树链剖分_长链剖分

长链剖分和重链剖分一样，是把一棵树分成若干条不相交的链。

长链剖分与重链剖分相比，将原来以子树**规模最大**的指标换成了子树**深度最大**。

子树深度最大值为子树内，所有节点的深度最大值。

为了方便称呼，我们使用长儿子与轻儿子进行称呼。对应的，有长边、轻边，长链。

首先，我们只是修改了选择的标准，所以，实现应该变化不大。

区别：子树规模可以直接得到，维护size数组。

但是，最大深度不行，需要另外开一个数组。

性质

一个节点到根的路径，最多经过$O(\sqrt{n})$个轻边。即此时不具备 $logN$ 性质。

证明

对于一个节点，深度为 k，它往上跳一个轻边，则跳完后的子树大小，会加上至少 $k+1$ （不然无法形成长链）。

那么如果我们跳了x 条轻边，此时树的大小，最少为 $\sum_{i=1}^{x}(i+1)=\frac{x(x+1)}{2}$ 个节点。

因此，若总节点数是 N，那么，一个节点到根的路径，最多经过 $O(\sqrt{n})$ 条轻边。

性质 所有长链长度之和为 N 。

性质

每条长链的深度链头不超过长链的长度。

所以，如果使用长链剖分进行祖先链查询，是劣于重链剖分的。

故通常而言，树剖均使用重链剖分，提到树剖，大家默认是重链剖分。（实链剖分LCT通常直接称为LCT）。

1个场景除外：k级祖先，理论复杂度优秀。

k级祖先

我们可以处理 k 级祖先的算法：

1.倍增。预处理 $NlogN$。查询 $QlogN$。

2.重剖。预处理 N。查询 $QlogN$。

我们思考倍增的局限性：必须要花费log的时间才能往上跳。能否加速？

性质 任意一个点的 k 级祖先所在链的链长一定大于等于 k 。

简单易证 这个点 k 级祖先所在的链长度肯定大于等于它到这个节点，否则不存在。

既然长链长度大于等于k，能不能直接跳k步？

不行。举反例。

dfs序的局限。我们也不可能每个点单独开数组记录全部祖先链。

但是，我们刚刚提到，所有长链长度之和为 N 。

如果记录长链头往上、往下跳的节点，这个复杂度可以接受。

如果我们记录了这个数组，能否直接跳k步获得 k级祖先？

不能。举反例。（记录的长度可能不够）

那么，我们不防直接跳 $2^{highbit(k)}$ 步试试。

设 $k'=k-2^{highbit(k)}$ (即我们还要跳 k' 层)。

此时，我们发现当前节点所在长链 链长一定大于 $2^{highbit(k)}$ 。

而我们只需要跳 k' 步，长度足够，可以跳。

现在，如果这个节点的 k' 级祖先在它的链顶之下，那么可以用链顶向下的数组求出

如果在这个节点的 k' 级祖先在它的链顶之上，就用向上的数组求出

然后我们就可以 $O(nlogn)$ 预处理,每次询问 $O(1)$ 在线求 k 级祖先了。

实现

dfs2的实现需要**举例子**。

大家可以通过该题目检验理论复杂度与实际复杂度的区别。提供一点洛谷record供参考：

重链剖分最优解 https://www.luogu.com.cn/record/167840115

看了前两页最优解，有可读性的代码中，正常解法重链剖分比较多。

主要原因：常数太大，导致理论复杂度的优势并不明显。

优化dp

但是，参考重链启发式合并，用其作为工具，优化dp会有奇效。

一般来说，优化的dp方程需要有一个维度与深度相关。

写出dp方程。

我们思考在长链上， $dp[u][\dots]$ 与 $dp[son[u]][\dots]$ 的区别：

$dp[u][\dots]$ 比 $dp[son[u]][\dots]$ 多一个1，其余部分均相同。

如果能够快速复制一次，那么，我们继承长儿子状态只需要O(1)。

指针。

举例。

如果从长儿子合并，显然是移动数组。

但是，我们递归本身就是从上往下，我们直接预备好数组位置就可以了。

对于非长儿子，那么，递归处理后，暴力合并。它一定是另一条长链的起点。

合并代价：$O(deep = length(\text{长链}))$ 。

总代价： $O(\sum_{deep}) = O(N)$

思考，为什么重链剖分看起来更优秀，为什么在我们此处使用长链剖分方式优化反而有奇效？

重链启发式合并之后，还会继续访问，总访问次数 log级。

为什么重链剖分不具备这样的性质：

1.长链剖分优化dp的时间保证来自深度信息，树链的深度连续，使得可以支持快速复制，同时，合并之后，我们只关心深度，不关心节点，彻底的将子树合并。

2.重链剖分没有维护深度，因此需要额外维度维护深度。

合并后为了处理子树，需要清空。

重链剖分不能支持快速复制。

题目

P5903 【模板】树上 K 级祖先

模版。

CF1009F Dominant Indices

题意

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。

分析

首先，如果我们知道 u 的所有子树相对于 u 的各种深度的种类数，那么，我们是可以合并得到 u 的答案的。

因此，有一个朴素的思想，使用 $dp[i][j]$ 表示 i 节点，深度为 j 的点的数量。

转移显然。

对其使用长链剖分优化 dp 即可。

Reference

https://www.luogu.com.cn/article/pyxfuiho

https://www.cnblogs.com/KellyWLJ/p/18015611

https://zhuanlan.zhihu.com/p/588385339

CF 526 G Spiders Evil Plan

http://codeforces.com/problemset/problem/526/G

Luogu_P5904 [POI2014] HOT-Hotels 加强版

https://www.luogu.com.cn/article/i1d0zqoo

Luogu_P3899 [湖南集训] 更为厉害

分析

Reference

https://www.luogu.com.cn/article/g61fy8fa

CF150E Freezing with Style [黑题]

Reference

https://www.nowcoder.com/discuss/353147962983391232?sourceSSR=users

https://www.cnblogs.com/jiangchen4122/p/17455781.html

https://www.cnblogs.com/YunQianQwQ/articles/heavy-light-decomposition2.html