

Tree Puzzle 解题报告

长沙市雅礼中学 刘研绎

1 试题来源

TCO2014 Round2A 1000pts - Tree Puzzle

http://community.topcoder.com/stat?c=problem_statement&pm=13185

2 试题大意

给你一棵 n 个点的树。初始时，在0号节点上放了一个红色标记，在其他某些节点（可能全部也可能没有一个节点）上放置了一些黑色标记。现在你在这棵树上进行游戏，当一个有标记的节点和一个没有标记的节点相邻时，你可以将这个标记从当前节点移到那个空节点，目标是将红色标记移动到某个特定的位置。

要求你求出对于每个位置，是否可能将红色标记移动到此。

$n \leq 300$

3 试题分析

我们可以直接按照题意模拟。这个想法十分的简单但会面临很多的细节。考虑我们将红色标记往相邻的节点移动，如果这个节点恰好是空的，那么正好直接移动过去即可。如果这个节点不是空的，则考虑以这个节点为根的子树是否是满的。如果不是满的当然好办，肯定可以移动过去。

关键问题是如果这棵子树是满的怎么办。基本思想是：

1. 将当前的红色标记移开

2. 将某一个黑色标记移动出来

3. 再将红色标记移动进去。

这个过程看似简单，但实际上又有好几种情况，代码实现也十分复杂，这是官方题解给出的解答。

我们将提供一种新的思路，理解起来和代码实现相对简单。

如果以当前红色标记所在的点为根，对于一棵子树，我们只需要知道里面具体有多少黑色标记就能判断出能否进入。很自然地，我们会设想是否可以将每棵子树内黑色标记的个数暴力记录下来，只不过很可惜，这样记录的时间复杂度几乎是等同于整数拆分的方案数，根本无法记录状态。

仔细观察我们设计的状态：假设当前红色标记在节点 x ，在这一步操作之后红色节点移动到了与 x 相邻的节点 y 。那么在移动之前，我们只需要知道 y 部分中的黑色标记个数，其他子树的信息对我们完全没有意义。在移动之后，有用的信息变成了 x 部分中黑色标记的个数，也就是说需要知道 x 其他子树中黑色标记的总和，而这个数值等于总黑色标记数 - y 部分中的标记个数。

通过上面的观察我们可以发现，如果我们需要将红色标记从 x 移动到与之相邻的 y ，需要的信息仅仅是 y 部分的黑色标记个数。

不妨将状态设计在边上，对于一条边 (x, y) ，设 $f[(x, y)][k]$ 表示当前红色标记在 x ， y 部分的黑色标记个数为 k ，这种情况是否可能出现。

转移只有两种情况，一是继续往 y 所在子树移动，这样我们枚举往哪移并且保留多少黑色标记即可；另一种是到 y 之后在返回 x ，因为不存在黑色标记的变动所以 $O(1)$ 转移即可，每次移动利用判断目标部分是否放满黑色标记来判断是否能够移动。 dp 完成之后直接按照哪些点可以 dp 到输出答案即可。

问题至此便圆满解决。

时间复杂度 $O(n^3)$