

树链剖分_重链启发式合并

题目

CF 600E Lomsat gelral
CF 570D Tree Requests
CF 375D Tree and Queries
CF 246 E Blood Cousins Return
CF 208 E Blood Cousins

References

树链剖分_重链启发式合并

应用场景 多次询问子树中有多少节点满足某种性质，要求复杂度上界 $N \log N$ 。

具体场景

给出一棵树，每个节点有颜色，询问一些子树中不同的颜色数量。

考虑一个问题，能不能朴素地直接用一个数组把子树的每种颜色数量返回？

因为对于一个点，进入以它儿子为根的各个子树之前需要清空。同时，清空之后，在父节点时，又会面临需要使用全部的数据，因此我们又需要搜索一次所有的子树。

若使用1个数组，每个节点被遍历的次数是祖先链的长度（复杂度瓶颈）。

多个数组，没有清空带来的影响，但是合并复杂度、空间复杂度较高。

我们可以发现，只要我们可以保证，每个点被清空、再次搜索（访问）的次数，就可以保证复杂度。

我们看看这个过程是否有优化空间：先访问所有子树，然后清空。

不是所有子树答案必须清空。**最后一棵子树可以不清空。**

我们考虑让最后一棵子树尽可能大试试。即选择最后遍历**重儿子**。

模拟过程。

那，我们此时，不妨引入所有的重链剖分的概念（重儿子会实际使用，其他概念用于辅助证明）。

观察访问次数，5 6为例：每个节点，除了自己被访问的那一次以外，还会额外祖先链上轻节点访问。

因此，访问次数复杂度上界 $N \log N$ 。

核心思想：对询问离线处理，在树上合并时，调整了合并的顺序，用到了类似于并查集启发式合并的优化，故名dsu on tree。合并的逻辑与重链相关，因此，也称为重链启发式合并。

注意此时对dfs2，对答案相关的更新函数的实现。

题目

CF 600E Lomsat gelral

题意

给你一棵有根的树，根位于顶点 1。每个顶点都涂有某种颜色。

如果在顶点 v 的子树中，没有其他颜色比颜色 c 出现的次数更多，那么我们就称颜色 c 在顶点 v 的子树中占主导地位。因此，在某个顶点的子树中，可能会有两种或两种以上的颜色占主导地位。

顶点 v 的子树是顶点 v 和其他所有包含顶点 v 的顶点。

针对每个顶点 v 求顶点 v 的子树中所有支配色的总和。

分析

CF 570D Tree Requests

题意

一棵以1为根的树，每个节点上都有1个字母，有m个询问。每次询问v对应的子树中，深度为h的这层节点的字母，能否打乱重排组成回文串。根的深度为1，每个点的深度为到根的距离。

Reference

<https://www.luogu.com.cn/article/gl5u3xcj>

CF 375D Tree and Queries

题意

- 给定一棵 n 个节点的树，根节点为 1。每个节点上有一个颜色 c_i 。 m 次操作。操作有一种：
 1. u, k : 询问在以 u 为根的子树中，出现次数 $\geq k$ 的颜色有多少种。
- $2 \leq n \leq 10^5, 1 \leq m \leq 10^5, 1 \leq c_i, k \leq 10^5$ 。

CF 246 E Blood Cousins Return

CF 208 E Blood Cousins

[716E - Digit Tree](#)

References

[Sack \(dsu on tree\) - Codeforces Blog](#) 英语, 有题单

[\[Explanation\] dsu on trees \(small to large\)](#) 英语, 讲解细节

<https://www.cnblogs.com/Mrzdtz220/p/11723663.html>