组合数学专题分享

Little09

2025年8月3日

前言

这次分享的内容为一些 OI 里的数学相关内容，总体难度不大。
题目中会包含一些简单题、经典题，想必在座的同学有很多见过，不过拿出来讲一是考虑一些没见过对应套路的同学，二是见过的同学也可以加深一下影响。
希望大家都能有所收获!

组合数

记在 a 个物品中选 b 个的方案数为 $\binom{a}{b}$，即常说的“组合数”。

定义式：

$$\binom{n}{m} = \frac{\mathrm{A}_n^m}{m!} = \frac{n!}{m!(n-m)!}$$

一些最基础的组合数应用、简单排列组合的计数在此省略。由组合意义立得：

$$\binom{a}{b}\binom{b}{c} = \binom{a}{c}\binom{a-c}{b-c}$$

二项式定理

二项式定理（可以逆用）：

$$(x+y)^n=\sum_{i=0}^{n}\binom{n}{i}x^iy^{n-i}$$

注意二项式定理同样适用于上升幂、下降幂。只需把所有幂次替换为上升幂、下降幂即可。

范德蒙德卷积

范德蒙德卷积：

$$\sum_{i=0}^{k}\binom{n}{i}\binom{m}{k-i}=\binom{n+m}{k}$$

证明可通过组合意义得到。
范德蒙德卷积适用于广义上的组合数（用下降幂定义的）。

组合数前缀和

组合数列和：

$$\binom{n}{r}+\binom{n-1}{r}+\cdots+\binom{r}{r}=\sum_{i=r}^{n}\binom{i}{r}=\binom{n+1}{r+1}$$

组合数行求和可以莫队，每次扩展端点是 $O(1)$ 的。

Lucas 定理

Lucas 定理：

$$\binom{n}{m} \bmod p = \binom{\lfloor n/p \rfloor}{\lfloor m/p \rfloor} \binom{n \bmod p}{m \bmod p} \bmod p$$

这里只列举了一些常用组合数公式。

容斥原理

容斥原理的一种形式：

$$\left|\bigcup_{i\in S} A_i\right| = \sum_{T\subseteq S, T\neq\varnothing} (-1)^{|T|-1} \left|\bigcap_{j\in T} A_j\right|$$

通俗地说，你现在要求满足 m 个限制的方案数。但是同时满足每个限制不好求，那就考虑钦定其中若干个限制不满足，其他限制任意。假设钦定了 k 个限制，那么容斥系数就是 $(-1)^k$。

二项式反演

首先有一组基础形式：

$$f_n=\sum_{i=0}^{n}\binom{n}{i}g_i \Leftrightarrow g_n=\sum_{i=0}^{n}(-1)^{n-i}\binom{n}{i}f_i$$

还有更常用的形式：

$$f_i=\sum_{j=i}^{n}\binom{j}{i}g_j \Leftrightarrow g_i=\sum_{j=i}^{n}(-1)^{j-i}\binom{j}{i}f_j$$

二项式反演

两种形式其实十分相似，注意到 -1 的系数与组合数有关。
二项式反演的本质是容斥原理。f_i 表示钦定选 i 个的方案数，g_i 表示恰好选 i个的方案数。
可以发现，对于任意的 $i \geq n$，g_i 在 f_n 中被计算了 $\binom{i}{n}$ 次，故 $f_n = \sum_{i=n}^{m} \binom{i}{n} g_i$，其中 m 是数目上界，恰好对应常用的形式。
值得注意钦定 k 个和至少 k 个有本质区别。
不过二项式反演也不一定用来钦定转恰好，很多时候可以推式子然后注意到这是二项式反演形式。

[ARC121E] Directed Tree

给定一棵有根树，根结点为 1，结点 i 的父亲为 p_i，边的方向由父亲连向儿子。定义一个 $1 \sim n$ 的排列 a 是合法的，当且仅当对于任意 i，不存在 $a_i \to i$ 的，经过至少一条边的路径。对合法排列计数，答案对 998244353 取模。

$1 \leq n \leq 2 \times 10^3$。

[ARC121E] Directed Tree

先注意到这个限制很难搞，考虑容斥，钦定几个不合法。条件等价于 a_i 是 i 祖先。
但是 a_i 是 i 祖先还是不大好，因为每个点可以同时是很多个点的祖先，所以我们考虑 a 的逆排列 b，那么条件就是 i 是 b_i 的祖先了。
这个可以直接 DP：设 $dp_{i,j}$ 表示做了 i 子树，其中有 j 个钦定的点的方案数。

[ZJOI2022] 树

求有多少种两棵树满足，第一棵树的 $fa_i < i$，第二棵树的 $fa_i > i$，且每个点恰好在其中一棵树内是叶子。答案对 M 取模。

$n \leq 500$

[ZJOI2022] 树

求有多少种两棵树满足，第一棵树的 $fa_i < i$，第二棵树的 $fa_i > i$，且每个点恰好在其中一棵树内是叶子。答案对 M 取模。

$n \le 500$

设 $f(S)$ 表示第一棵树的非叶子集合恰好为 S 时构造第一棵树的方案数，设 $g(T)$ 表示第二棵树的非叶子集合恰好为 T 时构造第二棵树的方案数。
那么答案为

$$Ans = \sum_{S\cap T=\varnothing, S\cup T=\{1,2,\cdots,n\}} f(S)g(T)$$

直接求这个不太好求，我们考虑容斥。

[ZJOI2022] 树

设 $f'(S)$ 表示第一棵树的非叶子集合包含于 S 时构造第一棵树的方案数，设 $g'(T)$ 表示第二棵树的非叶子集合包含于 T 时构造第二棵树的方案数。

$$
\begin{aligned}
Ans &= \sum_{S\cap T=\varnothing, S\cup T=\{1,2,\cdots,n\}} f(S)g(T) \\
&= \sum_{S\cap T=\varnothing, S\cup T=\{1,2,\cdots,n\}} \sum_{S'\subseteq S, T'\subseteq T} f'(S')g'(T')(-1)^{|S|-|S'|+|T|-|T'|} \\
&= \sum_{S'\cap T'=\varnothing} f'(S')g'(T')(-1)^{n-|S'|-|T'|}2^{n-|S'|-|T'|} \\
&= \sum_{S'\cap T'=\varnothing} f'(S')g'(T')(-2)^{n-|S'|-|T'|}
\end{aligned}
$$

其中倒数第二个等号的理由是：不属于 S' 和 T' 的那些元素可能在 S 中，也可能在 T 中。

[ZJOI2022] 树

然后可以 DP，我们设 $dp_{i,j,k}$ 表示确定了 $[1,i]$ 中的点在 S' 和 T' 中的情况，$|\{1,\cdots,i\}\cap S'|=j$，$|\{i+1,\cdots,n\}\cap T'|=k$，且我们已经为 $(1,i]$ 选好了第一棵树中的父亲，为 $[1,i)$ 选好了第二棵树中的父亲时，这些父亲的方案数。

转移时，分类讨论 i 在 S' 和 T' 中的情况，然后确定 i 在第一棵树中的父亲和 $i-1$ 在第二棵树中的父亲：

1. i 属于 S'，$dp_{i-1,j,k}$ 转移到 $dp_{i,j+1,k}$，系数为 $j\times k$。
2. i 属于 T'，$dp_{i-1,j,k}$ 转移到 $dp_{i,j,k-1}$，系数为 $j\times k$。
3. i 两个都不属于，$dp_{i-1,j,k}$ 转移到 $dp_{i,j,k}$，系数为 $-2\times j\times k$。

ABC266G Yet Another RGB Sequence

求出有多少字符串由 x 个 'r'，y 个 'g'，z 个 'b' 构成，且包含 k 个 'rg' 子串。

$x, y, z \leq 10^6$。

ABC266G Yet Another RGB Sequence

求出有多少字符串由 x 个 'r'，y 个 'g'，z 个 'b' 构成，且包含 k 个 'rg' 子串。
$x, y, z \leq 10^6$。

恰好 k 个子串，一看就可以转成钦定 k 个然后反演，而且钦定 k 个想必是好算的。直接用组合数算出钦定的是哪 k 个，然后剩下的怎么填也是组合数。

P4859 已经没有什么好害怕的了

“糖果”和“药片”各有 n 个，每个都有能量，现在将其两两匹配，求：糖果比药片能量大的组数比“药片”比“糖果”能量大的组数多 k 组的方案数。
$n \le 2000$。

P4859 已经没有什么好害怕的了

“糖果”和“药片”各有 n 个，每个都有能量，现在将其两两匹配，求：糖果比药片能量大的组数比“药片”比“糖果”能量大的组数多 k 组的方案数。
$n \leq 2000$。

首先这个多 k 组是可以算出糖果比药片能量大的组数 x 的。我们考虑先把两个序列排序然后直接 DP，那就是 $dp_{i,j}$ 表示第一个序列做到第 i 位置，有 j 次匹配到了小于第一个序列的数。那么每次转移如果匹配的是小的，那是容易的，但如果匹配的是大的，那就不方便记录状态了。因此我们统一把第二种匹配都不记录，最后再匹配，只考虑第一种匹配的 DP 是容易的，然后你会发现这就是钦定的形式，做一个二项式反演即可。

P6478 [NOI Online 2 提高组] 游戏

$2m$ 个点的树上有 m 个黑点和 m 个白点，对于每个 k 求出有多少个两两匹配的方案，满足恰好有 k 对是祖先关系。
$2m \leq 5000$。

P6478 [NOI Online 2 提高组] 游戏

$2m$ 个点的树上有 m 个黑点和 m 个白点，对于每个 k 求出有多少个两两匹配的方案，满足恰好有 k 对是祖先关系。
$2m \leq 5000$。

考虑类似的套路，转恰好为钦定，然后可以考虑 DP 出这个钦定 k 个的答案。考虑 $dp_{i,j}$ 表示 i 的子树内已经匹配了 j 对黑白点的方案数，每次做一个树形背包，再单独考虑子树的根节点是匹配还是不匹配。最后算答案反演一下。

第一类斯特林数

$\begin{bmatrix} n \\ k \end{bmatrix}$，也可记做 $s(n,k)$，表示将 n 个两两不同的元素，划分为 k 个互不区分的非空轮换的方案数。
递推式：

$$\begin{bmatrix} n \\ k \end{bmatrix} = \begin{bmatrix} n-1 \\ k-1 \end{bmatrix} + (n-1)\begin{bmatrix} n-1 \\ k \end{bmatrix}$$

边界是 $\begin{bmatrix} n \\ 0 \end{bmatrix} = [n=0]$。

$$n! = \sum_{i=1}^{n} \begin{bmatrix} n \\ i \end{bmatrix}$$

排列与轮换一一对应。

第一类斯特林数

构造同行第一类斯特林数的生成函数，即

$$F_n(x) = \sum_{i=0}^{n} \begin{bmatrix} n \\ i \end{bmatrix} x^i$$

根据递推公式，不难写出

$$F_n(x) = (n-1)F_{n-1}(x) + xF_{n-1}(x)$$

于是

$$F_n(x) = \prod_{i=0}^{n-1}(x+i) = \frac{(x+n-1)!}{(x-1)!}$$

这其实是 x 的 n 次上升阶乘幂，记做 $x^{\overline{n}}$。可以倍增计算。

P4609 [FJOI2016] 建筑师

T 组询问，每次给定 n, A, B，求有多少长度为 n 的排列 p，满足前缀最大值个数为 A，后缀最大值个数为 B，对 10^9+7 取模。

$T \le 2 \times 10^5, n \le 5 \times 10^3, A, B \le 100$。

P4609 [FJOI2016] 建筑师

T 组询问，每次给定 n, A, B，求有多少长度为 n 的排列 p，满足前缀最大值个数为 A，后缀最大值个数为 B，对 10^9+7 取模。
$T \le 2 \times 10^5, n \le 5 \times 10^3, A, B \le 100$。

我们考虑先取出最大值，那么前缀最大值和后缀最大值都不会延伸过这个位置。以前缀最大值为例，考虑一个前缀最大值及其后面的数到下一个前缀最大值之前的区间，假设中间部分有 k 个元素，包括这个前缀最大值有 $k+1$ 个元素，由于可以任意排列，那么我们把这 $k+1$ 个元素分为一组的贡献为 $k!$（因为只能选择最大值为第一个）。那么其实就是任意分组，一个大小为 k 的组贡献是 $(k-1)!$，那这不是第一类斯特林数嘛。注意到每个组还可以放在左或者右，最后也就是在 $A+B-2$ 个组中选 $A-1$ 个放在左边，这就是一个组合数。所以答案就是一个斯特林数乘一个组合数，容易预处理后 $O(1)$ 回答。

第二类斯特林数

第二类斯特林数（斯特林子集数）

$\begin{Bmatrix} n \\ k \end{Bmatrix}$，也可记做 $S(n,k)$，表示将 n 个两两不同的元素，划分为 k 个互不区分的非空子集的方案数。

递推式

$$\begin{Bmatrix} n \\ k \end{Bmatrix} = \begin{Bmatrix} n-1 \\ k-1 \end{Bmatrix} + k \begin{Bmatrix} n-1 \\ k \end{Bmatrix}$$

边界是 $\begin{Bmatrix} n \\ 0 \end{Bmatrix} = [n=0]$。

通项公式：

$$\begin{Bmatrix} n \\ m \end{Bmatrix} = \sum_{i=0}^{m} \frac{(-1)^{m-i} i^n}{i!(m-i)!}$$

使用容斥原理（或二项式反演）可以证明该公式。

下降幂、普通幂和上升幂

可以利用下面的恒等式将上升幂转化为普通幂：

$$x^{\overline{n}} = \sum_k \begin{bmatrix} n \\ k \end{bmatrix} x^k$$

如果将普通幂转化为上升幂，则有下面的恒等式：

$$x^n = \sum_k \begin{Bmatrix} n \\ k \end{Bmatrix} (-1)^{n-k} x^{\overline{k}}$$

下降幂、普通幂和上升幂

可以利用下面的恒等式将普通幂转化为下降幂（最重要）：

$$x^n = \sum_k \left\{ {n \atop k} \right\} x^{\underline{k}}$$

如果将下降幂转化为普通幂，则有下面的恒等式：

$$x^{\underline{n}} = \sum_k \left[{n \atop k} \right] (-1)^{n-k} x^k$$

[省选联考 2020 A 卷] 组合数问题

计算：

$$\left(\sum_{k=0}^{n} f(k) \times x^k \times \binom{n}{k}\right) \bmod p$$

$1 \le n, x, p \le 10^9, 0 \le a_i \le 10^9, 0 \le m \le \min(n, 1000)$。

[省选联考 2020 A 卷] 组合数问题

注意到普通幂和组合数没有什么好的性质，但下降幂和组合数有。所以我们可以先用第二类斯特林数把普通幂多项式 f 转化成下降幂多项式 b。稍微推一下式子：

$$\sum_{k=0}^{n}\sum_{i=0}^{m} b_i \times k^{\underline{i}} \times x^k \times \binom{n}{k}$$

$$\sum_{i=0}^{m} b_i \sum_{k=0}^{n} k^{\underline{i}} \times x^k \times \binom{n}{k}$$

$$\sum_{i=0}^{m} b_i \sum_{k=i}^{n} n^{\underline{i}} \times x^k \times \binom{n-i}{k-i}$$

[省选联考 2020 A 卷] 组合数问题

$$\sum_{i=0}^{m} b_i \sum_{k=i}^{n} n^{\underline{i}} \times x^k \times \binom{n-i}{k-i}$$

$$\sum_{i=0}^{m} b_i \times n^{\underline{i}} \times x^i \sum_{k=i}^{n} x^{k-i} \times \binom{n-i}{k-i}$$

$$\sum_{i=0}^{m} b_i \times n^{\underline{i}} \times x^i \times (1+x)^{n-i}$$

CF1278F Cards

有 m 张牌，其中一张是王牌。现在你执行 n 次如下操作：洗牌后查看第一张牌是什么。令 x 为洗牌后第一张牌为王牌的次数，现在假设洗牌时 $m!$ 种牌的排列出现的概率均相等，求 x^k 的期望。

$n, m < 998244353$，$k \leq 5000$。

CF1278F Cards

有 m 张牌，其中一张是王牌。现在你执行 n 次如下操作：洗牌后查看第一张牌是什么。令 x 为洗牌后第一张牌为王牌的次数，现在假设洗牌时 $m!$ 种牌的排列出现的概率均相等，求 x^k 的期望。

$n, m < 998244353$，$k \le 5000$。

下降幂组合意义经典例题。

记一次翻到王的概率为 p。考虑把 x^k 转成下降幂：

$$x^k = \sum_i \left\{ {k \atop i} \right\} x^{\underline{i}} = \sum_i \left\{ {k \atop i} \right\} i! \binom{x}{i}$$

CF1278F Cards

代原式去：

$$\sum_{x=0}^{n}\binom{n}{x}p^x(1-p)^{n-x}x^k$$

$$=\sum_{x=0}^{n}\binom{n}{x}p^x(1-p)^{n-x}\sum_{i=0}^{x}\left\{{k \atop i}\right\}i!\binom{x}{i}$$

$$=\sum_{i=0}^{n}\left\{{k \atop i}\right\}i!\sum_{x=i}^{n}\binom{x}{i}\binom{n}{x}p^x(1-p)^{n-x}$$

$$=\sum_{i=0}^{n}\left\{{k \atop i}\right\}i!\sum_{x=i}^{n}\binom{n}{i}\binom{n-i}{x-i}p^x(1-p)^{n-x}$$

后面一大堆只需要逆用一下二项式定理拆一下。

斯特林反演

$$f_i=\sum_{j=i}^{n}\left\{ {j \atop i} \right\} g_j \longleftrightarrow g_i=\sum_{j=i}^{n}(-1)^{j-i}\left[{j \atop i} \right] f_j$$

$$f_i=\sum_{j=1}^{i}\left\{ {i \atop j} \right\} g_j \longleftrightarrow g_i=\sum_{j=1}^{i}(-1)^{i-j}\left[{i \atop j} \right] f_j$$

广义斯特林反演 / 集合划分容斥

限定两两元素互不相同，求方案数。考虑一个等价类集合为 S，那么容斥系数为 $(-1)^{|S|-1}(|S|-1)!$。
证明考虑钦定一个等价类的时候其实是要计算在 S 选边使得联通的所有情况的容斥系数和。

贝尔数

B_n 是基数为 n 的集合的划分方法的数目。集合 S 的一个划分是定义为 S 的两两不相交的非空子集的族，它们的并是 S。每个贝尔数都是相应的第二类斯特林数的和。
贝尔数适合的递推公式：

$$B_{n+1} = \sum_{k=0}^{n} \binom{n}{k} B_k$$

证明的话，考虑最后一个元素和哪些元素分到了一个组即可。

卡特兰数

$$H_0 = 1, H_1 = 1$$

$$H_n = \sum_{i=1}^{n} H_{i-1} H_{n-i}$$

对以上递推，存在通项公式：

$$H_n = \frac{\binom{2n}{n}}{n+1} = \binom{2n}{n} - \binom{2n}{n-1}$$

最主要的应用为合法括号串计数。

伯努利数

记

$$S_m(n)=\sum_{k=0}^{n-1}k^m$$

众所周知，$S_m(n)$ 是关于 n 的 $m+1$ 次多项式。接下来用伯努利数刻画这个多项式：

$$S_m(n)=\frac{1}{m+1}\sum_{k=0}^{m}\binom{m+1}{k}B_k n^{m+1-k}$$

其中 B 为伯努利数。
当需要求伯努利数时，在上式中代入 $n=1$ 即可，有：

$$\frac{1}{m+1}\sum_{k=0}^{m}\binom{m+1}{k}B_k=[m=0]$$

Min-Max 容斥

$$\max_{i\in S} a_i = \sum_{T\subset S, T\neq\varnothing} (-1)^{|T|-1} \min_{j\in T} a_j$$

$$\min_{i\in S} a_i = \sum_{T\subset S, T\neq\varnothing} (-1)^{|T|-1} \max_{j\in T} a_j$$

Min-Max 容斥一个很经典的使用是其期望形式，将一个集合最后出现的时间期望转化成第一个出现的时间期望。
在数论角度我们可以得到 gcd-lcm 容斥：

$$\operatorname{lcm}(S) = \prod_{T\subset S, T\neq\varnothing} \gcd(T)^{(-1)^{|T|-1}}$$

[HAOI2015]按位或

刚开始你有一个数字 0，每一秒钟你会随机选择一个 $[0, 2^n - 1]$ 的数字，与你手上的数字进行或操作。选择数字 i 的概率是 p_i。保证 $0 \le p_i \le 1$，$\sum p_i = 1$ 。问期望多少秒后，你手上的数字变成 $2^n - 1$。

$n \le 20$。

[HAOI2015]按位或

刚开始你有一个数字 0，每一秒钟你会随机选择一个 $[0, 2^n - 1]$ 的数字，与你手上的数字进行或操作。选择数字 i 的概率是 p_i。保证 $0 \leq p_i \leq 1$，$\sum p_i = 1$ 。问期望多少秒后，你手上的数字变成 $2^n - 1$。
$n \leq 20$。

不妨令 $\max(S)$ 表示 S 集合全部变成 1 的时间，也就是最大时间；$\min(S)$ 表示 S 集合有一个变成 1 的时间，也就是最小时间。我们需要求的就是 $E(\max(U))$，根据 Min-Max 容斥，有：

$$E(\max(U)) = \sum_{S \subset U, S \neq \varnothing} (-1)^{|S|-1} E(\min(S))$$

所以我们需要对每个集合 S，求出 $E(\min(S))$。对于每个 S，我们如果选 T 满足 $T \cap S \neq \varnothing$，那么就会立刻停止。我们只要算出 $\sum P_T$ 即可，使用高维前缀和解决。

格路计数问题 I

你需要从 $(0,0)$ 走到 (n,m)，在此过程中你不能经过直线 $y=x-b$，求方案数。

格路计数问题 I

你需要从 $(0,0)$ 走到 (n,m)，在此过程中你不能经过直线 $y=x-b$，求方案数。

考虑先算出总方案数，是 $\binom{n+m}{n}$，再减掉不合法的。考虑假设不合法的路径第一个经过的直线上的点是 (x_0,y_0)，那我们考虑把 $(0,0)$ 关于直线对称一下，假设对称到 (x,y)，那么可以发现这条路径和 (x,y) 到 (n,m) 的路径是一一对应的。所以不合法的路径数就等于 (x,y) 到 (n,m) 的路径数。
$b=1$ 且 $n=m$ 就是卡特兰数，所以也可以用这个证明卡特兰数通项公式。

格路计数问题 II

你需要从 $(0,0)$ 走到 (n,m)，在此过程中你不能经过直线 $f=x-b$ 和 $g=x+c$，求方案数。

格路计数问题 II

你需要从 $(0,0)$ 走到 (n,m)，在此过程中你不能经过直线 $f=x-b$ 和 $g=x+c$，求方案数。

考虑上一题的方法能否搬进来，但是一条线可能先经过 f 再经过 g 然后又经过 f，这样很难计算。那么我们考虑容斥，我们用 总方案数 $-$ 过 f 的方案数 $-$ 过 g 的方案数 $+$ 过 f,g 的方案数 $+$ 过 g,f 的方案数 $-$ 过 f,g,f 的方案数 $-$ 过 g,f,g 的方案数 ……

每个的方案数都是用上一题的方法算，例如过 g,f,g 的方案数就是先关于 g 做对称，再关于 f 对称，在关于 g 对称，得到的点到达 (n,m) 的方案数。这样计算复杂度是 $O(\frac{n+m}{b+c})$。
我们把这个模型叫做“反射容斥”。

格路计数问题 III（dyck 路）

你需要从 $(0,0)$ 走到 (n,m)，在此过程中你不能经过直线 $y=\frac{m}{n}x$，求方案数。保证 n,m 互质。

格路计数问题 III（dyck 路）

你需要从 $(0,0)$ 走到 (n,m)，在此过程中你不能经过直线 $y=\frac{m}{n}x$，求方案数。保证 n,m 互质。

考察任意一条路径，找到以 $y=\frac{m}{n}x$ 线为基准最低的点。假设这个点低于 $y=\frac{m}{n}x$ 就不合法，否则合法。注意到 n,m 互质，所以这样的点只能有一个（$(0,0$ 和 (n,m) 视为一个点，在循环意义下）。考察这个 $n+m$ 长度的序列的任意循环移位，有且仅有一种方式是合法的（就是把这个最小值放在 $(0,0)$ 的时候），所以答案是 $\frac{\binom{n+m}{n}}{n+m}$。

UOJ424 [集训队作业2018] count

定义对于一个序列 A，$f_A(l,r)$ 表示 $[l,r]$ 中最大值的下标，有多个就取下标最小的。定义两个序列 A,B 本质相同，当且仅当对于任意 l,r 有 $f_A(l,r)=f_B(l,r)$。求出有多少本质不同的序列满足值域 $1\sim m$ 且 $1\sim m$ 每个数都出现过。

$n,m\le 10^5$。

UOJ424 [集训队作业2018] count

定义对于一个序列 A，$f_A(l,r)$ 表示 $[l,r]$ 中最大值的下标，有多个就取下标最小的。定义两个序列 A,B 本质相同，当且仅当对于任意 l,r 有 $f_A(l,r)=f_B(l,r)$。求出有多少本质不同的序列满足值域 $1\sim m$ 且 $1\sim m$ 每个数都出现过。
$n,m\le 10^5$。

若我们对序列建笛卡尔树，容易发现两个序列同构，当且仅当它们的笛卡尔树形状相同。并且不同构的序列，和最长左链（某个节点到根的路径中，作为左儿子的次数 $+1$）不超过 m 的笛卡尔树是一一对应的。反过来，用归纳法可以证明，一棵满足上述条件的笛卡尔树，必然能构造出对应的序列。

UOJ424 [集训队作业2018] count

问题转化成求 n 个节点且最长左链不超过 m 的笛卡尔树数量。考虑把二叉树转化成括号序列：假设左儿子的括号序列是 s_l，右儿子的括号序列是 s_r，那么定义这个节点对应的就是 $(s_l)s_r$。这样二叉树和括号序列是一一对应的，但是我们还需要左链 $\leq m$，对应到括号序列上容易发现是任何时刻栈中左括号 $\leq m$。这个问题做一下格路计数问题 II 即可。

P5825 排列计数

求有多少个长 n 的排列恰好有 k 个位置满足 $a_i < a_{i+1}$。
$n \le 2 \times 10^5$。

P5825 排列计数

求有多少个长 n 的排列恰好有 k 个位置满足 $a_i < a_{i+1}$。
$n \le 2 \times 10^5$。

考虑先对升高容斥，钦定 k 个位置是升高，其他位置任意。这时将序列分为了 $n-k$ 段，每段需要满足单调递增。问题变成了把 $1 \sim n$ 放入 $n-k$ 个有标号的集合，使得每个集合不为空的方案数。
再对空集合容斥。钦定这 $n-k$ 个集合中的 p 个集合为空，其他集合任意。那么问题就变成把 $1 \sim n$ 放入 $n-k-p$ 个有标号的集合的方案数，这个问题是小学数学题，答案就是 $(n-k-p)^n$，可以快速幂处理。最后卷两次。

P10004 [集训队互测 2023] Permutation Counting 2

给定 n，对于每组 $x, y \in [0, n)$ 求出有多少个 $1 \sim n$ 的排列 p 满足以下条件：

- $\sum_{i=1}^{n-1} [p_i < p_{i+1}] = x$。
- $\sum_{i=1}^{n-1} [p_i^{-1} < p_{i+1}^{-1}] = y$。

其中 p^{-1} 表示 p 的逆排列，满足 $p_{p_i}^{-1} = i$。

答案对给定的质数 MOD 取模。

$1 \le n \le 500$，$10^9 \le MOD \le 1.01 \times 10^9$，保证 MOD 为质数。

P10004 [集训队互测 2023] Permutation Counting 2

这个题相当于是上一题二维上的问题。那么我们对两维分别做一个容斥（也就是二项式反演），每个容斥是 $O(n^3)$ 的，这样两维都转成钦定。接下来就是求 $f(a,b)$ 表示排列和逆排列分别被钦定了 a,b 个非空上升段。考虑画出一个 $a \times b$ 的表格，按值域从小到大将逆排列的上升段填入排列的上升段，那么每个值域上升段一定依次填入排列上升段，所以如果我们确定表格里的每个数，我们就知道第 i 个值域上升段向第 j 个排列上升段填入了多少个数。这个表格需要满足和为 n，以及没有一行或一列全是 0，那么对这个再做一次容斥即可。
总共四次容斥，复杂度 $O(n^3)$。

P3349 [ZJOI2016]小星星

给定 n 个点和 n 个点的图，求有多少种将树上的点和图上的点对应的方式，满足树上的边可以对应到图上。
$n \leq 17$。

P3349 [ZJOI2016]小星星

给定 n 个点和 n 个点的图，求有多少种将树上的点和图上的点对应的方式，满足树上的边可以对应到图上。
$n \leq 17$。

首先有一个状压 DP，考虑 $dp_{i,j,S}$ 表示做到 i 这个子树，i 对应的点是 j，现在已经使用了 S 这个集合的方案数。这样每次转移需要枚举子集，复杂度大概是 $O(3^n n^3)$ 的，过不去。
发现主要的瓶颈在于枚举子集，我们希望把这个东西干掉。那么考虑容斥，原本的排列 p 定义是每个点都被选过了，现在我们钦定 S 集合不能被选择，这容斥系数是 $(-1)^{|S|}$。接下来对于每个 S 我们做一个 DP 即可，定义 $dp_{i,j}$ 表示做完了 i 子树，把 i 对应图上的 j 的方案数，一次 DP 的复杂度是 $O(n^3)$ 的，那么总复杂度就是 $O(2^n n^3)$。

DAG 计数

对 n 个点带标号的有向无环图进行计数，对 10^9+7 取模。
$n \leq 5 \times 10^3$。

DAG 计数

对 n 个点带标号的有向无环图进行计数，对 10^9+7 取模。
$n \le 5 \times 10^3$。

DAG 计数的经典套路：剥掉入度（或出度）为 0 点进行 DP 转移。这题如果直接枚举有 k 个入度为 0，需要 $O(n^2)$ 的状态，这样复杂度就变成了 $O(n^3)$。
变成钦定 j 个点入度为 0，对选出的 j 个点，这 j 个点可以和剩下的 $i-j$ 个点有任意的连边，即 $(2^{i-j})^j = 2^{(i-j)j}$ 种情况：

$$f_i = \sum_{j=1}^{i} (-1)^{j-1} \binom{i}{j} 2^{(i-j)j} f_{i-j}$$

为什么钦定 j 个点的容斥系数是 $(-1)^{j-1}$？考虑假设实际上有 3 个点度是 0，那么钦定 1 个点会来带 $+3$ 的贡献，钦定 2 个点会来带 -3 的贡献，钦定 3 个点会来带 $+1$ 的贡献。也就是最开始的那个容斥原理的式子。

UOJ37 主旋律

有一个含 n 个点，m 条边的有向图，保证强连通。现在我们想删除一些边，使得该有向图仍然强连通。求方案数。

$n \leq 15$。

UOJ37 主旋律

有一个含 n 个点，m 条边的有向图，保证强连通。现在我们想删除一些边，使得该有向图仍然强连通。求方案数。
$n \le 15$。

考虑到强连通图不大好搞，能转换成我们之前会做的 DAG 就好了。事实上确实是可以转化过去的。
考虑 $f(S)$ 表示 S 集合删掉一些边形成强连通图的方案数，$g(S)$ 表示 S 集合删掉一些边形成非强连通图的方案数。考虑将非强连通图缩点得到 DAG，在 DAG 上枚举出度为 0 的子集做容斥。假设枚举的 0 度子集是 T，内含 k 个强连通分量，那么容斥系数为 $(-1)^{k-1}$。所以我们还要做一个 DP 算出 T 内部的强连通分量划分，带上容斥系数的贡献。转移时枚举 $\mathrm{lowbit}(S)$ 所在的强连通分量即可。

[ARC105F] Lights Out on Connected Graph

有一张 n 个点 m 条边的简单无向图，问选出一个边集，使得 n 个点与这些边构成的图连通，并且图是二分图的方案数。
对 998244353 取模。
$1 \le n \le 17, n-1 \le m \le \frac{n(n-1)}{2}$。

[ARC105F] Lights Out on Connected Graph

有一张 n 个点 m 条边的简单无向图，问选出一个边集，使得 n 个点与这些边构成的图连通，并且图是二分图的方案数。
对 998244353 取模。
$1 \leq n \leq 17, n-1 \leq m \leq \frac{n(n-1)}{2}$。

考虑不对二分图计数而对二分图染色计数。因为联通，那么一个二分图一定对应两个二分图染色。我们可以记 g_i 表示 i 自己二分图染色的方案数，求法就枚举黑色点的子集即可。
但是我们还要保证联通，用一下上面几个容斥的套路。dp_S 表示 S 自己构成二分图染色且联通的方案数。对其做容斥就行，枚举 lowbit 所在连通块。

结语

谢谢大家!