

Union on Tree解题报告

宁波市镇海中学 邹逍遥

1 试题来源

Codechef October Challenge 2014

<http://www.codechef.com/OCT14/problems/BTREE>

2 试题大意

给定一棵树，边权均为1。

给出 Q 组询问，每次在 k_i 个点上放置警卫，警卫可以保护与他所处的位置距离不超过 r_j 的点。问有多少个点至少能得到一个警卫的保护。

3 数据范围

$$1 \leq n \leq 50000$$

$$1 \leq Q \leq 50000$$

$$1 \leq r_j \leq n$$

$$\sum k_i \leq 500000$$

4 算法介绍

4.1 $k_i = 1$ 的情况

假如每个询问只有一个警卫，那么这个问题可以使用点分治解决。

因为题目没有要求强制在线，我们可以使用离线算法，记录下每个点有哪些长度的询问。

进行点分治的时候通过记下每个子树的每个深度的点各有多少个，以及除了某个子树以外的每个深度的点的个数，那么就可以很方便地在 $O(size)$ 的时间内统计出经过这个点的答案对每个询问的贡献。

由于点分治递归层数不超过 $O(\log n)$ ，所以可以在 $O(n \log n)$ 的时间内解决。

4.2 $k_i > 1$ 的情况

在有多点的情况下就不是那么好处理了。

首先为了保证询问只和 $\sum k_i$ 有关，需要建出一棵包含询问点的虚树。

接着考虑虚树上的每条边，设它的长度为 l ，端点为 a, b ， a 的控制范围为 p_a ， b 的为 p_b 且 $p_a \leq p_b$ 。

假如 $p_a + l > p_b$ ，那么这两个警卫会有重叠部分，而重叠的那部分刚好可以转化为一个新的警卫（只不过这个警卫的贡献是负的）。因为这条链上一定存在一个点（有可能在一条边的重点上）使得两个警卫到这个点后能管得剩余距离相同，那么这一段就是完全重复的。于是这就可以通过新建一个询问并在点分治中一起解决。

假如 $p_a + l \leq p_b$ ，那么 a 警卫管制的区域会完全地被 b 警卫覆盖，那么就可以直接将 p_a 替换为 $p_b - l$ ，然后用新的 p_a 接着更新其他点。

但是在这里暴力更新是不行的，所以要通过一个虚树上的DFS算出每个点被哪个点完全地管制了。在计算出最后的答案后就可以在每条边上新建一个负询问来去除重复部分。

添加完新的询问以后每个询问就变成了许多个 $k_i = 1$ 的询问的和（或差），套用 $k_i = 1$ 时的点分治做法即可。

时间复杂度 $O(\sum k_i \log n + n \log n)$