

ASDF 杂题选讲

ASDF Reunite

2024.11.22

Druk

P9417

Description:

P9417 [POI 2021/2022 R1] Druk

形式化题意：给一个全小写字母矩阵，你需要选择一个 $1 \times x$ 和 $x \times 1$ 的完全相同的模板，使得整个矩阵可以被这两种模板不重不漏地覆盖，不可旋转。求所有可行的模板长度。 $n, m \leq 1000$ 。

Hint1: 有用的模板数量不多。

Hint1: 有用的模板数量不多。

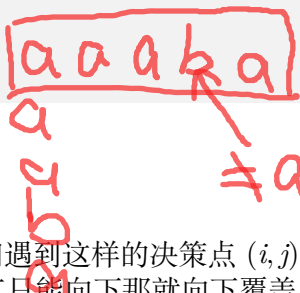
Hint2: 考虑找一种通用策略。

Solution:

显然模板长度 $L|nm$ ，但是可以再紧一点，考虑把 (i, j) 染成 $i + j \bmod L$ ，则任一种模板的摆放都会恰好取遍 $[0, L)$ ，那显然要求所有颜色出现次数相等，不难发现要求 $L|n$ 或 $L|m$ 。又因为模板一定在 $(1, 1)$ 开头，所以可能合法的模板只有 $2(d(n) + d(m))$ 种，在该数据范围下大约只有 60，考虑能否快速判定。

考虑直接从左上往右下填充，发现唯一的问题是可能在一些位置同时可以向下向右覆盖，此时我们无法决策。

这里我们给出一个结论：在同时可以向下向右覆盖时，我们直接向右覆盖，一定不会将有解漏判。



Solution:

考虑反证，如果我们遇到这样的决策点 (i, j) 时，选择向下覆盖，那继续向后考虑，如果一直只能向下那就向下覆盖，直到我们碰到一个可以向右覆盖的位置 $(i, j+p)$ ，此时因为模板首元素相同，因此 $(i, [j, j+p])$ 全相等，而在 i 行又有 $c_{[j, j+L-1]} = c_{[j+p, j+p+L-1]}$ ，所以整个这一块字符全相等，则模板字符全相等，那只有整个矩阵全相等才可行，判掉即可。如果一直不存在这样一个可以向右覆盖的位置，那只能一直向下覆盖，同样可以推出模板字符全相等。

因此无论怎样，该策略不会漏判，预处理哈希值模拟即可。复杂度 $O(dn^3)$ 或 $O(dn^2 \log n)$ ，因为一旦不合法就会跳掉，远远跑不满，因此都可以通过。

A Perfect Problem

CF1603E

Description:

CF1603E A Perfect Problem

形式化题意：有限制整数序列计数。一个序列是好的，当且仅当 $\min_{a_i} \times \max_{a_i} \geq \sum a_i$ 。一个序列是完美的，当且仅当其每个非空子序列都是好的。求， $a_i \in [1, n+1]$ ，长为 n 的完美序列数量。 $n \leq 200$ 。

A Perfect Problem

CF1603E

Hint1: 如何更好地刻画合法性。

A Perfect Problem

CF1603E

Hint1: 如何更好地刻画合法性。

Hint2: 如何 dp 计算方案数。

A Perfect Problem

CF1603E

Hint1: 如何更好地刻画合法性。

Hint2: 如何 dp 计算方案数。

Hint3: 继续分析性质，考虑优化一些上下界。

A Perfect Problem

CF1603E

Solution:

我觉得挺牛的题。我们先挖掘一些性质，不然这个子序列的限制是没法做的。

性质一：② 一个序列，排序后不改变完美性。因为子序列本质是子集，集合无序。

性质二：③ 若一个序列已经排序，它是完美的当且仅当它每个前缀都是完美的。考虑固定一个最大值，前面肯定是不停往前去，既可以让最小值最小，又可以增大元素和，显然会取满前缀。

Diagram illustrating the relationship between the sum of elements in a subsequence and the product of its minimum and maximum elements. The diagram shows two nested intervals: the outer one is $[1, r]$ and the inner one is $[l, r]$. To the right of the intervals, the inequalities are written:

$$a_1 \times a_r \geq \text{sum}[1, r]$$
$$\uparrow a_l \times a_r \geq \text{sum}[l, r] \downarrow$$

A Perfect Problem

CF1603E

Solution:

性质三: $\forall i, a_i \geq i$ 。 $\sum_{i=1}^k a_i \geq ka_1$, 而限制对应为

$a_1 \times a_k \geq \sum_{i=1}^k a_i \geq ka_1$, 所以有 $a_i \geq i$ 。

性质四: 若 $a_i = i$, 则 $\forall j \leq i, a_j = i$ 。 因为 $\sum_{j=1}^i a_j \leq a_1 a_i = ia_1$, 得

$\sum (a_1 - a_j) \geq 0$, 又因为 a_i 有序, 所以得证。

性质五: 若 $a_i \geq i+1$, 且 $[1, n]$ 是好的, 那么任意前缀是好的, 也就是完美的。代数变形可证。

那么现在就可以考虑 dp 了! 套路地, 根据值域进行 dp, 每次分配系数。

$$a_n \in [n+1, n+1]$$

$$a_n = n+1.$$

$$a_1, a_n = (n+1)a_1 \geq \sum_{j=1}^n a_j, a_{n-k+1} a_1 \geq \sum_{j=1}^k a_j, k \in [1, n]$$

$$a_1 \geq \sum_{j=1}^n (a_j - a_1) \geq \sum_{j=1}^k (a_j - a_1)$$

A Perfect Problem

CF1603E

首先是对 $\sum$ 的估计，我们有： $a_1 \times a_n \geq \sum a_i = na_1 + \sum(a_i - a_1)$ ，移项得： $\sum a_i - a_1 \leq a_1(a_n - n)$ 。

可以发现，如果我们只记录 $\sum a_i - a_1$ 的话，那 $\sum$ 是 $O(n)$ 级别的！~~这~~

Solution:

序列中的数

设 $f_{i,sum,k}$ 表示当前选了 i 个，总和为 sum ，将要选 k 的方案数

枚举一下选 k 的数量 cnt ，然后可以得到转移：

④ $i \geq k$ 时
 $i < k$ 时

$$f_{i,sum,k} \leftarrow \frac{f_{i+cnt,sum+k \times cnt,k-1}}{cnt!}$$

发现这个东西没有后效性，加个记忆化搜索就可以了。

$i=n$ 时, $f_n, ?, ? = n!$

$cnt!$

$\leftarrow cnt$ 个 k 的排列数 A_{cnt}^{cnt}

由于需要枚举 a_1 ，所以时间复杂度 $O(n^4 \lg n)$ 。

这个 $\lg$ 是枚举 cnt 的复杂度，复杂度瓶颈反而在 a_1 的枚举上了。

$$cnt \leq \frac{n}{k-a_1}$$

调和级数

A Perfect Problem

CF1603E

Solution:

性质 4: $a_1 \geq n - O(\sqrt{n})$.

大概就, 考虑确定了 a_1 如何让 $\sum a_i$ 最小, 很直接的想法是令前面的 a_i 全都等于 a_1 , 但是 a_1 不能超过 a_1 个, 因此直接这样构造是错的, 因此最优策略肯定是, 先放 a_1 个 a_1 , 然后再依次填上 $a_1 + 2, a_1 + 3, \dots, n + 1$, 解个方程可以得到 $a_1 \geq n - O(\sqrt{n})$.

不难发现, 真正能导致合法的 a_1 其实不多, 一些简单的数学分析或者打表可以发现, $a_1 \in [n - 2\sqrt{n}, n]$, 我们只需要枚举这些 a_1 即可。

复杂度 $O(n^3 \sqrt{n} \log n)$, 实际上还要带上 $\frac{1}{16}$ 的常数。

$$828 \times \frac{1}{16}$$

Tree Weights

CF1844G

Description:

CF1844G Tree Weights

形式化题意：给定一棵 n 个点无根树，对于 $i \in [1, n)$ 给出 $d_i = \text{dis}(i, i+1)$ ，还原一组边权或判断无解。 $n \leq 10^5$, $V \leq 10^{12}$ 。

Tree Weights

CF1844G

Hint1: 考虑无根换有根。

Tree Weights

CF1844G

Hint1: 考虑无根换有根。

Hint2: 考虑在模意义下递推。

Tree Weights

CF1844G

Solution:

$$\begin{aligned} \text{dep}_{i+1} &= 2\text{dep}_i + d_i - \text{dep}_{l_i} \\ &= \sum w \end{aligned}$$

先把无根换有根，不妨让 1 做为树根，我们引入 dep_i 辅助分析，记 $l_i = \text{LCA}(i, i+1)$ ，则有 $d_i = \text{dep}_i + \text{dep}_{i+1} - 2\text{dep}_{l_i}$ 。显然 $\text{dep}_1 = 0$ 。人类智慧地，考虑倍增递推，考虑顺序推出 $\text{dep}_i \bmod 2$ 的值，此时 $2\text{dep}_{l_i} \bmod 2 = 0$ ，可以直接递推。考虑一个结论：

$x \bmod 2^k = 2x \bmod 2^{k+1}$ ，因为模 2 的幂等价于二进制截尾。那么再考虑求出所有 $\text{dep}_i \bmod 4$ 的值，此时 $2\text{dep}_{l_i} \bmod 4 = 2(\text{dep}_{l_i} \bmod 2)$ 已知，可以直接递推。

那么就做完了，只需要对模数倍增递推到一个充分大值，即可确定 dep_i ，根据 dep_i 还原树边权判定有无解是容易的，复杂度 $O(n(\log n + \log V))$ 。

$$x_u \geq x_{\text{fa}_u} \quad \& \quad d \text{ check}$$

Description:

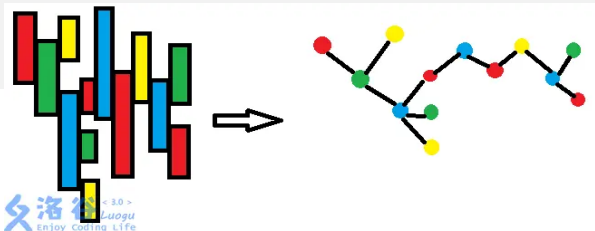
CF936E Iqea

形式化题意：无限网格图上给 n 个点，满足这些点形成一个连通块，且补集也是一个连通块。如果两个点在给定的 n 个点内且四连通，那可以走一步互达，现在有 q 次操作，或为在 n 个点中新增一个关键点（不是新增一个点，只是选一个当关键点）或为求一点 (x, y) 到其最近关键点距离。 $n, q, V \leq 3 \times 10^5$ 。

Hint1: 观察网格图的特殊性质。

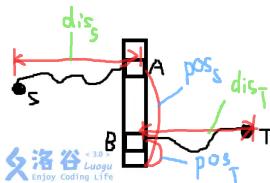
Hint1: 观察网格图的特殊性质。

Hint2: 标记关键点，查询最近距离，考虑点分。



Solution:

考虑这个网格图的特点，如果我们将列的极长连续段看成一个点，如果两个连续段有公共边则连边，那么这个新图一定会形成一棵树。操作近似看作树上标记关键点，求树上最近关键点，这个是点分树模板题，但是我们不能这么近似，因为距离计算非常麻烦。考虑直接建**点分树**，把整个点分中心里的点丢进去跑BFS，**点分树总点数 $O(n \log n)$** ，记一下每个点跑到点分中心的对应长条的最小距离 dis 以及位置 pos ，然后两个点的距离可以看成到点分中心的距离差 $|pos[S] - pos[T]|$ 再加上对应位置之间距离 $dis[S] + dis[T]$ 。可以前后缀两个树状数组维护。因为是距离 $\min$ ，因此同一个子树内的一定不会被算入，时间复杂度 $O(q \log^2 n)$ 。



Koxia and Sequence

CF1770F

Description:

CF1770F Koxia and Sequence

形式化题意：有限制序列权值计数。给定 n, x, y ，对于所有 $\sum_i a_i = x$ ， $\text{OR}_i a_i = y$ 的序列 $\{a_i\}$ ，求 $\bigoplus a_i$ 的异或和。

Koxia and Sequence

CF1770F

Hint1: 求异或和对本题有哪些提示。

Koxia and Sequence

CF1770F

Hint1: 求异或和对本题有哪些提示。

Hint2: 考虑拆位。

Koxia and Sequence

CF1770F

Hint1: 求异或和对本题有哪些提示。

Hint2: 考虑拆位。

Hint3: 容斥，卢卡斯定理，组合恒等式。

Koxia and Sequence

CF1770F

Solution:

首先要知道两个结论：

$2 \mid n$ ，答案为 0。

$2 \nmid n$ ，答案为所有序列的 a_1 的异或和。

结论一是因为，任意一个序列，将其反转后会和原来的异或抵消，这时候可能有些问题的就是回文，但是偶回文啊，异或和显然为 0。

结论二是因为每个位置上的数等价，假如我们单独求出某个位置上的数在所有可能的序列中的异或和，则这些数都是相等的，进行奇数次（ n ）异或后，就等于只考虑一个数了，不妨就考虑 a_1 。

子集反演

定义 $[x]$ 表示集合 $\{i | 1 \leq i \leq x, i \in \mathbb{Z}^*\}$ 。

设 $S \subset [n]$ 。有：

$$f(S) = \sum_{T \subset S} g(T) \iff g(S) = \sum_{T \subset S} (-1)^{|S|-|T|} f(T)$$

$$f(S) = \sum_{T \supset S} g(T) \iff g(S) = \sum_{T \supset S} (-1)^{|T|-|S|} f(T)$$

证明：

只证明第一个式子，第二个与第一个类似。

$$\begin{aligned} \sum_{T \subset S} (-1)^{|S|-|T|} f(T) &= \sum_{T \subset S} (-1)^{|S|-|T|} \sum_{Q \subset T} g(Q) \\ &= \sum_{Q \subset S} g(Q) \sum_{Q \subset T \subset S} (-1)^{|S|-|T|} \\ &= \sum_{Q \subset S} g(Q) \sum_{T \subset (S/Q)} (-1)^{|S/Q|-|T|} \end{aligned}$$

观察 $\sum_{T \subset (S/Q)} (-1)^{|S/Q|-|T|}$ 发现当 $S/Q = \emptyset$ 时为 1，否则为 0。两两配对可以证明。

所有除了 $S = Q$ ，其它的都是 0，所以原式就等于 $g(S)$ ，得证。

Koxia and Sequence

CF1770F

Solution:

$$f(y) = \sum_{z \subseteq y} \text{ans}(z)$$

已知条件

ans(y)

mod 2

然后考虑经典拆位，对于某一个 2^k ，考虑什么时候 a_1 包含 2^k ，注意到我们只关心方案数的奇偶性。

考虑对按位或做子集反演，方案数为 $\sum_{z \subseteq y} (-1)^{y-z} f(z)$ ，而 $1 \equiv -1 \pmod{2}$ ，因此也可以写成

$\bigoplus_{z \subseteq y} f(z)$ 。# $f(z)$: OR_i(a_i) $\subseteq$ z 的 {a_n} 的方案数

对于每个 $f(z)$ ，其方案为：

$$f(z) = \sum_{a_1 + a_2 + \dots + a_n = x - 2^k} [a_1 \subseteq s - 2^k] [a_2 \subseteq s] \cdots [a_n \subseteq s] \pmod{2}$$

(保证 $2^k \subseteq s$)

枚举 {a_n}

Koxia and Sequence

CF1770F

注意到根据 Lucas 定理, $[a \subseteq z] = \binom{z}{a} \bmod 2$, 因此化简得到:

$$\begin{aligned} f(z) &= \sum_{a_1 + \dots + a_n = x} \binom{z - 2^k}{a_1 - 2^k} \times \binom{z}{a_2} \times \dots \times \binom{z}{a_n} \bmod 2 \\ &= \binom{nz - 2^k}{x - 2^k} \bmod 2 \\ &= [x - 2^k \subseteq nz - 2^k] \end{aligned}$$


然后对于每一位暴力枚举 z 即可。

时间复杂度 $\mathcal{O}(y \log y)$ 。

范德蒙德卷积：

$$\sum_{i=0}^n \binom{a}{i} \binom{b}{n-i} = \binom{a+b}{n}$$

$$\sum_{\sum_i a_i = x} \binom{z-2^k}{a_1-2^k} \times \prod_{i=2}^n \binom{z}{a_i}$$

$$= \sum_{a_1+a_2=x_0} \binom{z-2^k}{a_1-2^k} \times \binom{z}{a_2} \sum_{\sum_{i=3}^n a_i = x-x_0} \prod_{i=3}^n \binom{z}{a_i}$$

$$= \binom{2z-2^k}{a_1+a_2-2^k} \sum_{\sum_{i=3}^n a_i = x-x_0} \prod_{i=3}^n \binom{z}{a_i}$$

Turtle

CF1239E

Description:

CF1239E Turtle

形式化题意： $2 \times n$ 的棋盘的左上角走到右下角，只能往下或往右，需要给出一种方案来重排 $2n$ 个数字，使得在所有可能的走过的路径中，路径上的数之和的最大值最小。 $n \leq 25, V \leq 5 \times 10^4$ 。

Hint1: 对于每一行，考虑另一行固定时，怎样会更优。

Hint1: 对于每一行，考虑另一行固定时，怎样会更优。

Hint2: 当两行固定时，怎么走会取到最大值。

Solution:

直接做是没法做的，那个 $n \leq 25$ 总是把思路带到各种神秘随机，状压那套东西中去，但这玩意其实是个陷阱。

性质一：第一行一定单调不降，第二行一定单调不升。

考虑调整法证明第一行，第二行同理。对于一个确定的局面，也就是第二行固定时，假设第一行存在逆序对 (i, j) ，那在 i 之前下行或在 j 之后下行无论是否交换 b_i, b_j 是没有影响的，但是在 $i \leq x \leq j$ 时就又影响了，因为这样的话交换比不交换至少会优 $a_i - a_j$ 。因此交换更优。所以调整证明了这个结论。

$\text{ans} = \max(\text{sum_line_1} + a_{\{2,m\}}, a_{\{1,1\}} + \text{sum_line_2})$

Solution:

性质二：一定会在第 1 列或者 m 列才下行。

如果在 $i (i \neq 1/n)$ 下去更优，考虑 $i-1$ ，和 $i+1$ ，则说明 $a_{1,i} > a_{2,i-1}$ ， $a_{2,i} > a_{1,i+1}$ ，由性质一可得矛盾。

所以，我们只需要把最小的和次小的分给 $(1,1)$ 和 $(2,m)$ ，剩下的做一个背包，希望划分成差值最小的大小都为 $n-1$ 的集合即可。复杂度 $O(n^3 V)$ ，可以 bitset 优化至 $O(\frac{n^3}{w} V)$ 。

于是将剩下 $2n-2$ 个数分成两组，使元素和较大那组的和尽量小。这是经典的背包问题， $vis_{i,j,k}$ 表示考虑到第 i 个数，第一行被填了 j 个，第一行的和为 k 的情况是否存在。转移方程为 $vis_{i,j,k} = vis_{i-1,j,k} \text{ OR } vis_{i-1,j-1,k-a_i}$ 。