看到这道题，可以从两个角度入手考虑：

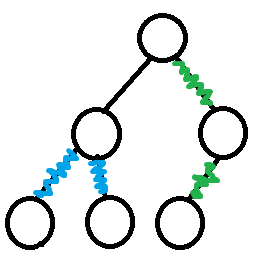
1）对于这棵树最终的形态，考虑哪些边可能是蓝边，并计算最大值

2）考虑这棵树建树的过程。在一开始只有根节点，后来不断加点过程中有哪些边是蓝边，并计算最大值

我一开始是从第一个角度考虑的

通过手动根据样例画图发现，蓝边一定是两条两条地连在一起（因为每次Insert操作都是两边一起变蓝，且这两边是相邻的）

考虑这两条蓝边（指同时出现的那两条蓝边）可能的连法：



连法一如两条蓝色边所示，连法二如两条绿色边所示

一个点若是两条蓝边的中心点，那两条蓝边要么是与自己两个孩子的连边，要么是与自己一个孩子的连边加上自己与父节点的连边

然后就可以树形dp了

f[0][i]表示i号点不做两条蓝边中心点：

f[0][i]=它每一个son可得到的最大值 的和

f[1][i]表示i号点做两条蓝边中心点，且是连法一：

f[1][i]=（选两个son，每个son不向i连边可得最大值 的和）+（这两个son到点i边的长度）+（其他son可得到的最大值 的和）

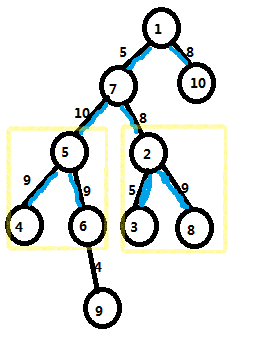
f[2][i]表示i号点做两条蓝边中心点，是连法二：

f[2][i]=（选一个son，该son不向i连边可得到的最大值）+（这个son到点i边的长度）+（点i到父节点边的长度）+（其他son可得到的最大值 的和）

大体思路就是这样，看起来很对吧……

然而是有问题的！我一开始就以为这样是对的，但发现WA，洛谷上也只得了20分……

一组反例：



按照上面的算法，最大值时蓝边应这么连

但注意两个黄框

必须先存在4、6才能出现5，必须先存在3、8才能出现2

对于7，必须2、5都存在且连上了一条边才可以出现

假设先以9-4-6-5的顺序把5及其子树都出现了，要想再出现2，需要让3、8出现

但3、8都是叶节点，唯一与2的连边还是蓝边，说明他们只能是一开始连了一条边，之后插入2

但这时5及其子树已插入完了，3、8没有与已存在的这棵树连在一起

2想出现的话只能3、8单成一棵树，与题目不符！

为了保证符合题意，我们考虑第二个角度

假设已确定根节点了，那么由题意，所有“两条蓝边”都是上面所说的连法二，不可能为连法一

这样同样可以树形dp，而且比上面的还简单

（转移方法与上面类似，不详细说了）

但是这是在确定一个根节点的情况下，而每个点都可能是根节点。

暴力每次树形dp的话O(n²)伤不起……

于是考虑换根。

从上到下换根。每次选当前“根节点”的一个未当过根的子节点作为新的根节点，把当前这个点作为它的子节点，O(1)维护各种信息

这样就很快了，只是细节需要多多多多多多注意一下