

# 《tree》题解报告

By Hazyknight

## 子任务 1 (10 分)

首先我们注意到，由于原图是颗树，所以联通块个数=满足要求的点数-两端点都满足要求边数。每次暴力遍历所有的边进行查询就好了。

复杂度  $O(nq)$

## 子任务 2 (20 分)

这个子任务当中每个点度数都很少，由子任务 1 当中的结论我们可以直接对每个点维护其周围的所有边，由于度数少所以每次修改造成的影响就会少。然后全局用一个树状数组维护答案。

复杂度  $O(20q+(n+q)\log|w|)$

## 子任务 3 (10 分)

本子任务中除了一号点以外度数都是 1,所以只需想办法处理 1 号点的修改。由于权值修改时不降，所以我们可以用一个 set 维护 1 号点周围的点权，假设修改时候 1 号点点权由  $x$  变成了  $x+d$ ，那么只需要关注周围点权在  $[x, x+d]$  范围内的点，在树状数组上暴力修改即可，剩余的可以简单维护。由于权值不降，每次操作均摊是  $O(1)$  的。

复杂度  $O((n+q)\log|w|)$

## 子任务 4 (25 分)

给各种奇怪算法（比如什么度数小连度数大的，这样出度就是 sqrt 级别等。。。或者两 log 做法）和正解写 T 的点分。

## 子任务 5 (35 分)

受到子任务 3 和子任务 2 的启发，我们可以合并他们的过程，每次对于一个点维护这个点连边中另一端比这个点权值大的出边，每次暴力遍历这样的出边并看是否需要改方向，并在树状数组上修改，由于权值不降，复杂度也是均摊  $O(1)$ 。

复杂度  $O((n+q)\log|w|)$