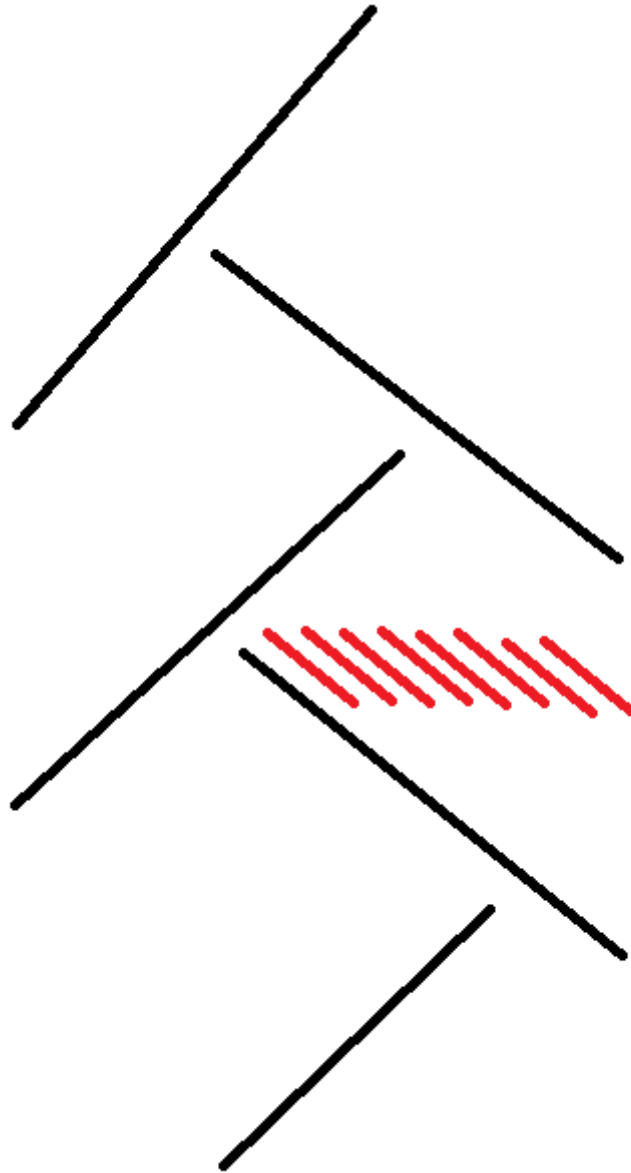
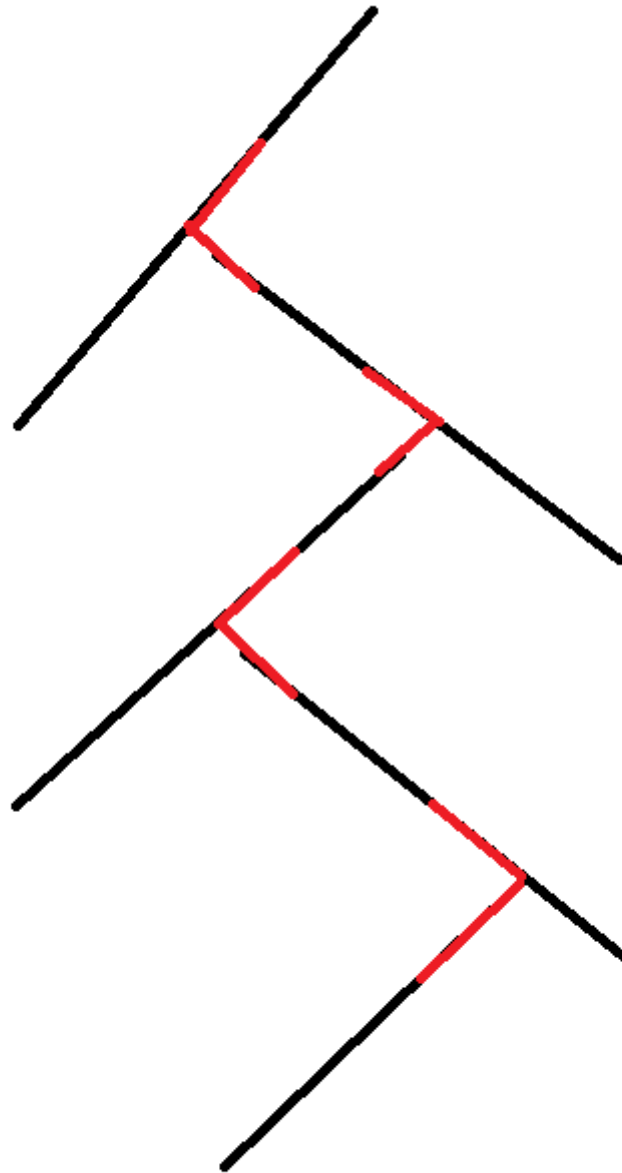


# TEST\_64

考虑对原树进行树链剖分，之后每条极长重链可以看做是一个序列，维护该序列上每个长为  $x$  的子路径的字符串哈希。定义哈希数组  $f_i$  表示区间  $[i, i + X - 1]$  的哈希。



询问时由树链剖分可以将路径拆分为  $O(\log n)$  跳极长重链上的询问。对于所有严格被包含在极长重链中的长为  $X$  的子路径，问题变为询问哈希数组  $f$  的一个区间中  $f_i$  等于给定字符串  $S$  的哈希值的元素个数。



对于没有严格被包含在极长重链中，但在询问路径上的长为  $X$  的子路径，我们可以在两条极长重链相连接处暴力，因为这里每个连接处只需要考虑  $2(X - 1)$  长度的字符串，相当于进行了  $O(\log n)$  次长为  $O(X)$  的两个字符串的匹配，这部分使用暴力字符串哈希或者 KMP 的方法，单次时间复杂度为  $O(X \log n)$ 。

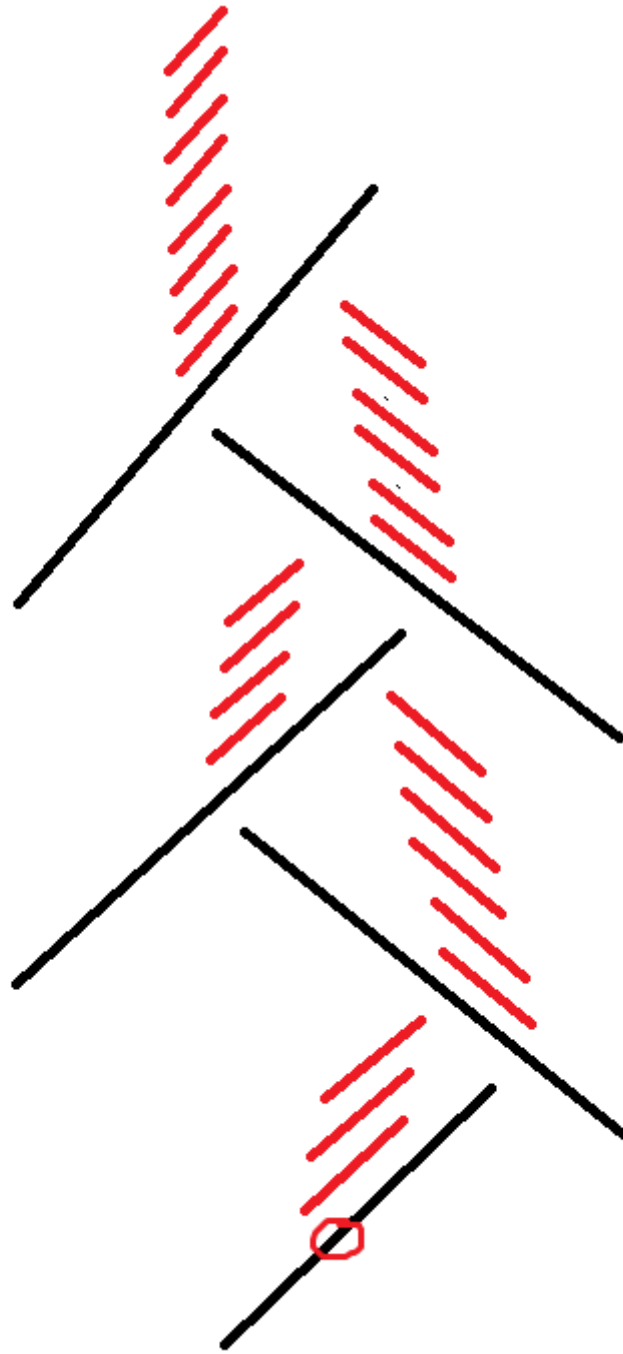
修改时直接修改所有受到影响的极长重链的哈希数组以及建立在其基础上的数据结构即可，因为修改的字符串长度也为  $X$ ，所以这里对于每条极长重链我们需要修改不超过  $3X - 2$  个位置，单次修改需要修改  $O(X \log n)$  个位置的  $f$ ，对应  $O(\log n)$  个连续的区间上。

维护区间有多少  $f_i$  等于给定数可以使用树状数组套哈希表的方法，或者其他一些哈希表套数据结构的方法。

总时间复杂度为期望，均摊  $O((n + m) \log^2 n)$ ，每次会在  $O(\log n)$  条极长重链上进行数据结构操作。

我们继续观察题目性质，发现这里如果  $X$  很小，则修改长为  $X$  的字符串最多影响  $O(X)$  条极长重链，也就是说每次实际上会在  $O(\min(\log n, X))$  条极长重链上进行数据结构操作，由于给定的字符串长度和与  $n, m$  同阶，于是这里修改的总时间复杂度为期望，均摊  $O((n + m) \log n)$ ，但是询问时间复杂度依然是  $O((n + m) \log^2 n)$ 。

考虑这个其实类似于单点修改询问链和的问题，我们之前的做法因为使用了树链剖分所以是  $O(\log^2 n)$  的，可以用经典的树上差分方法做到  $O(\log n)$ 。



我们对每个点维护其到根路径上所有被极长重链包含的长为  $X$  的字符串的出现次数，如图。

使用一棵线段树套哈希表或者哈希表套线段树维护树的 DFS 序。

每次路径修改由于只修改了  $O(X)$  个长为  $X$  的被极长重链包含的字符串，所以这里相当于进行了  $O(X)$  次区间插入一个数，询问经过树上差分后相当于询问单点上一个数的出现次数，这里使用标记永久化的线段树即可，然后对于所有跨重链（即经过轻边）的长为  $X$  的字符串我们还是使用上述的暴力 KMP 方法，复杂度分析同理。

于是这里的总时间复杂度为期望，均摊  $O((n + m) \log n)$ 。