

cactus

前20%

$m=n-1$ ,所以是一棵树, 直接树链剖分  
 $O(n)$ (能过)。

是一棵真的仙人掌

朴素的想法：每次加操作，从 $a_i$ 出发dfs，标记一下那些点可以走到 $b_i$ ，将标记的点进行加操作。

# 数据随机生成

仙人掌的期望高度是 $\log$ 的，直接dfs加操作，暴力加暴力查询即可。

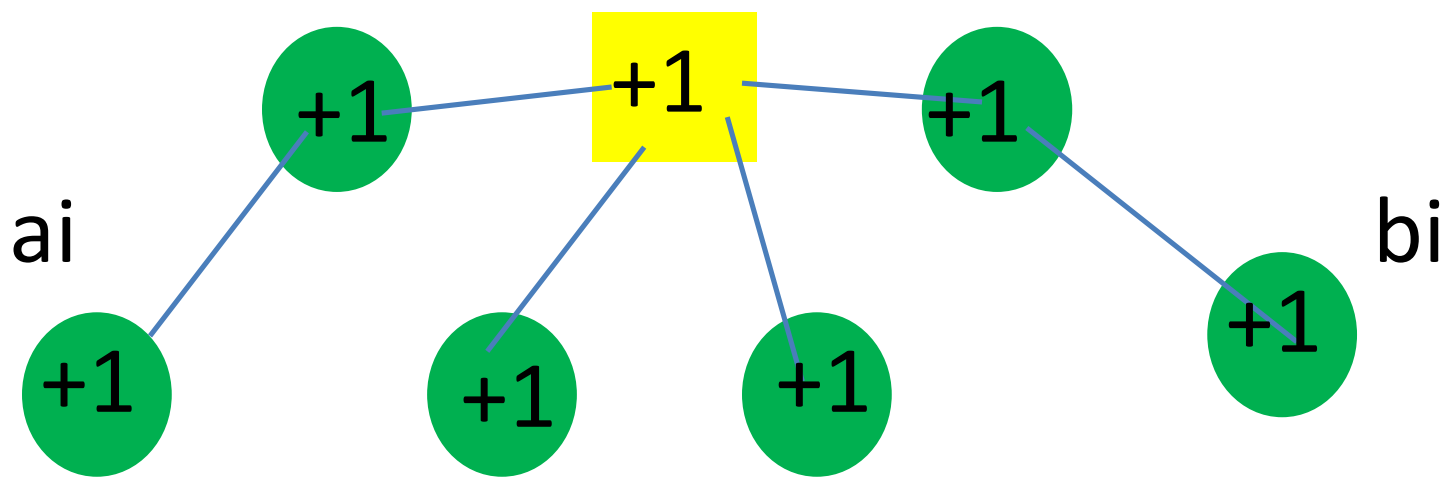
$O(\text{能过})$

# 比较高(ji)级(chu)的想法

你可能很早就学会圆方树了。

建出圆方树，题目相当于：

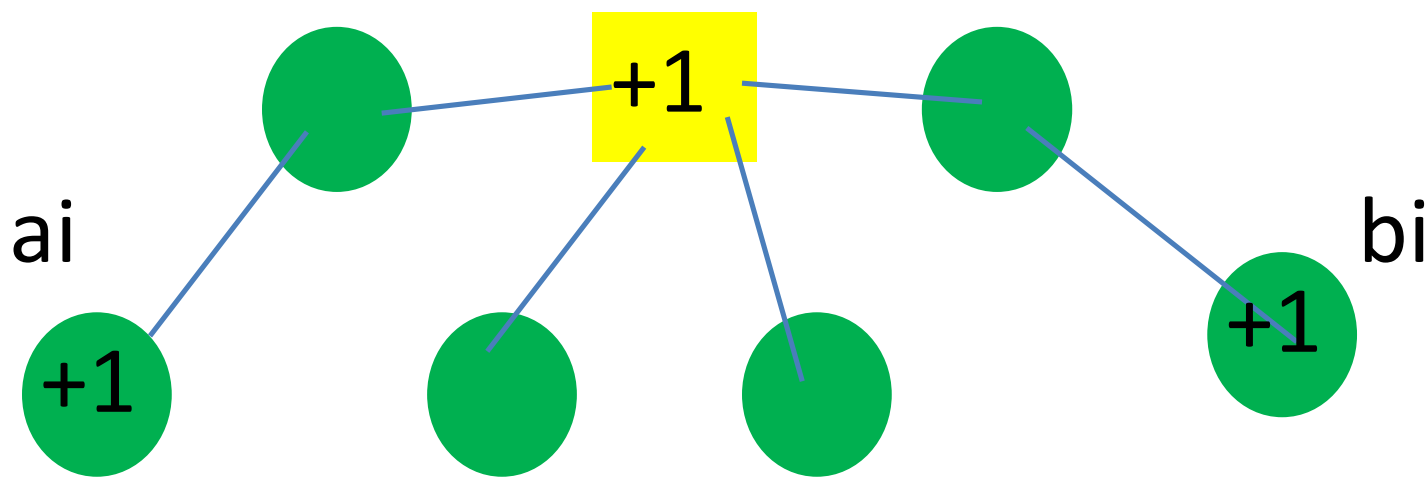
对一条圆方树的链进行加操作，同时对于每一个方点，与它相邻的所有圆点加。  
但相邻的圆点中也在这条链上的不加。



## 比较高(ji)级(chu)的想法

对于方点的贡献，我们就记录在方点上，相邻的圆点上不加，在查询时再单独查询对应圆点相邻方点的值的和。

所以现在做的只有一条链加和每个方点相邻的在链上的圆点减。



仙人掌只有一个环

只有一个方点，用上述做法直接完成。

效率 建圆方树+树链剖分+线段树

$O(n+n\log^2)$

# 比较高(ji)级(chu)的想法

对圆方树树链剖分

查询圆点相邻的方点的和：当方点是的重儿子或父亲，可以暴力查询。若是轻儿子，那么进行一次加操作，最多经过 $\log$ 个轻儿子，那么经过这些轻儿子方点时就可以把对应权值直接加到它的父亲圆点上。



# 比较高(ji)级(chu)的想法

每个方点相邻的在链上的圆点减：  
好像不好做。

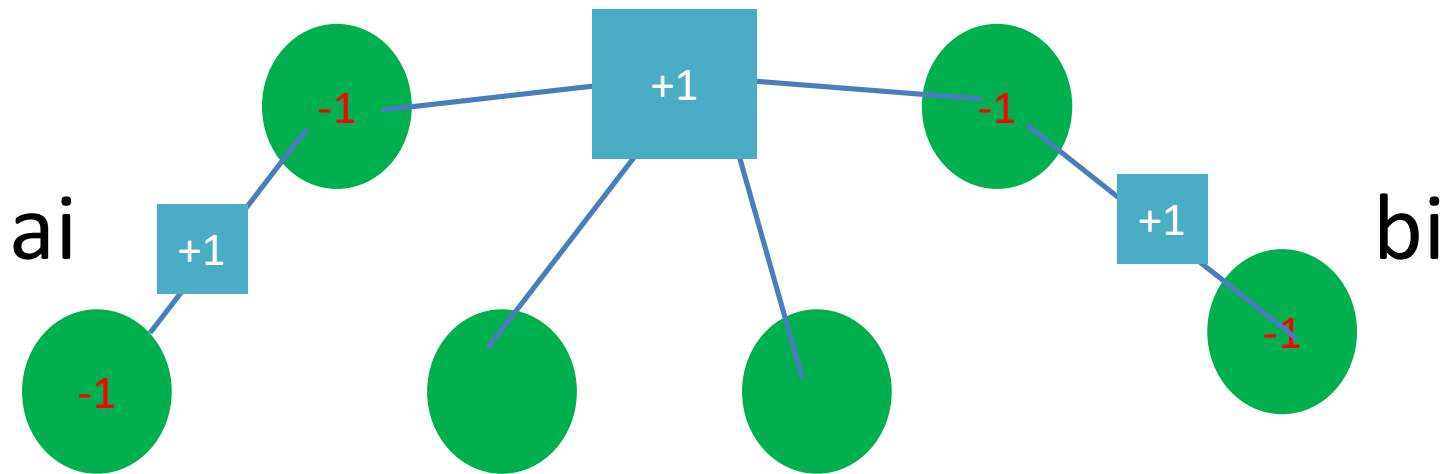
100%

圆方树在割边上也建一个方点  
每个方点上加，每个圆点上减，这样除了 $a_i$ 和 $b_i$ 没变，其它点的值都是对的。

只需 $a_i$ 和 $b_i$ 暴力加上去。

方点的圆点建两棵线段树即可。

$O(n \log^2)$



# 后记

但是这个题解是另外一个神仙想到，出题人出题时则写了一个极其sb的算法  
(std) 将代码长度扩展到5k.....