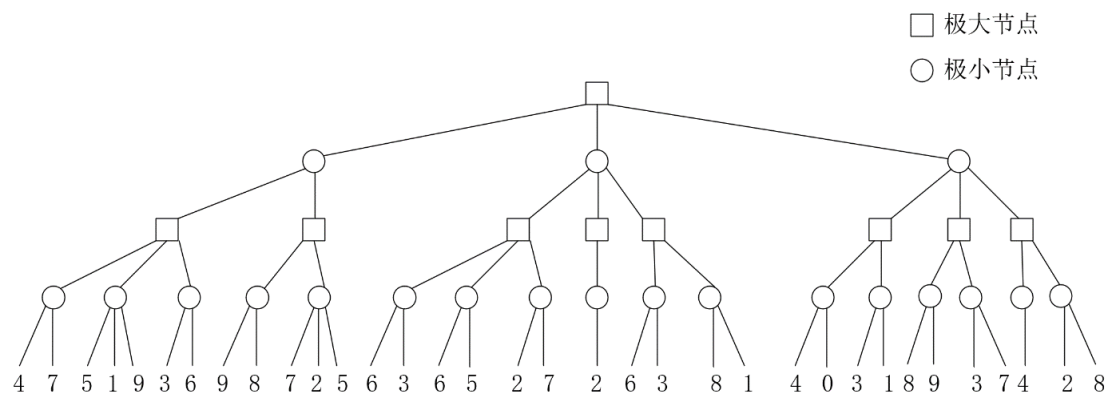
$h: 3$

→

2	8	3
1	6	4
7		5

目标

[3] 在下图所示的博弈树中，方框表示极大方，圆圈表示极小方。以优先生成左边结点的顺序来进行 α - β 剪枝搜索，试在博弈树上给出何处发生剪枝的标记。



解答：如下图，共有 8 处剪枝。

