

Solution

fatesky

2018 年 10 月 25 日

1 Hacker

考虑操作之间的贡献，当且仅当一个操作在另一个操作上面并且出现时间较早才会产生贡献，直接点分治 + 数点计算即可

首先假设修改全在询问前，考虑点分治的过程中，只需要计算分治重心在原树上父亲方向的链到分治重心的子树的贡献，那么可以直接考虑它父亲方向的一个点 u 和儿子方向的一个点 (v, val_v) ，并且 val_v 表示分治重心到 v 路径上的最小值，只需要 val_v 大于等于修改操作值即可，那么可以很简单统计

接下来考虑加入时间，那么就相当于加入一维限制，套上一个树状数组维护就可以了

2 Sand Lord

考虑问题要求的是什麼，用容斥原理的一般展开式化简，发现这个相当于计算至少有两个点颜色相同的方案数，然后在补集转化就可以直接算了

3 Ter Fly

首先考虑令 f_S 表示 S 的联通生成子图个数， g_S 表示 S 的生成子图个数，这个 g_S 很好求，那么只需要考虑求 f

定义乘法 $f * g$ 表示子集卷积，设 $F = \hat{f}, G = \hat{g}$ ，那么很容易发现一个集合幂级数方程：

$$1 + g = \sum_{k=0}^{\infty} \frac{f^k}{k!} = e^f \Rightarrow f = \ln(1 + g)$$

直接解占位幂级数方程