

T1 极乐迪斯科 (elysium)

作为瑞瓦肖57分局最优秀的警探，你被派到马丁内斯打击犯罪。马丁内斯的一些建筑物上安装了 m 个摄像头。你明白，普通人并不会因此受到多少保护，这些摄像头是保护*资本主义*的。作为一个信仰共产主义的战士，你决定滥用职权，拆掉一些摄像头。马丁内斯的街道是一棵 n 个点的**有根树**，每个点是一个商店，豪宅之类的重要建筑物，建筑物之间的边就是街道，一个摄像头可以监控**子树中**所有距离它 d_i 的建筑物。拆一个摄像头是不小的劳动，会减少 c_i 的快乐度。而一个毫无监视的重要建筑物会给予你 v_i 的快乐度。你想知道，你最大能增加多少快乐度。

solution

一个简单的想法是树形dp记录上一个未被覆盖的节点深度。复杂度 $O(n^2)$

另一个想法则是考虑最小割建图。

$$\begin{cases} \forall i, S \rightarrow \text{monitor}_i : c_i \\ \forall i, \text{node}_i \rightarrow T : v_i \\ \forall i, j, \text{dis}(x_i, j) \leq d_i \wedge \text{dep}_{x_i} \leq \text{dep}_j, \text{moniter}_i \rightarrow \text{node}_j : \infty \end{cases}$$

容易想到，若每个摄像头的流会优先供给最深的点，则不会发生退流。考虑维护深度-出流剩余量数组，每次就可以快速求出流了。因为数组是基于深度的，考虑长链剖分+启发式合并。用平衡树维护就可以快速查询。长链剖分每条链只会向上合并一次，复杂度 $O((m+n)\log n)$ 。在实现上可以直接使用map，每个新流可以暴力和map中元素比较。

T2 反讽 (irony)

有两个括号序列 $\{A_n\}, \{B_m\}, A_i, B_i \in \{(,)\}$ ，这两个括号序列**不一定合法**。现在，我们想要归并这两个序列，得到一个新的序列 $\{C_{n+m}\}$ （所谓归并，就是说每次将 $\{A\}$ 或 $\{B\}$ 中的第一个元素删去并加入新序列 $\{C\}$ 的末尾，直到 $\{A\}, \{B\}$ 均为空）。这样的序列依然**可能不合法**。我们再在其中插入一些括号，使得 $\{C\}$ 合法。你的任务就是找到一种归并方法，使得最后一步中插入的括号最少。

solution

建立一个映射 $(=1,)=-1$ 。首先显然加括号一定是开头加一些左括号，然后结尾补足右括号。如果 $\sum_{i=1}^n C_i < 0$ ，我们先在开头加 $-\sum_{i=1}^n C_i$ 个左括号。在 $\sum_{i=1}^n C_i \geq 0$ 时答案实际上是 $\sum_{i=1}^n C_i - 2 * \min(0, \sum_{i=1}^k C_i)$ ，即在最开头加上一些左括号使最小前缀和 ≥ 0 ，并在最后添加足量右括号使序列合法。

简单的暴力可以设 $dp[i][j]$ 表示从 A 取了 i 个数，从 B 取了 j 个数的最大最小前缀和。

更好的做法可以考虑经典的打怪兽问题。即给数对 (a_i, b_i) 表示杀死 i 号怪兽先扣 a_i 血，再加 b_i 血，攻击顺序必须满足一棵树的约束，攻击儿子必须在攻击父亲之后，求初始血量最小值使得可以杀死所有怪兽后不死（HDU6326）。设左括号为 $(1, 0)$ ，右括号为 $(0, 1)$ 即可套用。

这个问题的做法是先不考虑树的约束，可知攻击顺序可贪心，根据 $a < b \rightarrow \inf - a : b$ 的键值从大到小攻击（简要说明：显然当 $a < b$ 时按 a 从小到大打最优，而从后往前倒推攻击顺序则 $a \geq b$ 的数对变成 (b, a) ，于是同理可得此时应按 b 从大到小的顺序攻击）。考虑树的约束时，依然按此顺序依次考虑，可知这个怪一定会在它的父亲被杀后立刻被杀，于是合并两者即可。由于需要用堆维护，复杂度 $O(n \log n)$

考虑维护最小值的部分，可以发现 (a, b) 满足 $a, b \leq n + m$ ，用桶维护合并，可以优化至 $O(n)$ 。

T3 敏感词 (that)

你写了一篇文章，想要发布在知乎。不幸的是，你发现你文章中多了不少的星号。研究了一番，你发现你发现*那个词*被过滤掉了。过滤的方法是：标记出文章中出现一个词的所有位置，然后替换之。比如 `weininiubi` 对 `ini` 进行过滤后变成 `we*****ubi`，对 `in` 过滤则是 `we*****iubi`。热爱思考的你觉得这很有意思。你现在想知道，如果只将一个词设置为敏感词，能使你的文章中出现恰好 k 个星号的最短的词中字典序最小的是什么？

solution

考虑对单词 t 求星号数。拿出所有出现位置，根据位置间距会形成一个分段函数。求值即可。

那么我们建出 sam，在 parent 树上维护信息。我们需要：出现位置集合 \rightarrow 间距集合 \rightarrow 函数图像。

使用启发式合并或线段树合并就可以在树上逐点维护这些信息，即可得知每个点的最短合法串。接下来需要找到最小的。那么我们倒建 sam 维护前缀，在 parent 树上按字典序寻找合法点即可。