点分治 & 边分治

wxt1221

2023 SWOI 寒假集训

Table of Contents - 目录

引入

P3806

题意：给定一颗具有 n 个节点的树，边上有权值，一共给定 m 次询问，每次询问给定 k，要求求出树上距离为 k 的点对是否存在。

树上任意两点的路径是可以分为两条从一个节点到其某个祖先节点的路径。考虑递归到节点 i，对于此节点，为了进行统计，可以划分为两个部分，一种是经过 i 号节点，另一种是没有经过 i 号节点，很明显没有经过 i 号节点的路径一定在 i 号节点某个儿子的子树内，可以进行递归处理，接下来考虑如何处理，很明显经过 i 号节点的路径一定是经过两个儿子的，定义 dis_b 表示 b 号节点到 i 的距离，很明显这是一条树上从一个节点到其某个祖先节点的路径，我们要找的是二元组 (a, b)，使得有 $dis_a + dis_b = k$，这样直接处理是 $O(n^2)$，但是可以开数组 tf_b 表示是否存在 v 使得 $dis_v = b$。

时间复杂度

很明显每一层递归，设以 i 节点为根的子树大小为 siz_i ，很明显每次递归是 $O(siz_i \times m)$ ，总时间复杂度不难得出，设树深度为 h ，则时间复杂度为 $O(nhm)$ 。
这个时间复杂度依赖于数据，深度最坏可能达到 n ，于是最坏时间复杂度变成了 $O(n^2 m)$。
所以是否可以适当变换树的结构使得深度可以减小到 log_n 呢？

重心

log 的时间复杂度关键在于分成一半，如果可以在递归过程中更换一个根节点，即可，因为这样更改根节点实际上只是更改了一条边，也就是从原来的根节点到他的父亲的边，所以子树内点间距离不会改变，所以怎么选呢？分成一半，在于相等，如果能确保所有儿子的子树大小尽可能相等，即可，换句话说也就是子树大小的最大值最小，注意这里讨论无向边，可以理解为以这个节点作为根节点的时候，求出一个 W_i 是 $\max(siz_j)\ j \in son_i$ ，求出最小的 W_i ，这个点即为重心。
$mss(u)$ 表示 u 的最大子树大小。
形式化的说，找到一个 u 使得 $mss(u)$ 最小。

引理 1

结论：
某个树的重心一定满足他的最大子树大小不大于整棵树的一半，反命题同样成立。

论证：
充分性：
$siz_u(v)$ 表示以 u 为根，v 子树大小。
$mss(u)$ 表示 u 的最大子树大小。
反证法证明，u 为树的重心，假设存在与他相邻的一个节点 v 满足 $siz_u(v) > \frac{n}{2}$ ，于是有 $siz_v(u) = n - siz_u(v) < \frac{n}{2}$ ，于是 $siz_v(u) < siz_u(v) = mss(u)$ ，因为 $siz_u(v) = 1 + \sum siz_v(w)\ (w \in son_v)$，所以有 $siz_v(w) < siz_u(v)\ (w \in son_v)$ ，推出 $mss(v) < siz_u(v) = mss(u)$ ，与重心定义不符。
必要性：
如果 $mss(u) < \frac{2}{n}$ ，则说明对于每个儿子 v 都存在 $siz_u(v) < \frac{2}{n}$ ，于是有 $\sum siz_v(w) < \frac{2}{n}\ (w \in son_v\ \&\ w \neq u)$ ，可以推出 $siz_v(w) < \frac{2}{n} (w \in son_v\ \ \&\ \ w \neq u)$。

引理 2

结论：
如果一棵树有两个重心，则这两个重心一定相邻，并且树有偶数个节点，可以被划分为两个大小相等的分支，每个分支各自包含一个重心。

论证：
很明显对于两个重心 u, v 有 $mss(u) = mss(v)$，u 的最大子树肯定包含 v，假设法证明，假设 w 是 u 的最大子树，则有 $mss(u) = siz_u(w) < siz_v(u)$，很明显 $siz_v(u) \leq mss(v)$，则有 $mss(u) < mss(v)$，矛盾。
设 u 到 v 经过 k 个中间节点，于是有 $wss(u) = siz_u(v) = k + mss(v)$，明显有 $k = 0$。
因为 $siz_u(v) = siz_v(u)$，又因为 $siz_u(v) + siz_v(u) = n$，所以 $siz_u(v) = siz_v(u) = \frac{2}{n}$，结论不证自明。

引理 3

结论:
树至少一个重心，至多有两个重心。

求重心在做求最小值操作，肯定存在最小值。不可能存在三个重心，因为他们两两相邻，会形成环，树怎么可能有环？

引理 4

结论：
假设树上所有边权为 1，则记 dis_i 表示所有节点到 i 号节点的距离和，则重心 u 的 dis_u 最小，反命题成立。

论证：
很明显对于非重心点 u，有 $wss(v) > \frac{2}{n}\ (v \in son_v)$，那向 v 移动会使 dis 减少 $siz_u(v) - (n - siz_u(v) = 2siz_u(v) - n > 0$，结论得证。

实现

根据定义以及引理 1 即可。

优化后

只用每次 dfs 时转移到子树重心即可，寻找重心的时间复杂度仅增加常数，所以优化后时间复杂度达到 $O(nlog_nk)$。

引入

P3806

考虑分治的思想，是选择合适的边使得树裂成两半，然后递归处理，路径只有经过这条边和不经过这条边，不经过可以递归处理，怎么处理经过的呢？

假设选择了第 i 号边，则可以处理出 dis_j ，若 j 在 u_i 的子树内，dis_j 就是 j 到 u_i 的距离，对于 v_i 同理。所以实际上是在找 $dis_a + dis_b + w_i = k$，其中 a 在 u_i 的子树，b 在 v_i 的子树内。

但是递归的时间复杂度呢，实际上，是不可能找到边一定能均匀的将树分成两半的，比如说菊花图。

优化

但是，是否可以变换树的形态呢，很明显若原树为二叉树时，时间复杂度可以得到保证。
因为考虑重心，假设选的边一定是有一端在重心上，根据性质 1，可知会把树分成 $\frac{2}{3}$ ，保证了最坏的时间复杂度，在 n 不是太大时，可以认为是 $O(log_n)$ 。

总结

选择适合的边分治整棵树，适当对树进行变形可以保证时间复杂度。