

下饭模拟赛 题解

Skydogli & Pengzhike & CraZYali

January 21, 2021

Contents

1	联邦解体	2
2	遗失的河图	3
2.1	简要题意	3
2.2	算法零	3
2.3	算法一	3
2.4	算法二	3
2.5	算法三	4
2.6	算法四	4
3	拉托机	5
3.1	简要题意	5
3.2	算法一	5
3.3	算法二	5
3.4	算法三	5
3.5	算法四	6

1 联邦解体

114514

2 遗失的河图

2.1 简要题意

给出一 $n \times m$ 的正整数矩阵每行每列的最大值。

对合法矩阵计数。 $n \leq 10^5$

2.2 算法零

可以证明，存在解的充要条件是 $A = B$ 。

那么，直接输出 0，期望得分 5。

2.3 算法一

引入容斥原理：

$$\sum_x \sum_y (-1)^{n+m-x-y} f_{x,y}$$

其中， $f_{x,y}$ 为至多有 x 行， y 列能取到最大值的方案数。

在 $n, m \leq 10$ 的情况下，可以 $\mathcal{O}(2^{n+m})$ 枚举每行每列是否取到最大值，然后求出每个格子的上界，相乘即可。

进一步的，在 $n \leq 10$ 的情况下，我们只枚举行的限制情况，发现每一列的贡献是独立的，相乘即可。

复杂度 $\mathcal{O}(2^n nm)$ ，结合算法零，期望得分 20。

2.4 算法二

对于 $\{a_i\}, \{b_i\}$ 中的数全部相等的情况，考虑优化式子：

$$\sum_x \sum_y (-1)^{n+m-x-y} \binom{n}{x} \binom{m}{y} (q-1)^{nm-xm-yn+xy} q^{xm+yn-xy}$$

直接枚举就可以做到 $\mathcal{O}(nm)$ 。

进一步的,

$$(q-1)^{nm} \sum_x (-1)^{n+m-x} \binom{n}{x} (q-1)^{-xm} q^{xm} \sum_y \binom{m}{y} (-1)^y (q-1)^{y(-n+x)} q^{y(n-x)}$$

容易发现是二项式定理,

$$(q-1)^{nm} \sum_x \binom{n}{x} (-1)^{n+m-x} (q-1)^{-xm} q^{xm} \left(\left(\frac{q}{q-1} \right)^{n-x} - 1 \right)^m$$

可以做到 $\mathcal{O}(n \log n)$ 。结合前面两个算法, 期望得分 40。

2.5 算法三

容易发现, 交换 a_i 和 a_j 不会使答案改变。这启示我们直接对 a_i, b_i 排序。

那么, 对于 $A, B \leq 2$ 的情况, 那些行或列最大值为 1 的行和列已经确定了, 套用算法二, 期望得分 60。

2.6 算法四

$A, B \leq 2$ 的部分分, 用以启发正解: 对于被全局最小值限制的行和列, 一定不会满足其他行和列上界的限制。

所以, 容易想到, 我们可以每次剥去全局最小值的行和列, 剩下的问题是子问题, 分步相乘即可。

那么, 相当于处理一个 L 型, 每行每列都要取到上界的方案数。略微更改算法二的式子, 令 c, d 分别为全局最小值的行数和列数:

$$\sum_x^c \sum_y^d (-1)^{c+d-x-y} \binom{c}{x} \binom{d}{y} q^{xm+yn-xy} (q-1)^{cm+dn-cd-xm-yn+xy}$$

同样的, 用二项式定理, 复杂度 $\mathcal{O}(n \log n)$, 期望得分 100。

3 拉托机

3.1 简要题意

在模 m 意义下有一累加器 T ，每次以 $p_i = \frac{a_i}{\sum a_j}$ 的概率把 T 加上正整数 $i \in [1, n]$ ，求 T 变成 k 的期望步数。 $n \leq 500, k < m \leq 10^{18}, n < m$ 。

3.2 算法一

$n, m \leq 3$ ，想怎么做怎么做。

skydogli的本意是手解方程，不知道有没有人这个也挂子。

3.3 算法二

直接硬上一个高消解方程即可。

3.4 算法三

n 这么小， m 这么大，肯定有蹊跷。

不妨考虑一下我们到底高消了个啥。

设 T 变成 i 的期望步数为 E_i ($E_0 = 0$)，假设我们已经有了 $E_{0 \dots n-1}$ 。

那么对于 $j \geq n$ ，我们有：

$$E_j = 1 + \sum_{i=1}^n p_i \times E_{j-i}$$

不难发现， E_j 是 $E_{j-1, j-2 \dots j-n}$ 和 1 的一个常系数线性组合。

最终 E_j 用 $E_{0 \dots n-1}$ 和 1 组合出来的系数可以矩阵快速幂求出。

再回过头来看看 $0 \leq j < n$ 时 E_j 的情况，不难发现他们满足：

$$E_0 = 0, E_j = 1 + \sum_{i=1}^n p_i \times E_{(m+j-i) \pmod m}$$

所以此时有用的值就只有 $E_{m+1-n}, E_{m+2-n} \dots E_{m-1}$ ，而他们用 $E_{0 \dots n-1}$ 和 1 组合的系数可以一次矩阵快速幂求出来！

此时我们就快速(?)得到了关于 $E_{0 \dots n-1}$ 的方程了，高消解之即可。

至于要求的 E_k 也可以一次快速幂得到系数，再用 $E_{0 \dots n-1}$ 算出来。

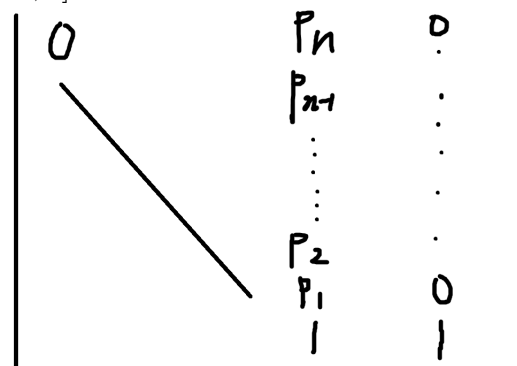
时间复杂度： $\mathcal{O}(n^3 \log m)$

3.5 算法四

老生常谈的矩阵快速幂子。

瓶颈在于 $\mathcal{O}(n^3 \log m)$ 的快速幂太慢了，我们要 $\mathcal{O}(n^2 \log m)$ 的。

出题人的初始矩阵（行向量）： $[E_0, E_1, \dots, E_{n-1}, 1]$ 。



出题人的转移矩阵（那长长的一条斜线是1）：

不难发现，这个转移矩阵的特征多项式是（这里正负号无所谓）：

$$(-1)^{n+1+n}(-x)^{n-1} + \sum_{i=1}^n (-1)^{i+n} (p_{n-i+1} - [i=n]x)(-x)^{i-1}(1-x)$$

当然你 $\mathcal{O}(n^3)$ 上海森堡上海堡垒求出这个也行。

然后直接 $\mathcal{O}(n^2 \log m)$ 多项式取模就可以了。要上NTT也可以。

还有一个小细节：预处理矩阵的 k 次方要注意之乘有值的地方，不然复杂度可能会变成 $\mathcal{O}(n^4)$ 。

总时间复杂度： $\mathcal{O}(n^3 + n^2 \log m)$ 。