

50 分: $n \leq 1000, p \leq 300$, 对于每一个修改, 暴力从上往下修改, 时间复杂度 $O(n)$, 查询 $O(1)$, 理论时间复杂度 $O(np)$;

100 分: $n \leq 50000, p \leq 100000$ 。

1. 线段树。因为是一棵树, 那么一颗子树的 dfs 序一定是连续的。所以可以在 dfs 序上进行区间修改, 区间查询。因为修改操作可以拆成深度, 加的个数和加的次数, 所以修改操作是可以合并的, 我们可以打懒标记就可以做到区间修改。查询时区间查询该子树对应的 dfs 序和就行了。时间复杂度 $(p \log(n))$;

2. 定期重构。神奇的根号算法。我们设计一个阈值 q , 表示每当修改次数达到 q 时就 $O(n)$ 重构整棵树。对于一个询问, 我们可以知道当前保留的询问个数不会超过 q 个, 我们分别讨论每个修改对于询问的影响。设询问的点为 x , 该修改修改的点为 y , 如果 y 的 dfn 在 x 子树的 dfs 序范围中, 说明 y 是 x 的子树, 答案加上 y 这颗子树产生的贡献。如果 x 的 dfn 在 y 子树的 dfs 序范围中, 说明 x 是 y 的子树, 答案加上 x 自己产生的贡献。设修改的次数为 p_1 , 询问的次数为 p_2 , 时间复杂度为 $((p_1/q) * n + p_2 * q)$; q 取根号 p 时, 时间复杂度为 $O((p_1/\sqrt{p}) * n + p_2 * \sqrt{p})$, 约为 $O(\sqrt{p})$;

3. 实在想不到, 会链剖的大神就打链剖吧。