

1. a) 状态: 当前平面上各划分区域的染色状态

初始状态: 划分的平面区域均未上色

目标状态: 划分的平面区域都上了四种颜色中的一种, 且相邻两区域颜色不同

行动: 将某个区域染上某种颜色

代价函数: 染色的次数之和

b) 状态: 当前三个容器中的水量 $[a_1, a_2, a_3]$ ($0 \leq a_1 \leq b, 0 \leq a_2 \leq 3, 0 \leq a_3 \leq 1$)

初始状态: $[b, 0, 0]$

目标状态: 某一容器中恰好有4升水, 即 $[4, a_2, a_3]$

行动: 将某个容器装满水, 清空, 或将水移动到其他容器

代价函数: 每次移动的水量的总和

2. a) \checkmark 令 $g(n) = 0$. $h(n)$ 为随层数单调递减的函数, 由于 $f(n) = g(n) + h(n)$ 随深度单调减, 故 A^* 算法

会优先扩展深层节点, 与深度优先搜索的顺序相同, 故深度优先搜索是一种特殊的 A^* 算法

b) \checkmark $\because h(n) = 0 \therefore h(n) \leq h^*(n)$ 显然成立, 故 $h(n) = 0$ 是可采纳的启发函数

c) \checkmark 设目标状态为 e , 则 $f(e) = f^*(e)$; 由于 A^* 算法的最优性, $f(n)$ 小的节点会先扩展; 节点 e 一定在最后被扩展, 故 $f(n) < f(e)$ 的所有节点 n 会先于 e 被扩展; 因此, OPEN 表中满足 $f(n) < f^*(e)$ 的任何节点 n , 最终都会被 A^* 算法选择去扩展。

3. a) 状态: 当前的密码 (仅由 A, B, C 三个字母组成, 且长度不超过 8 位)

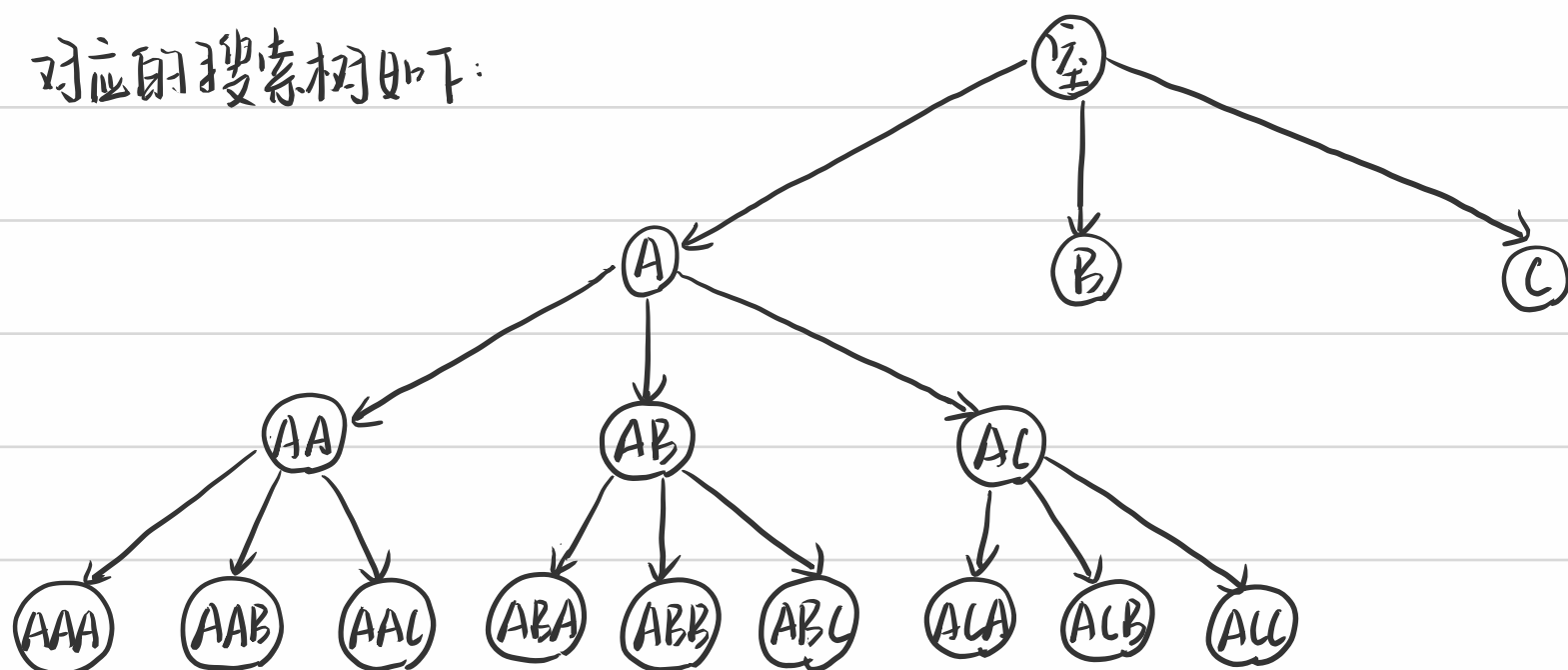
初始状态: 空, 当前无密码输入

目标状态: 6 个可用密码中的一个

状态之间转换的操作: 加入或删除一位密码

b) ALL

对应的搜索树如下:



c) AABCL

对应的搜索树如下:

