2024“钉耙编程”
中国大学生算法设计超级联赛(7)题解
解题报告

2024.8.9

目录

1 纠缠点对

若两条路径 (u,v) 和 (x,y) 存在交点，则必然有某条路径的 LCA 在另一条路径上。

不妨枚举 T_1 上 LCA 较浅的路径，设为 (u,v)，计算所有 LCA 被它覆盖的点对有多少条在 T_2 上和它相交。这需要处理两种情况，要么其在 T_2 上的 LCA 被 T_2 上的 (u,v) 覆盖，要么其在 T_2 上覆盖 (u,v) 的 LCA。

考虑解决第一种情况，若对所有 (u,v) 在 T_1 上覆盖其 LCA 的路径，在 T_2 上进行 LCA 处的单点 $+1$，则在 T_2 上进行 (u,v) 的路径查询就可以了。这可以用树上差分转换为对 LCA 处的子树 dfs 序上的区间 $+1$，之后进行 4 次单点查询，进而使用线段树进行维护。

若在 T_1 的每个点 u 处维护，从 u 到根的所有点，LCA 位于其处的路径，在 T_2 上对应的区间加，形成的可持久化线段树结构，则 T_1 上的路径所覆盖的所有点，形成的线段树结构，亦可以使用树上差分而表示为 4 个线段树的线性组合。

对于第二种情况的处理方法是十分对称的，需要注意两种情况中可能计算重复的部分需要进行一些特殊处理。总的时间复杂度为 $O(n\log n)$。

2 生产机器

可考虑如何检定一序列是否合法，可以从前往后地进行匹配，每次考虑一个充能球，如果当前的这一小时的对应颜色的生产数量足够，则将其减 1。否则，移动到下一小时继续匹配过程。如果在 n 个小时内可以匹配完输入的序列，则序列合法。

由于检定的过程和合法的序列是一一对应的关系，可以对这一检定的过程进行动态规划。$dp(i,j)$ 表示考虑到第 i 小时，这一小时所生产的第一个充能球的颜色必须为 j，对应多少种合法的方案。枚举超出这一小时的计划的充能球的颜色，该颜色的充能球对应的生产数量必须全部用完，且下一小时必须以该颜色开始，另一种则随意，即可进行转移。

假如当前生产计划中，去除掉被钦定为开头的颜色中的一个之后，两种充能球的生产个数上限分别为 x,y，且确定超出这一小时的计划的充能球生产个数上限为 x，则方案数为

$$\sum_{i=0}^{y}\binom{i+x}{i}=\binom{x+y+1}{x}$$

。

而若决定匹配过程在当天结束，则方案数为

$$\sum_{i=0}^{x}\sum_{j=0}^{y}\binom{i+j}{i}=\frac{(x+2)\binom{x+y+2}{x+2}-y-1}{y+1}$$

若记值域为 w，时间复杂度为 $O(n+w)$。

3 自动人偶

考虑建立自毁指令的 KMP 自动机。若记 T^∞ 为无限个 T 拼接而成的序列，可以注意到 T^∞ 和 KMP 自动机上从某点 u 出发而回到 u 的环是一一对应的。

这是因为，若 T^∞ 匹配到的点为 u，则其往后拼接一个 T 之后一定回到 u，说明其对应这一个环。而若从某点 u 出发，往后拼接一个 T 之后回到 u，则说明 T^∞ 一定匹配到 u，因为任何串往后拼接无数个 T 走到的点一定是同一个点。

再考虑如何计算剩余几个机器人，首先如果不考虑峭壁，若该指令序列的最大前缀 $\geq n-x+1$，则第 x 个机器人一定掉下悬崖而损坏。而如果有机器人曾经触碰过峭壁且损坏，则说明指令序列中存在大小为 n 的子串，此时所有机器人一定损坏。

综上，可以考虑在 KMP 自动机上 dp，如 $dp(i,j,k,l,m,n)$ 表示从第 i 个点出发，走了 j 步，走到点 k，指令序列的最大前缀和，最大后缀和，总和分别为 l,m,n 的方案数。复杂度为 $O(|S|^2n^3m)$。

4 战争游戏

首先，如果树的直径小于等于 $r1\times 2+1$ ，那么进攻方只需选择树的直径的中点，便可一回合就把防守方炸死。

否则，若 $r1\times 2\geq r2$ ，那么进攻方总可以用下面的方法把防守方逼死：

首先，进攻方选择一个与防守方的一开始所在位置距离为 $r1$ 的点（可以证明这样的点总是存在）作为轰炸中心，之后每回合，将轰炸中心往防守方当前所在位置移动 1 步，直到防守方移动至叶子节点，那么这一回合

进攻方就能把防守方炸死。由于防守方怎么移动都不能经过轰炸中心所在的点（不然一定会被炸死），所以这种策略一定可以在一定回合内把防守方炸死。

若 $r1 \times 2 < r2$ ，那么无论防守方在什么地方，进攻方选择什么位置作为轰炸中心，树上总会存在防守方可以到达的没有被轰炸的位置，此时防守方可以获胜。

5 Hold'em Shark

常见的一种写好大模拟的方式是尽量将题目的要求解耦合，从而避免调试时多个因素互相影响且难以独立测试，导致判断问题出在何处十分困难。对于本题，我们可以对题目要求作如下的拆分：

- 一张牌（点数大小，花色）
- 一手牌（牌型大小，点数大小）
- 不进行交换操作时指定 Capoo 的胜率
- 交换的两张牌分别来自公共牌和一只 Capoo 的私有牌时指定 Capoo 的胜率
- 交换的两张牌均来自 Capoo 的手牌时指定 Capoo 的胜率

对于下面三个问题，我们又可以提取出一个前置的问题：有 7 张牌（5 张公共牌和 2 张私有牌），从中选出一手牌（5张）可以得到的最大手牌。解决了这个问题，我们就可以枚举公共牌和 Capoo 的私有牌的全部可能情形，进而判断每一种可能的情形下获胜的 Capoo。由于可能出现的情形总数是固定的，因此可获胜的情形数目最多的 Capoo 即为此时胜率最高的 Capoo。

在完成这样的分解后，其中的每一个部分都可以比较容易的调试，从而有效提高解题效率。

6 怯战蜥蜴 II

首先先用分治 NTT 分别算出在坎格鲁斯普雷评估的实力小于等于 c 的

爪牙中，有 i 个是真正实力小于 c 的的概率 a_i 和在坎格鲁斯普雷评估的实力小于等于 c 的爪牙中，有 j 个是真正实力大于 c 的的概率 b_j 。

设真实实力小于 c 的爪牙一共有 m_1 个，枚举坎格鲁斯普雷胜利的场数 p ，答案为下面这个式子：

$$ans = \sum_{p=1}^{m}\sum_{i=p}^{n}\sum_{j=p}^{m_1} a_j \times b_{i-j} \times \frac{j^{\underline{p}}}{i^{\underline{p}}}$$

其中， $i^{\underline{p}} = i(i-1)(i-2)...(i-p+1)$ 。

将这个式子化简一下：

$$ans = \sum_{p=1}^{m}\sum_{i=p}^{n}(\frac{1}{i^{\underline{p}}}\sum_{j=p}^{m_1}((a_j \times j^{\underline{p}}) \times b_{i-j}))$$

容易发现括号里的式子是卷积形式，做 m 次 NTT 即可。

7 创作乐曲

首先，题目可以转化为询问子乐曲内最多可以保留多少个音符，使这些保留的音符构成的乐曲是“美妙动听的”。

考虑这样一个性质：对于一个音高为 a_i 的音符，在最优解中，接在其后面的音符至多有两种可能，分别为离这个音符最近的音高在 $[a_i, a_i+k]$ 内的音符和离这个音符最近的音高在 $[a_i-k, a_i]$ 内的音符。

这个性质的证明如下：显然，若这个音高为 a_i 的音符后面接了其它音符，那么总可以把上面两个音符中的至少其中一个插入到这两个音符中间，使解更优。

对于每个音符，用线段树找到上述所说的两种音符，然后便可以 $O(n)$ DP 回答每个询问了，时间复杂度 $O(nq+n\log n)$ 。

8 循环图

首先，设 $calc(i)$ 为走到 $1 \sim i$ 号节点的方案数之和，那么所求答案即为 $calc(R)-calc(L-1)$ 。

再设 f_i 为走到编号为 i 的节点一共有多少种方法，先 DP 算出这个数组下标为 $1 \sim n$ 的所有元素的值。

考虑矩阵快速幂，目标矩阵显然为

$$\begin{bmatrix} f_{i-n+1} \dots f_{i-2}\ f_{i-1}\ f_i \end{bmatrix}$$

DP 算出转移矩阵显然是容易的，这里不再赘述。

现在的问题是，我们要求的目标矩阵有很多，如果暴力求暴力相加的话显然会超时，下面给出解决办法：

容易发现，我们要求的目标矩阵构成了一个矩阵的等比数列，可以采用分治法计算矩阵等比数列的和，时间复杂度为 $O(n^3 \log R)$ 。

9　关于 agKc 实在不喜欢自动化于是啥都自己合成这件事

根据配方来建立一颗由产物指向原料的树，边权为对应的 x。

记 $cnt[i]$ 为第 i 种物品需要的数量，则对于一个配方而言，$cnt[a_i] = x_i \times cnt[b]$。记 $tot[i]$ 为第 i 种物品需要花费的总时间，则对于一个配方而言，$tot[b] = \sum_{i=1}^{k} tot[i]$

注意在上述过程中，将所有超过 10^9 上限的物品标记，不再进行后续计算。

最后根据目标产物子节点的情况，去除一个最大值，即可得到结果。

10　故障机器人想活下去

考虑贪心，每个烟雾弹都是对故障机器人能活着遇到的敌人中战损最大的使用，假设故障机器人在第 i 场战斗结束恰好死亡，那当前烟雾弹能使用的对象就是第 1 到 i 个敌人，对其中战损最大的使用烟雾弹，需要注意的是烟雾弹对每个敌人只能使用一次，每次使用烟雾弹后需要将该敌人移除。

按序遍历敌人，记录当前的总战损，用大根堆记录遭遇敌人的战损，每次需要使用烟雾弹时对堆顶的敌人使用，将其从堆中弹出并在总战损中减去，直到烟雾弹用完故障机器人死亡或故障机器人活到了第 n 场战斗结束时找到答案。时间复杂度 $O(n \log n)$。

本题不卡两个 log 的做法，使用二分等做法也能通过本题。

11 蛋糕上的草莓是蛋糕的灵魂

要使草莓的大小尽量大，草莓切的刀数就要尽量少。

为了保证每块蛋糕上都有草莓且数量一样，必须保证最终的草莓数是蛋糕数的倍数。不切蛋糕的情况下，蛋糕数的因数较少，草莓数可以切尽量少的刀凑成蛋糕数的倍数。

设草莓的初始个数为 x，切 n 刀，则草莓数为：$2nx$，蛋糕数为 y ，最终的草莓数是蛋糕数的 c (c 为正整数) 倍。

有式子：$2nx = cy$ ，得出 $n = \frac{cy}{2x}$ ，要保证 n 尽量小且 n 为正整数，则 c 取 $\frac{y}{2x}$ 的最简形式的分母 $\frac{2x}{\gcd(y,2x)}$ 。

当然，还有特殊情况，当初始草莓数已经为蛋糕数的倍数时，草莓不需要切。

12 华丽牧场

首先想到或观察样例可以注意到, 一个能够无限出伤时的状态是: 将几乎所有手牌抽到手中(抽弃牌堆只剩一张抽牌卡(卡面数字 > 0)), 再交替循环打出任意两张抽牌卡. 具体来说, 将这两张牌分别记作 A B , 打出 A 会抽出 B 并弃掉 A 本身, 再打出 B 就会抽回 A 弃掉 B , 从而反复循环.

我们发现其实任何无限出伤的状态都可以转化为上述状态. 因此题目等价于: 寻找最早的, 能够将所有牌抽上来的回合.

首先考虑怎么全抽上来. 由于抽牌堆始终按顺序排列, 可以将抽牌序列可以看成一个环, 从环上某点开始抽牌, 只要能沿环抽一圈, 就能将所有牌都抽到手上.

我们来描述抽牌的过程. 注意到打出一张牌会抽 a_i 张牌, 同时会弃 1 张牌, 因此一张牌会使手牌数 $+a_i - 1$. 因此我们记 $\operatorname{prefix}(p) = \sum_{i=1}^{p} a_i - 1$, 则能在一回合内从 l 抽到 r, 等价于抽牌过程中手牌数始终> 0, 即

$$\min_{i \in [l,r]} \{\operatorname{prefix}(i)\} - \operatorname{prefix}(l-1) > -m$$

. 特别地, 由于 $\operatorname{prefix}(i) - \operatorname{prefix}(l-1) = -m$ 意味着此时手牌为空, 不可能继续抽牌, 所以一个合法的抽牌区间这样的 i 只能有一个, 且如果有, 只能是 r. 由上, 如果从某点 p 开始一个回合, 且 $[p, p+n]$ 是一个合法的抽牌区间, 则这个回合就是可以无抽满牌, 也就是可以限出伤的.

然后考虑怎么抽最快. 发现如果从某一位置 p 开始抽的回合能无限, 那么从 p 之前的位置开始抽, 只要能抽到 p 位置就也能无限, 因为此时你手上的牌不会比从 p 位置开始抽时少. 因此尽量快的到达(或超过)第一个能抽满牌的位置就能无限. 这意味着打出所有抽牌卡, 能抽就抽就是最快的.

接下来实现这个过程. 由抽牌过程的描述知道, 若一个回合从 l 开始, 这个回合最多能抽到第一个 $\text{prefix}(i) - \text{prefix}(l-1) = -m$ 的位置, 因此我们可以直接考察 $\text{prefix}(i)$, 他每下降 $-m$, 就必须多耗费一个回合来达到无限, 因此最终答案就是 $\max\{1, \lfloor\frac{-\min_{i\in[1,n]}\{\text{prefix}(i)\}}{m}\rfloor\}$. 最小值用线段树维护, 复杂度 $O(q\log n)$.