

### 第一题

发现一个节点只会被儿子感染或者父亲感染（废话）。

我们先看被儿子感染的情况，那么如果一个城市刚被感染病毒了或者已经被感染并且刚刚解封了，那么我们把他父亲扔到危险集合中，这个步骤最多扔  $n + q$  次。

如果一个节点可能被父亲感染了，那么如果一个城市刚被感染病毒了或者已经被感染并且刚刚解封了，我们把他扔到边缘集合中，这个步骤最多扔  $n + q$  次，而一个点被解封时候把它丢进它父亲的危险儿子集合（初始也要丢哦），这个步骤最多扔  $n + q$  次。

然后我们枚举危险集合，如果这个点的确会被感染（统计危险的儿子数量  $O(1)$  判断），那么就记录他，访问完后清空危险集合，这一步总复杂度  $O(n + q)$ 。

然后我们枚举边缘集合，如果这个点的确可传染（没有封城），那么就枚举他可能危险的儿子城市，试图感染（判断其是否未感染且未封城），枚举后清空这个危险儿子集合（只有枚举后才清空！），这一步总复杂度  $O(n + q)$ ，记得也要清空边缘集合。

可以发现用 `set` 维护即可  $O(n \log n)$ ，但是好像有重复元素没有事情，那就用 `vector` 吧，时间复杂度  $O(n + q)$ 。

### 第二题

对每一个节点维护一个数  $X_i$ ，表示节点  $i$  的父亲是否已经被染色，或者节点  $i$  的父亲的它的其余子树之中是否存在黑色的点。使用树链剖分之后，我们再对每一条重链，维护其延伸出去的轻边之中满足  $X_i = 0$  的轻边集合。

容易发现可以直接维护  $S_i$ ，使用树状数组扫描线可以轻易解决，时间复杂度  $O(n \log n)$ 。

### 第三题

我们先对  $a[i]$  排序。

取一个变量  $K$ ，我们令  $x[i] = \max(a[i] - K, 0)$ ，容易发现  $x[i]$  的和是不降的，我们取一个使得  $x[i]$  的和为  $C$  的  $k$ ，就可以了。

计算可行的  $K$  时，我们枚举其所在的区间  $[a[w-1], a[w]]$ ，然后可以发现在这个区间内其是一个一次函数，解出可行的  $K$  并判断是否可行即可。