树上合并类问题

程渝峰

SCU

August 19, 2024

知识概述

树上合并类问题一般是指启发式合并或线段树合并。启发式合并被称为树上启发式合并 (`dsu on tree`)

例如并查集的按秩合并，对于两个大小不一样的集合，我们将小的集合合并到大的集合中，而不是将大的集合合并到小的集合中，这个优化可以称为启发式合并算法。

线段树合并

线段树合并就是在转移过程中，需要将两个动态开点线段树的信息求和。对两颗树中同时存在的点暴力递归到儿子，其余点直接使用。由于合并相当于删除一部分点，所以均摊 $O(n\log n)$

例题

Description

给一棵树，每个点有颜色，询问每个子树里所有颜色种类数。

$1 \le n \le 10^5$

例题

遍历一个节点 u，我们按以下的步骤进行遍历：

- 先遍历 u 的轻（非重）儿子，并计算答案，但不保留遍历后它对 cnt 数组的影响；
- 遍历它的重儿子，保留它对 cnt 数组的影响；
- 再次遍历 u 的轻儿子的子树结点，加入这些结点的贡献，以得到 u 的答案。

可以看到对于每一个节点，除开遍历的一次外，在节点到根的路径上每 t 有一条轻边就会多访问一次，根据重链剖分的原理可以分析出每个点不会访问超过 $O(\log n)$ 次，总复杂度为 $O(n \log n)$

Description

有一棵 n 个结点的以 1 号结点为根的有根树。
每个结点都有一个颜色，颜色是以编号表示的，i 号结点的颜色编号为 c_i。
如果一种颜色在以 x 为根的子树内出现次数最多，称其在以 x 为根的子树中占主导地位。显然，同一子树中可能有多种颜色占主导地位。
你的任务是对于每一个 $i \in [1, n]$，求出以 i 为根的子树中，占主导地位的颜色的编号和。
$n \le 10^5, c_i \le n$

同上一道题的解决方法，通过 dsu on tree 维护出每个子树的 cnt 数组，同时维护最大出现次数和最大出现次数和即可

Description

快乐游戏鸡玩过的游戏是这样的：给定一棵 n 个结点的树，其中 1 号结点是根。每次玩家可以在树上行走，走过一条边需要 1 秒的时间，但只能往当前所在的点的某个儿子走，不能往父亲走。每次游戏需要从 s 号结点走到 t 号结点去。
玩家有一个总死亡次数，初始为 0。每个结点上有一个程序猿和一个参数 w_i，如果走到结点 i 的时候，当前总的死亡次数小于 w_i，那么玩家就会立刻死亡并回到起点 s，且该死亡过程不需要时间；如果总死亡次数大于等于 w_i，那么玩家就能熟练地对付程序猿从而安然无恙。注意每次游戏时不需要考虑 s 和 t 上的程序猿。
该游戏会进行若干轮，每轮会清空总死亡次数并给出一组新的 s, t。现在请你对于每一轮游戏输出走到 t 所需要的最短时间（单位为秒）。
保证每个询问的 s 可以到达 t。
$1 \le n, q \le 3 \times 10^5, 1 \le w_i \le 10^9$

通过题意，可以分析 s 到 t 的最短时间是先一直走，直到死亡次数等于 s 到 t 路径上的最大值，然后走 s 到 t

同样，可以分析出从某一个点出发每一次死亡到下一次死亡的时间是独立的，在树是一条链的情况下可以直接用单调栈维护。

对于任意树，启发式合并栈即可。

Description

你有一棵无根树，点数为 n，每个点有个点权 a_u，定义一条路径 $P(u, v)$ 的权值为经过的所有点的点权的异或和。定义一棵树是合法的，当且仅当树上所有简单路径的的权值都不为 0。
你可以对权值进行修改，可以改成任意正整数，问最少修改多少次才能让这棵树合法。
$n \leq 2 \times 10^5, a_i \leq 2^{30}$

CF1709E

首先可以简单的想到修改一个点后，所有经过这个点的路径都满足条件，把这个点修改为 $2^{u+random\_big\_val}$ 即可。
所以题意就转化为选一些点，使得树上所有值为 0 的路径至少包含一个被选的点。
考虑一个子树 u ，值为 0 的路径有几种情况

- 不经过 u 的路径：是否修改 u 没有影响
- lca 为 u 的路径
- 经过 u 但 lca 不为 u 的路径

对于后两种情况，只要存在一个 lca 为 u 的路径，就至少要选中 u 子树内的一个点，后两种情况所有的路径交中一定存在 u 这个点，所以选中 u 最优。
所以只需要在遍历时对所有子树 u 维护是否存在以 u 为 lca 的路径，如存在，删除这一子树。
用一个 set 维护子树内所有节点到根的路径的值，每遍历一个子树加入时枚举是否可能出现值为 0 的情况，用启发式合并即可

Description

一棵根为 1 的树，每条边上有一个字符（a 到 v 共 22 种）。一条简单路径被称为 Dokhtar-kosh，当且仅当路径上的字符经过重新排序后可以变成一个回文串。求每个子树中最长的 Dokhtar-kosh 路径的长度。

$1 \le n \le 5 \times 10^5$

显而易见，如果一个路径可以变为回文串，出现次数为奇数的字符最多有 1 个
二进制压缩出现次数为奇数的状态，就可以通过两点路径上所有点异或计算，启发式合并一下即可。

CF291E

Description

一颗 n 个节点的以一为根的树，树边上有小写字符串。
一个节点的串定义为从根到节点的路径上的字符串相连。
求串 t 在所有节点的串中出现的次数。
$1 \le n \le 10^5, 1 \le |S| \le 3 \times 10^5$

首先发现字符串长度和比较小，考虑把边权为一个字符串的边，转化为一跳边权为一个字符的路径。
现在就是查询有哪些路径和 t 匹配，考虑 dsu on tree，还需要知道一个点向上最多匹配多少前缀和后缀，二分 +hash 维护即可

CF208E

Description

有一个家族关系森林，描述了 n（$1 \le n \le 10^5$）人的家庭关系，成员编号为 1 到 n 。

如果 a 是 b 的父亲，那么称 a 为 b 的 1 级祖先；如果 b 有一个 1 级祖先，a 是 b 的 1 级祖先的 $k-1$ 级祖先，那么称 a 为 b 的 k 级祖先。

家庭关系保证是一棵森林，树中的每个人都至多有一个父母，且自己不会是自己的祖先。

如果存在一个人 z ，是两个人 a 和 b 共同的 p 级祖先：那么称 a 和 b 为 p 级表亲。

m（$1 \le m \le 10^5$）次询问，每次询问给出一对整数 v 和 p，求编号为 v 的人有多少个 p 级表亲。

CF208E

对于每一个询问，找到每一个 p 级祖先有几个 p 级后缀即可，离线查询，dsu on tree 维护即可

Description

给定一棵有 n 个节点的有根树 T，每个节点从 1 到 n 被编号，根为 1，每个节点都有一个字符。
对于一个节点 v 的子树，从 v 到子树中某一个节点的路径上的字符可以组成一个字符串（可能只有一个字符），定义所有不同的字符串的个数为 $\mathrm{dif}(v)$。
并且，对于每个节点 v 都有一个数值 c_v，求 $\mathrm{dif}(v) + c_v$ 的最大值，以及最大值的个数。
$1 \leq n \leq 3 \times 10^5$，$0 \leq c_i \leq 10^9$

考虑到对一棵子树的所有字符串建一棵 trie 统计节点数即可求得 $\mathrm{dif}(v)$。
在 dsu on tree 是保留重儿子的 trie，令当前点为 trie 新的根，暴力加入轻子树的节点即可。

CF715C

Description

给定一棵树，边权 $1 \le w \le 9$。给定一个数 M，保证 $\gcd(M, 10) = 1$。对于一对有序的不同的顶点 (u, v)，他沿着从顶点 u 到顶点 v 的最短路径，按经过顺序写下他在路径上遇到的所有数字（从左往右写），如果得到一个可以被 M 整除的十进制整数，那么就认为 (u, v) 是有趣的点对。

求有趣的对的数量。

$n \le 10^5$，$M \le 10^9$。

CF715C

首先有 $gcd(M, 10)$，令路径上得到的数为 x，若 $x \times 10^y \equiv 0 \pmod{M}$ 则 $x \equiv 0 \pmod{M}$
令根左右的两边为数中的第 $n, n+1$ 位，在 dsu on tree 时统计即可。

CF1009F

Description

给定一棵以 1 为根，n 个节点的树。设 $d(u,x)$ 为 u 子树中到 u 距离为 x 的节点数。

对于每个点，求一个最小的 k，使得 $d(u,k)$ 最大。

$1 \le n \le 10^6$

考虑在 dsu on tree 时维护每个深度的点有多少，因为在过程中只有加点的操作，所以同时维护最大的 $d(u, k)$ 和这种情况下最小的 k

CF375D

Description

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，询问在以 u 为根的子树中，出现次数 $\geq k$ 的颜色有多少种。
$2 \leq n \leq 10^5$，$1 \leq m \leq 10^5$，$1 \leq c_i, k \leq 10^5$。

CF375D

在 dsu on tree 时维护每个颜色的点有多少，同时维护次数 $\geq k$ 的颜色有多少种

CF570D

Description

给定一个以 1 为根的 n 个结点的树，每个点上有一个字母（'a'-'z'），每个点的深度定义为该节点到 1 号结点路径上的点数。每次询问 a, b 查询以 a 为根的子树内深度为 b 的结点上的字母重新排列之后是否能构成回文串。

$n \leq 5 \cdot 10^5$

CF570D

一个串能重构成回文串当且仅当出现次数为奇数的字符最多一个。二进制压缩出现次数的奇偶性，用桶维护，枚举所有可能答案二进制值查询

Description

给定一片森林，每次询问一个节点的 K-Son 共有个多少不同的名字。一个节点的 K-Son 即为在该节点子树内的，深度是该节点深度加 K 的节点。

$1 \leq n \leq 10^5$

CF246E

把每一个名字离散成一个数，在 dsu on tree 时用 set 维护每个深度有几种名字即可。

[IOI2011] Race

Description

给一棵树，每条边有权。求一条简单路径，权值和等于 k，且边的数量最小。
$1 \le n \le 2 \times 10^5$，$0 \le k, w_i \le 10^6$，$0 \le u_i, v_i < n$。

注意到 k 很小，在 dsu on tree 时对每一个路径长度维护一个最短的边数即可

Description

有一个家族关系森林，描述了 n（$1 \le n \le 10^5$）人的家庭关系，成员编号为 1 到 n 。
如果 a 是 b 的父亲，那么称 a 为 b 的 1 级祖先；如果 b 有一个 1 级祖先，a 是 b 的 1 级祖先的 $k-1$ 级祖先，那么称 a 为 b 的 k 级祖先。
家庭关系保证是一棵森林，树中的每个人都至多有一个父母，且自己不会是自己的祖先。
如果存在一个人 z ，是两个人 a 和 b 共同的 p 级祖先：那么称 a 和 b 为 p 级表亲。
m（$1 \le m \le 10^5$）次询问，每次询问给出一对整数 v 和 p，求编号为 v 的人有多少个 p 级表亲。

先用倍增求出该点 p 级祖先，对这个祖先求 p 级后代。这个问题直接 dsu on tree 维护每个深度的点数即可。

Luogu P4556

Description

首先村落里的一共有 n 座房屋，并形成一个树状结构。然后救济粮分 m 次发放，每次选择两个房屋 (x, y)，然后对于 x 到 y 的路径上（含 x 和 y）每座房子里发放一袋 z 类型的救济粮。
然后深绘里想知道，当所有的救济粮发放完毕后，每座房子里存放的最多的是哪种救济粮。
$1 \leq n, m \leq 10^5$，$1 \leq a, b, x, y \leq n$，$1 \leq z \leq 10^5$。

树上差分，这样一条路径上的操作就成了路径端点上加，lca 处减，一个点所有的救济粮就是子树内所有的贡献和。用线段树维护每个救济粮有多少，每次与儿子的线段树合并即可。

CF932F

有一颗 n 个节点的树（节点从 1 到 n 依次编号），根节点为 1。每个节点有两个权值，第 i 个节点的权值为 a_i, b_i。
你可以从一个节点跳到它的子树内任意一个节点上。从节点 x 跳到节点 y 一次的花费为 $a_x \times b_y$。跳跃多次走过一条路径的总费用为每次跳跃的费用之和。请分别计算出每个节点到达树的每个叶子节点的费用中的最小值。
注意：就算根节点的度数为 1，根节点也不算做叶子节点。另外，不能从一个节点跳到它自己.
$2 \le n \le 10^5$，$-10^5 \le a_i \le 10^5$，$-10^5 \le b_i \le 10^5$。

CF932F

首先，有一个很显然的 $O(n^2)$ 的 dp，考虑优化。
对于一个转移，他的贡献形如 $kx + b$ 的形式，对于这种形式，一般使用李超线段树或凸包维护。
显然，对于这道题，凸包的合并并不好做。所以就可以李超线段树合并完成维护。

Description

小 C 有一棵 n 个结点的有根树，根是 1 号结点，且每个结点最多有两个子结点。
定义结点 x 的权值为：
1. 若 x 没有子结点，那么它的权值会在输入里给出，保证这类点中每个结点的权值互不相同。
2. 若 x 有子结点，那么它的权值有 p_x 的概率是它的子结点的权值的最大值，有 $1-p_x$ 的概率是它的子结点的权值的最小值。
现在小 C 想知道，假设 1 号结点的权值有 m 种可能性，权值第 i 小的可能性的权值是 V_i，它的概率为 $D_i(D_i>0)$，求：

$$\sum_{i=1}^{m} i \cdot V_i \cdot D_i^2$$

你需要输出答案对 998244353 取模的值
$0<p_i \cdot 10000<10000$，所以易证明所有叶子的权值都有概率被根取到。

设 $f_{i,j}$ 为 i 节点出现 j 的概率， l, r 分别为左右儿子，有转移

$$
\begin{aligned}
f_{i,j} =& f_{l,j}(p_i \sum_{k=1}^{j-1} f_{r,k} + (1-p_i) \sum_{j+1}^{m} f_{r,k}) \\
&+ f_{r,j}(p_i \sum_{k=1}^{j-1} f_{l,k} + (1-p_i) \sum_{j+1}^{m} f_{l,k})
\end{aligned}
$$

式子有关前缀和和后缀和，可以用线段树合并的操作来进行转移，从下到上转移，求出根

[JOISC 2021 Day4] 最悪の記者

Description

有 n 个人，第 i 个人的分数为 h_i ，修改 h_i 需要 c_i，要求 $h_i \geq h_{a_i}$ ，求最小修改费用。

$2 \leq n \leq 2 \times 10^5, 1 \leq h_i, c_i \leq 10^9$

首先，偏序关系的结构构成了一个内向基环森林，环上所有点的权值必定相等，所以先把环看成一个点。

对于一颗基环树，修改权值的最优方案一定是修改成树上另一个点的权值或一个极大值。

设 $f_{u,i}$ 是 u 最后权值为 i 的最小修改代价，可以轻松 $O(n^2)$ dp。

用线段树合并优化 dp，考虑到区间加在线段树合并中不好实现，先假设所有点都有修改，状态改为最大的不修改的代价和，就成了单点修改。

Description

给你一棵无根树，每个点 i 有两个属性 t_i, v_i。
定义有向路径 $i \to j$ 的 $f_{i,j}$ 为:
若 $i \to j$ 上 t_x 最小的点为 x 且 $v_j \le v_x \le v_i$，则 $f_{i,j} = x$。
否则，$f_{i,j} = 0$。
求 $\sum\limits_{i=1}^{n} \sum\limits_{j=1}^{n} f_{i,j}$。
对于 100% 的数据，t 互不相同，$1 \le n \le 10^5$，$1 \le t_i, v_i \le 10^9$。

因为 x 是路径上 t_x 最小的点，从大到小枚举所有点加入图中，加入后 x 所在连通块经过 x 的路径 x 必然是 t_x 最小点。
注意到，一条路径上只与端点和 x 有关系，所以只需要维护一个连通块里所有的 v_i，每次线段树合并同时统计答案即可。

Description

给定一棵 n 个节点的，以 1 为根的树，每个点有权值 a_i。
对于一个点 u，定义函数 $f(u,v,d)$ 表示在 u 的子树内选择一些点（至少需要选取一个点），选出的点中最大权值为 v，且它们编号的最大公约数为 d 的方案数。
给定一个常数 k 和一个序列 b，对于每个点 u，你需要求出 $\sum_{k|i}^{n}\sum_{j=1}^{n} f(u,i,j)\times b_j$ 的值，其中 $k|i$ 表示 k 可以整除 i。由于该值可能很大，所以你需要输出其对 998244353 取模的结果。
$1\le a_i\le n\le 10^5$，$1\le b_i\le 10^9$。

首先把式子换成 $\sum_{j=1}^{n} b_j \sum_{k|i}^{n} f(u,i,j)$，由于恰好等于 gcd 这种式子不好统计，考虑容斥，计算 j 是 gcd 的因数的方案，记作 $g(u,i,j)$。
对于计算 j 是 gcd 的因数的方案，把所有编号是 j 的倍数的点建虚树，在虚树上选方案的 gcd 一定是 j 的倍数。同时 i,j 是独立的，所以可以计算 $\sum_{k|i}^{n} g(u,i,j)$ 来容斥 $\sum_{k|i}^{n} f(u,i,j)$。
因为在虚树上可以忽略 j 的影响，$g(u,i)$ 就可以 dp 计算。线段树合并优化 dp，同时维护 i 为 k 倍数位置的和方便统计答案。

Description

给定一棵 n 个结点的有根树 T，结点从 1 开始编号，根结点为 1 号结点，每个结点有一个正整数权值 v_i。
设 x 号结点的子树内（包含 x 自身）的所有结点编号为 $c_1, c_2, \dots, c_k$，定义 x 的价值为：

$$val(x) = (v_{c_1} + d(c_1, x)) \oplus (v_{c_2} + d(c_2, x)) \oplus \cdots \oplus (v_{c_k} + d(c_k, x))$$

其中 $d(x, y)$ 表示树上 x 号结点与 y 号结点间唯一简单路径所包含的边数，$d(x, x) = 0$。$\oplus$ 表示异或运算。
请你求出 $\sum_{i=1}^{n} val(i)$ 的结果。
100% 的数据：$1 \le n, v_i \le 525010, 1 \le p_i \le n$。

考虑用 trie 维护子树内所有的 $v_i + d(i, x)$ 合并到父亲需要将所有的数加 1，一般的 trie 从高位维护比较困难，考虑从低位维护，这样最低位为 0 的变成最低位为 1 的，最低位为 1 的会进位后最低位变成 0。具体操作就是交换左右儿子，继续递归到左儿子。这样左 trie 的结点数不会增加，然后同线段树合并即可。

[NOI2020] 命运

Description

给定一棵树 $T = (V, E)$ 和点对集合 $\mathcal{Q} \subseteq V \times V$，满足对于所有 $(u, v) \in \mathcal{Q}$，都有 $u \neq v$，并且 u 是 v 在树 T 上的祖先。其中 V 和 E 分别代表树 T 的结点集和边集。求有多少个不同的函数 $f : E \to \{0, 1\}$（将每条边 $e \in E$ 的 $f(e)$ 值置为 0 或 1），满足对于任何 $(u, v) \in \mathcal{Q}$，都存在 u 到 v 路径上的一条边 e 使得 $f(e) = 1$。由于答案可能非常大，你只需要输出结果对 $998,244,353$（一个素数）取模的结果。

$n \leq 5 \times 10^5$，$m \leq 5 \times 10^5$。输入构成一棵树，并且对于 $1 \leq i \leq m$，u_i 始终为 v_i 的祖先结点。

[NOI2020] 命运

考虑 dp，dp[x][y] 表示 x 子树内的边状态已经确定，下端点在子树内且没有被满足的条件中，上端点最深的深度是 y，特殊地，如果子树内的条件都满足那么 $y=0$。记录最深是因为深得满足浅的也就会跟着满足。
转移为：

$$dp'[x][i]=\sum_{j=0}^{dep_x}dp[x][i]\times dp[y][j]+\sum_{j=0}^{i}dp[x][i]\times dp[y][j]+\sum_{j=0}^{i-1}dp[y][i]\times dp[x][j]$$

这个式子里只有单点和前缀和，考虑线段树合并优化即可。