

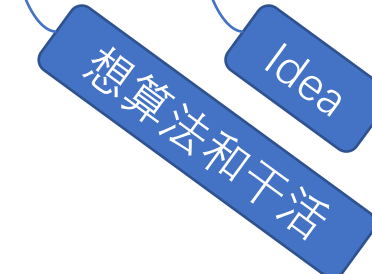
《情报中心》

试题讲评



清华大学

陈俊镔 吕欣 杨景钦 王逸松



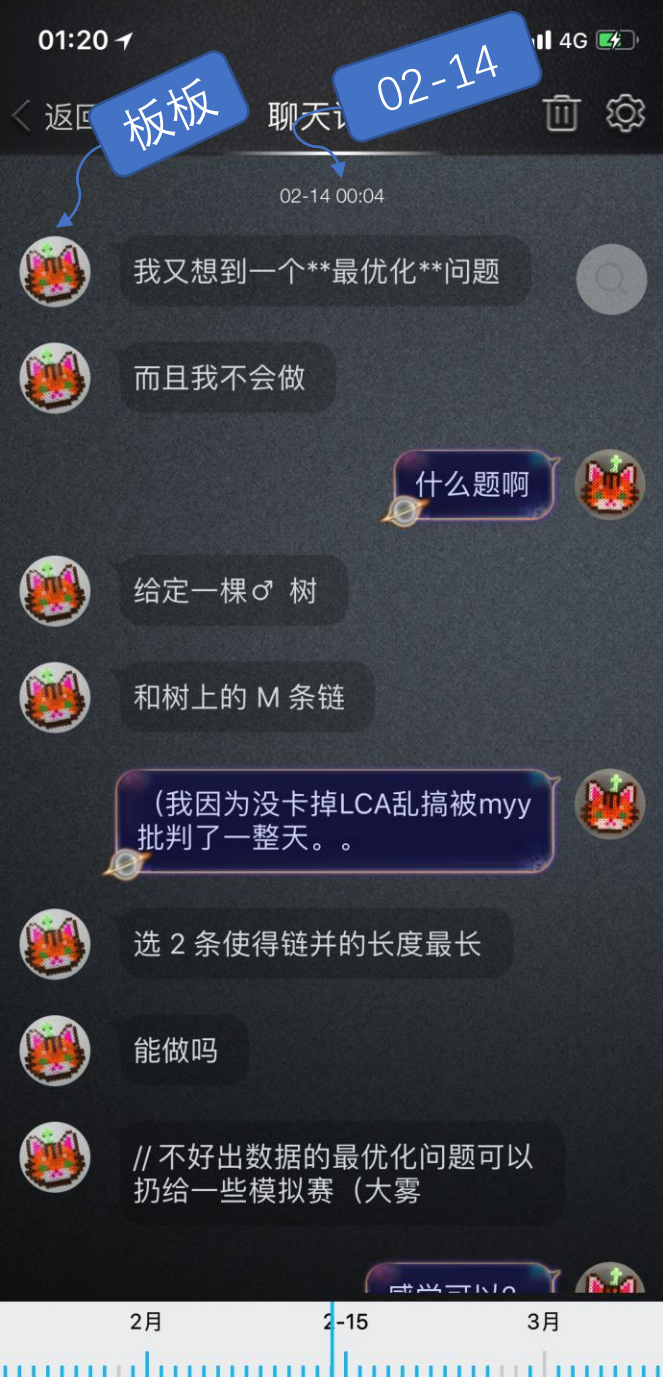
想算法和干活



Idea



乱搞和验题



预谋已久

- 早在 WC 刚结束时，这题已颇具雏形
- 但是当时，我们都并不会做，于是这题被搁置
- 4 月中旬的一天下午，TAC 忽然想出了这题的算法

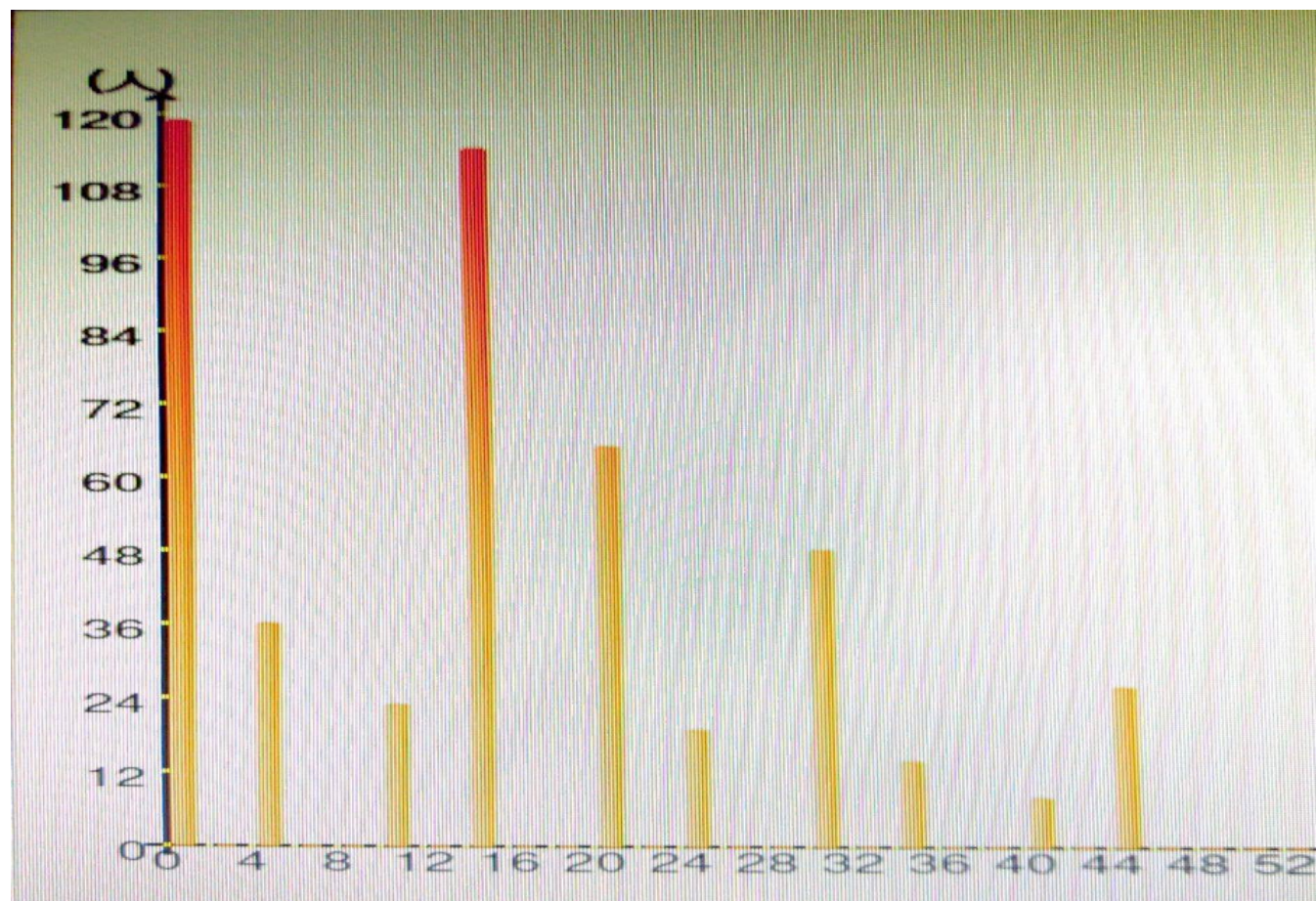


- 本想把这题投给 CTSC，但惨遭拒绝
- 于是——这题被留到了 NOI！

题意简述

- 给出一棵 n 个点的树，每条边有非负边权
 - 再给出 m 条树上的链，用端点表示，每条链都有价值
 - 要求选出两条链，满足这两条链**至少有一条边相交**
 - 最大化选出的**链的权值和 + 被至少一条链覆盖的边的边权和**
-
- 一个测试点可能有很多组数据， $n \leq 50,000$ 、 $m \leq 100,000$
 - 测试点内的 n 和 m 的和的数量级达到 10^6
 - 时间限制 8 秒，空间限制 512 MB

得分情况



梗与吐槽

- C 国和 D 国 → 猫 (Cat) 狗 (Dog) 大战
- reversed(BGG = 板给给哥哥) 和 reversed(CAT)
- 样例 4 的强度接近终测数据
- std 是一个 log 的

吐槽

吐槽

【样例 4】

见选手目录下的 *center/center4.in* 与 *center/center4.ans*。

这个样例，无疑是善良的出题人无私的馈赠。大量精心构造的 $n \leq 100, m \leq 200$ 的测试数据，涵盖了测试点中所有出现性质的组合。你可以利用这个测试点，对自己的程序进行全面的检查。足量的数据组数、不大的数据范围和多种多样的数据类型，能让程序中的错误无处遁形。出题人相信，这个美妙的样例，可以给拼搏于 AC 这道题的逐梦之路上的你，提供一个有力的援助。

大暴力

- 测试点 1、2、3 数据范围非常非常非常小
- 可以枚举两条链，到树上去统计覆盖的边
- $O(nm^2) = 15'$

聪明的暴力

- 测试点 1、2、3、4 数据范围有点小
- 考虑在枚举链的时候加优化
- 枚举第一条链，把每条边的权值 $w(e)$ 改成
$$w'(e) = [e \text{ 在第一条链上}]w(e)$$
并预处理求树上距离的数据结构
- 然后枚举第二条链求树上距离，注意判断是否相交
- $O(nm) = 20'$

真·暴力

- 仔细观察数据范围
- 我们相信出题人为了卡乱搞，不会卡暴力
- 于是只枚举有公共部分的链
- 注意使用树链剖分求 LCA 来减小常数
- 可以通过测试点 1,2,3,4,7,8,9,10,11,12,17
- 55'
- 加上一条链的情况可以 65'

测试点	$n \leq$	$m \leq$	$T \leq 50$	特殊性质
1	2	3	保证	无
2	10	30		
3	200	300		
4	10^3	2,000		$a_i = b_i - 1$
5	10^4	3×10^4		
6	5×10^4	10^5		
7	10^4	3×10^4		$c_i = 0$
8	5×10^4	10^5		
9				
10	10^4	n	不保证	S_1
11	5×10^4			
12				
13	10^4	3×10^4	保证	S_2
14				
15	5×10^4	10^5	不保证	
16				
17	10^4	3×10^4	保证	无
18	5×10^4	10^5		
19			不保证	
20				

基于迭代的乱搞

- 前一个暴力中，我们事实上是求出了“确定了一条链后的最大答案”
- 不难直接把这个过程改成乱搞
 - 随机取若干条链分别求答案取 max
 - 随机若干次，每次取一条链，找另一条链使得两条链答案最大，然后从另一条链出发执行这个操作，迭代多次并在每一步更新答案
 - 把链按照 {链代价, 链长度, 链经过边数} 的 $\{-1, 0, 1\}$ 的系数组合排序，对前面一些链求出这些链的答案
- 构造数据时，只需要保证每条链的答案接近，这样就能让链交部分的影响扩大到极致，从而卡掉所有这样的迭代乱搞
- $\leq 30'$

链的数据

- 测试点 4、5、6 中，树是一条链
- 从而每条链都是一个区间
- 枚举一条链后，直接用线段树维护即可
- $O(m \log n) = 15'$

边权为 0

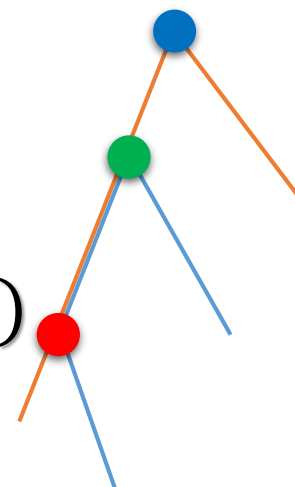
- 测试点 4、5、6 中边权为 0
- 只需要两条链的权值和最小，链只要相交即可
- 枚举一条边，求所有经过这条边的链的最大值和次大值，用它们的和更新答案
- 求经过一条边的链可以用树上差分 + 启发式合并/可并堆完成
- $O(m \log n) = 15'$

基于边的乱搞

- 这一个暴力同样也可以改成乱搞
- 对每条边求出经过它的按 {链代价, 链长度, 链经过边数} 的 $\{-1, 0, 1\}$ 的系数组合排序的前若干大, 两两之间求答案
- 然而, 前面的构造方法同样可以把这个乱搞卡的痛不欲生
- $\leq 35'$

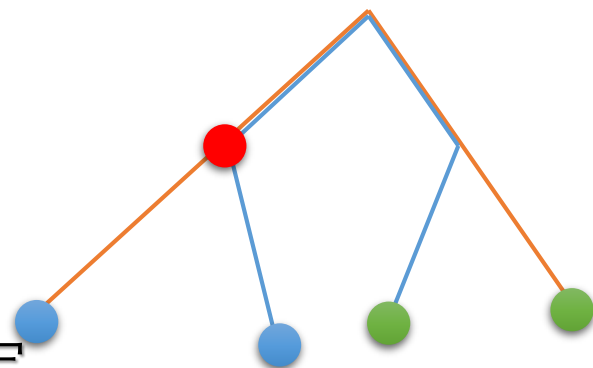
LCA 两两不同 (S_1)

- 对于 LCA 不同的两条链，两条链的交必然是直上直下的一段
- 因此可以把一条链拆成两条直上直下的链考虑
- 在树上枚举较下的交点 (红点)
 - 要求两条链的下端点必须在红点的不同儿子子树中
- 长度和 + 权值和 - 红点深度 + $\max(\text{绿点深度}, \text{蓝点深度})$
- 于是可以想到对每个点记 $f(i, j)$
 - 下端点在 i 子树内，上端点深度为 j 的最大的链长度 + 权值
- 用启发式合并或线段树合并维护这个数组， $O(m \log n) = 30'$

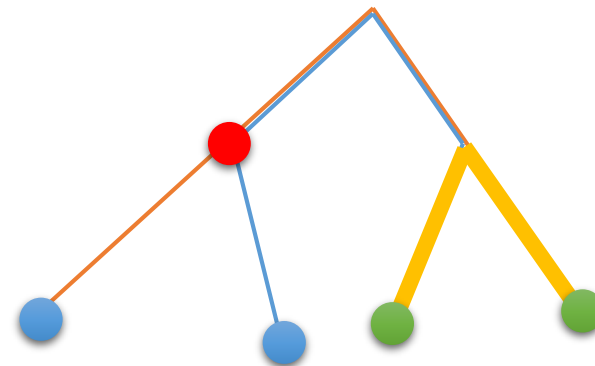


LCA 全部相同 (S_2)

- 对于这一部分数据，两条链的交可能不是直上直下的
- 关键性质：链并的两倍 = 两条链长 + 蓝点距离 + 绿点距离
- 那么可以考虑枚举红点，即蓝点的 LCA
 - 要求两个蓝点属于红点不同儿子的子树
- 这时候要考虑绿点的距离
 - 如果仍然用深度减去 LCA 深度考虑，无从下手
 - 事实上可以强行树链剖分套线段树维护，但这样太难写
 - 但我们是要最大化绿点的距离，可以直接从绿点的最远点对入手！



LCA 全部相同 (S_2)



- 枚举红点之后，我们的任务是
 - 选出两个属于红点不同儿子子树的蓝点 a, b
 - 使得它们对应绿点 p_a, p_b 的距离 + 两条链链长和权值和 + 两个蓝点深度和 $- 2 \times$ 红点深度最大
- 注意到红点深度是常数无需考虑
- 对于其他项，可以通过添加附加点 p'_a, p'_b 向 p_a, p_b 连接边权为自己所在链长 + 权值的边，转化为找最远点对 p'_a, p'_b
 - 要求支持集合合并
 - 在合并的时候，求出跨越两个集合的最远点对

LCA 全部相同 (S_2)

- 对于非负边权的树，两个点集的并的最远点对的一端，一定是原来两个点集中，某个点集的最远点对的一端
- 只需要对点集记录最远点对端点，即可支持合并
- 这个算法只需要在树上进行 DFS，以及求 m 次的 LCA
- $O(n + m) = 20'$

满分算法

- 两条相交的链，要么 LCA 相同，要么 LCA 不同
- 对于 **LCA 不同的所有情况**已经可以处理了
- 对于 **LCA 相同**的情况，考虑**同时处理**以每个点为两条链的共同 LCA 的情况
- 对每个点 p 记录 $f(p, j)$ 表示在点 p 处理 **LCA 为 j** 时保存的**蓝点**的集合（记录的是最远点信息）
- 对 f 数组进行启发式合并，复杂度 $O(n + m \log n)$
 - 也可以进行点分治，考虑所有过重心的链，转化为全过一个点的情况，复杂度 $O(n \log n + m \log n)$
- $O(n + m \log n) = 100'$

考察内容及难度评估

- 题目描述简洁易懂，解法自然
- 考察的知识点只有找 LCA 和启发式合并（和/或线段树），无偏门知识点，属于一道传统的数据结构题
- LCA 相同时给最远点对的规约，虽然算法比较自然，但有一定的思维难度，考察了选手是否有问题的深入分析能力和模型的转化能力，以及处理问题的一些思路
- 综合来看，本题代码难度中等，思维难度较高，可以较全面地考察选手的综合能力

感谢

- 感谢父母的养育之恩
- 感谢 CCF 给我这个命题和交流的机会
- 感谢板板、针针和松松松帮我验题题
- 感谢大家在这题上所付出的时间
- 感谢大家的聆听



谢谢大家！

祝大家学业有成