计数

liuhengxi

2024-08-02

废话

本课件使用 Typst 的 polylux package 编写。

本次讲课面向 NOI 及以前的比赛，因此不会涉及过难内容，较少涉及超纲内容。

废话

本课件使用 Typst 的 polylux package 编写。

本次讲课面向 NOI 及以前的比赛，因此不会涉及过难内容，较少涉及超纲内容。

严格地说，计数类 DP 不算 DP。但是为了方便，这里仍然称其为 DP。

目录

基础

加法原理、乘法原理

注意加法原理、乘法原理可以带上权：

$$\sum_{x\in A\cup B} f(x) = \sum_{x\in A} f(x) + \sum_{x\in B} f(x), (A\cap B = \varnothing)$$

$$\sum_{x\in A\times B} f(x) = \left(\sum_{x\in A} f(x)\right)\left(\sum_{x\in B} f(x)\right), f((a,b)) = f(a)f(b)$$

不带权即为 $f(x) = 1$。

加法原理、乘法原理

注意加法原理、乘法原理可以带上权：

$$\sum_{x\in A\cup B} f(x) = \sum_{x\in A} f(x) + \sum_{x\in B} f(x), (A\cap B=\varnothing)$$

$$\sum_{x\in A\times B} f(x) = \left(\sum_{x\in A} f(x)\right)\left(\sum_{x\in B} f(x)\right), f((a,b)) = f(a)f(b)$$

不带权即为 $f(x)=1$。

在计数问题中，要计算的也可以不是数量，而是所有可能的元素的某个指标之和。之后提到的大多数等式，都可以带上权。

减法原理、除法原理

- 减法原理常用于容斥

减法原理、除法原理

- 减法原理常用于容斥
- 除法原理常用于 k 倍映射。

Problem 1：【UER #2】手机的生产

给定一个由 `fork()`, &&, `||` 构成的表达式，没有括号，&& 优先级高于 `||`。

&& 和 `||` 进行短路求值（若在求值第一个操作数后结果已知，则不求值第二个）。

从左到右计算表达式，每次调用 `fork()` 时，当前线程分裂成两个线程， 一个线程中 `fork()` 返回 `false`，另一个返回 `true`。

求最终会有多少个线程。对 998244353 取模。

`fork()` 的数量 $\leq 10^5$。

1s, 256MiB

引入：十二重计数法

有 n 个球和 m 个盒子，要全部装进盒子里。还有一些限制条件，那么有多少种方法放球？（与放的先后顺序无关）

引入：十二重计数法

有 n 个球和 m 个盒子，要全部装进盒子里。还有一些限制条件，那么有多少种方法放球？（与放的先后顺序无关）

球之间互不相同	无要求	每个盒子至多装一个球	每个盒子至少装一个球
盒子之间互不相同	**I**	**II**	**III**
盒子全部相同	**IV**	**V**	**VI**
球之间全部相同	无要求	每个盒子至多装一个球	每个盒子至少装一个球
盒子之间互不相同	**VII**	**VIII**	**IX**
盒子全部相同	**X**	**XI**	**XII**

引入：十二重计数法

有 n 个球和 m 个盒子，要全部装进盒子里。还有一些限制条件，那么有多少种方法放球？（与放的先后顺序无关）

球之间互不相同	无要求	每个盒子至多装一个球	每个盒子至少装一个球
盒子之间互不相同	**I**	**II**	**III**
盒子全部相同	**IV**	**V**	**VI**
球之间全部相同	无要求	每个盒子至多装一个球	每个盒子至少装一个球
盒子之间互不相同	**VII**	**VIII**	**IX**
盒子全部相同	**X**	**XI**	**XII**

一部分简单的会在本节说明。

I, II, V, VII, VIII, IX, XI

球之间互不相同	无要求	每个盒子至多装一个球	每个盒子至少装一个球
盒子之间互不相同	**I**	**II**	**III**
盒子全部相同	**IV**	**V**	**VI**
球之间全部相同	无要求	每个盒子至多装一个球	每个盒子至少装一个球
盒子之间互不相同	**VII**	**VIII**	**IX**
盒子全部相同	**X**	**XI**	**XII**

- **I**: m^n
- **II**: $P(m,n) = \binom{m}{n} n! = \prod_{i=m-n+1}^{m} i$
- **V**: $[n \leq m]$
- **VII**: 隔板法, $\binom{n+m-1}{m-1}$
- **VIII**: $\binom{m}{n}$
- **IX**: 先钦定每个盒子放了一个球, $\binom{n-1}{m-1}$
- **XI**: $[n \leq m]$

二项式系数

二项式定理

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

令 $a=1, b=x$，取 m 次项系数：$[x^m](1+x)^n = \binom{n}{m}$。

二项式定理

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

令 $a=1, b=x$，取 m 次项系数：$[x^m](1+x)^n = \binom{n}{m}$。

推广：

$$\left(\sum x_i\right)^n = \sum_{\sum a_i = n} \binom{n}{a_1, a_2, \ldots, a_n} \prod x_i^{a_i}$$

其中 $\binom{n}{a_1,a_2,\ldots,a_n} = \frac{n!}{\prod a_{i!}}$。

性质

- $\binom{n}{m}$ 的组合意义是在 n 个不同的球中取 m 个的方案数。（前面已经提到。）

性质

- $\binom{n}{m}$ 的组合意义是在 n 个不同的球中取 m 个的方案数。(前面已经提到。)
- $\binom{n}{m} = \binom{n-1}{m} + \binom{n-1}{m-1}$。(可用于 $\mathcal{O}(n\sqrt{q})$ 计算 $\sum_{i=0}^{m}\binom{n}{i}$,$q$ 为询问次数。)

性质

- $\binom{n}{m}$ 的组合意义是在 n 个不同的球中取 m 个的方案数。（前面已经提到。）
- $\binom{n}{m} = \binom{n-1}{m} + \binom{n-1}{m-1}$。（可用于 $\mathcal{O}(n\sqrt{q})$ 计算 $\sum_{i=0}^{m}\binom{n}{i}$，$q$ 为询问次数。）
- $\sum_{i=0}^{n}\binom{n}{i} = 2^n$。（二项式定理的推论。）

性质

- $\binom{n}{m}$ 的组合意义是在 n 个不同的球中取 m 个的方案数。(前面已经提到。)
- $\binom{n}{m} = \binom{n-1}{m} + \binom{n-1}{m-1}$。(可用于 $\mathcal{O}(n\sqrt{q})$ 计算 $\sum_{i=0}^{m}\binom{n}{i}$,$q$ 为询问次数。)
- $\sum_{i=0}^{n}\binom{n}{i} = 2^n$。(二项式定理的推论。)
- 上指标求和:$\sum_{i=m}^{n}\binom{i}{m} = \binom{n+1}{m+1}$。
  - 组合意义:在 $n+1$ 个球中选 $m+1$ 个,枚举最后一个球的位置。
  - 代数推导:

$$\sum_{i=m}^{n}\binom{i}{m} = \sum_{i=m}^{n}\binom{i}{i-m} = \binom{m}{0} + \binom{m+1}{1} + \cdots + \binom{n}{m}$$

$$= \binom{m+1}{0} + \binom{m+1}{1} + \cdots + \binom{n}{m} = \binom{m+2}{1} + \binom{m+2}{2} + \cdots + \binom{n}{m} = \cdots = \binom{n+1}{m+1}。$$

性质

- $\binom{n}{m}$ 的组合意义是在 n 个不同的球中取 m 个的方案数。（前面已经提到。）
- $\binom{n}{m}=\binom{n-1}{m}+\binom{n-1}{m-1}$。（可用于 $\mathcal{O}(n\sqrt{q})$ 计算 $\sum_{i=0}^{m}\binom{n}{i}$，$q$ 为询问次数。）
- $\sum_{i=0}^{n}\binom{n}{i}=2^n$。（二项式定理的推论。）
- 上指标求和：$\sum_{i=m}^{n}\binom{i}{m}=\binom{n+1}{m+1}$。
  - 组合意义：在 $n+1$ 个球中选 $m+1$ 个，枚举最后一个球的位置。
  - 代数推导：

$$\sum_{i=m}^{n}\binom{i}{m}=\sum_{i=m}^{n}\binom{i}{i-m}=\binom{m}{0}+\binom{m+1}{1}+\cdots+\binom{n}{m}$$

$$=\binom{m+1}{0}+\binom{m+1}{1}+\cdots+\binom{n}{m}=\binom{m+2}{1}+\binom{m+2}{2}+\cdots+\binom{n}{m}=\cdots=\binom{n+1}{m+1}。$$

- 范德蒙德卷积：$\sum_i\binom{n}{i}\binom{m}{k-i}=\binom{n+m}{k}$。
  - 组合意义：在 $n+m$ 个球中选 k 个，枚举前 n 个中选了几个。
  - 另一形式：$\sum_i\binom{n}{i}\binom{m}{i+k}=\binom{n+m}{n+k}$。

Problem 2 : Codeforces 1784F. Minimums or Medians

你有一个包含 $1 \sim 2n$ 共 $2n$ 个整数的集合 S。你必须执行恰好 k（$1 \le k \le n \le 10^6$）次操作，每个操作都是以下两种其中之一：

- 将 S 中第 $1, 2$ 个元素删去。
- 将 S 中第 $\frac{|S|}{2}, \frac{|S|}{2}+1$ 个元素删去。（显然 $|S|$ 一直是偶数，所以 $\frac{|S|}{2}$ 也一定是整数）

请你统计，通过这些操作可以获得多少个本质不同的最终集合 S？答案对 998244353 取模。

$1 \le k \le n \le 10^6$。

4s, 512MiB

容斥与反演

容斥与反演的本质是我们要求的若干个答案构成的向量 $\boldsymbol{x}$ 不容易计算，但是 $A\boldsymbol{x}$ 容易计算。

容斥与反演的本质是我们要求的若干个答案构成的向量 $\boldsymbol{x}$ 不容易计算，但是 $A\boldsymbol{x}$ 容易计算。

矩阵 A 通常是三角的，且对角线为 1，因此可逆。我们需要在左边乘上逆矩阵 A^{-1}。

容斥原理

如果满足 n 个条件 $p_1, p_2, ..., p_n$ 中的某一些条件的元素容易计算，如何计算同时满足 $\neg p_1, ..., \neg p_n$ 的元素个数？

容斥原理

如果满足 n 个条件 $p_1, p_2, ..., p_n$ 中的某一些条件的元素容易计算，如何计算同时满足 $\neg p_1, ..., \neg p_n$ 的元素个数？

单个元素带来的贡献为 $(1 - [p_1])(1 - [p_2])\cdots(1 - [p_n])$。

容斥原理

如果满足 n 个条件 $p_1, p_2, ..., p_n$ 中的某一些条件的元素容易计算，如何计算同时满足 $\neg p_1, ..., \neg p_n$ 的元素个数？

单个元素带来的贡献为 $(1-[p_1])(1-[p_2])\cdots(1-[p_n])$。

展开得 $\sum_{S\subseteq\{1,2,...,n\}}(-1)^{|S|}\prod_{i\in S}[p_i]$。

容斥原理

如果满足 n 个条件 $p_1, p_2, ..., p_n$ 中的某一些条件的元素容易计算，如何计算同时满足 $\neg p_1, ..., \neg p_n$ 的元素个数？

单个元素带来的贡献为 $(1 - [p_1])(1 - [p_2])\cdots(1 - [p_n])$。

展开得 $\sum_{S\subseteq\{1,2,...,n\}} (-1)^{|S|} \prod_{i\in S} [p_i]$。

将单个元素改为集合，即可得到容斥原理一个常见的形式：

$$\bigcap_{i=1}^{n} \overline{A_i} = \sum_{S\subseteq\{1,2,...,n\}} (-1)^{|S|} \left| \bigcap_{i\in S} A_i \right|$$

容斥原理

如果满足 n 个条件 $p_1, p_2, \dots, p_n$ 中的某一些条件的元素容易计算，如何计算同时满足 $\neg p_1, \dots, \neg p_n$ 的元素个数？

单个元素带来的贡献为 $(1-[p_1])(1-[p_2])\cdots(1-[p_n])$。

展开得 $\sum_{S\subseteq\{1,2,\dots,n\}}(-1)^{|S|}\prod_{i\in S}[p_i]$。

将单个元素改为集合，即可得到容斥原理一个常见的形式：

$$\bigcap_{i=1}^{n}\overline{A_i}=\sum_{S\subseteq\{1,2,\dots,n\}}(-1)^{|S|}\left|\bigcap_{i\in S}A_i\right|$$

将两遍同时用全集的大小减去，即可得到另一个常见的形式：

$$\bigcup_{i=1}^{n}A_i=\sum_{\varnothing\neq S\subseteq\{1,2,\dots,n\}}(-1)^{|S|-1}\left|\bigcap_{i\in S}A_i\right|$$

Problem 3 : ARC101E - Ribbons on Tree

给定一个大小为 n 的树，保证 n 为偶数。

你需要给树上的点两两配对，对于一对点 (u, v)，在树上将 $u \to v$ 的路径染色，定义一个配对方案合法当且仅当所有边都被染色。

求方案数对 $10^9 + 7$ 取模。

$n \leq 5000$。

2s, 1024MiB

十二重计数法 III

球之间互不相同，盒子之间互不相同，每个盒子至少装一个球。

球之间互不相同，盒子之间互不相同，每个盒子至少装一个球。

需要满足的条件即为对于 m 个盒子，每个盒子都不空。

十二重计数法 III

球之间互不相同，盒子之间互不相同，每个盒子至少装一个球。

需要满足的条件即为对于 m 个盒子，每个盒子都不空。

钦定 m 个盒子的一个子集 S 是空的，装球的方式有 $(m-|S|)^n$ 种，而选一个大小为 i 个子集有 $\binom{m}{i}$ 种方式。

十二重计数法 III

球之间互不相同，盒子之间互不相同，每个盒子至少装一个球。

需要满足的条件即为对于 m 个盒子，每个盒子都不空。

钦定 m 个盒子的一个子集 S 是空的，装球的方式有 $(m-|S|)^n$ 种，而选一个大小为 i 个子集有 $\binom{m}{i}$ 种方式。

因此答案为 $\sum_{i=0}^{m}(-1)^i\binom{m}{i}(m-i)^n$。

二项式反演

如果满足 n 个条件 $p_1, p_2, ..., p_n$ 中的某一些条件的元素容易计算，如何计算满足 $\bigwedge_{i=1}^{n} \begin{cases} p_i & \text{if } i \in S \\ \neg p_i & \text{otherwise} \end{cases}$ 的元素个数？

如果满足 n 个条件 $p_1, p_2, \ldots, p_n$ 中的某一些条件的元素容易计算，如何计算满足 $\bigwedge_{i=1}^{n} \begin{cases} p_i & \text{if } i \in S \\ \neg p_i & \text{otherwise} \end{cases}$ 的元素个数？

对于要求不满足的条件，使用容斥原理，可得：

$$\left| \left(\bigcap_{i \in S} A_i \right) \cap \left(\bigcap_{i \notin S} \overline{A_i} \right) \right| = \sum_{S \subseteq T \subseteq \{1,2,\ldots,n\}} (-1)^{|T|-|S|} \left| \bigcap_{i \in T} A_i \right|$$

二项式反演

如果满足 n 个条件 $p_1, p_2, ..., p_n$ 中的某一些条件的元素容易计算，如何计算满足 $\bigwedge_{i=1}^{n} \begin{cases} p_i & \text{if } i \in S \\ \neg p_i & \text{otherwise} \end{cases}$ 的元素个数？

对于要求不满足的条件，使用容斥原理，可得：

$$\left| \left(\bigcap_{i \in S} A_i \right) \cap \left(\bigcap_{i \notin S} \overline{A_i} \right) \right| = \sum_{S \subseteq T \subseteq \{1,2,...,n\}} (-1)^{|T|-|S|} \left| \bigcap_{i \in T} A_i \right|$$

如果我们不关心一个元素满足了**哪些**条件，而只关心其满足了**几个**条件呢？

对大小相同的 S 求和：

$$
\begin{aligned}
&\sum_{|S|=k}\left|\left(\bigcap_{i\in S} A_i\right)\cap\left(\bigcap_{i\notin S}\overline{A_i}\right)\right| \\
&=\sum_{|S|=k}\sum_{S\subseteq T\subseteq\{1,2,\ldots,n\}}(-1)^{|T|-k}\left|\bigcap_{i\in T} A_i\right| \\
&=\sum_{|S|=k}\sum_{j=k}^{n}(-1)^{j-k}\sum_{S\subseteq T\subseteq\{1,2,\ldots,n\},|T|=j}\left|\bigcap_{i\in T} A_i\right| \\
&=\sum_{j=k}^{n}(-1)^{j-k}\sum_{T\subseteq\{1,2,\ldots,n\},|T|=j}\sum_{S\subseteq T,|S|=k}\left|\bigcap_{i\in T} A_i\right| \\
&=\sum_{j=k}^{n}(-1)^{j-k}\sum_{T\subseteq\{1,2,\ldots,n\},|T|=j}\left|\bigcap_{i\in T} A_i\right|\sum_{S\subseteq T,|S|=k}1 \\
&=\sum_{j=k}^{n}(-1)^{j-k}\binom{j}{k}\sum_{T\subseteq\{1,2,\ldots,n\},|T|=j}\left|\bigcap_{i\in T} A_i\right|
\end{aligned}
$$

如果把**恰好**满足 i 个条件的元素个数记为 f_i，把**钦定**满足 i 个条件的元素个数记为 g_i，则

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

换句话说，

$$\begin{aligned} g_i &= \sum_{j=i}^{n} \binom{j}{i} f_j \\ \Rightarrow f_i &= \sum_{j=i}^{n} (-1)^{j-i} \binom{j}{i} g_j \end{aligned}$$

如果把**恰好**满足 i 个条件的元素个数记为 f_i，把**钦定**满足 i 个条件的元素个数记为 g_i，则

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

换句话说，

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

二项式系数反过来也成立：

$$g_i = \sum_{j=0}^{i} \binom{i}{j} f_j$$
$$\Rightarrow f_i = \sum_{j=0}^{i} (-1)^{j-i} \binom{i}{j} g_j$$

Problem 4 : [NOI Online #2 提高组] 游戏

给定一个 n（偶数）个点的有根树，树上恰有 $\frac{n}{2}$ 个白点和 $\frac{n}{2}$ 个黑点。

对 $k = 0, 1, ..., \frac{n}{2}$，求所有把白点和黑点两两匹配的方案中，有多少方案恰有 k 对点是祖孙关系。

$2 \leq n \leq 5000$。

1s, 512MiB

min-max 容斥

如果序列 a 中一些元素的 min 容易计算，如何计算 $\max\{a_i\}$?

min-max 容斥

如果序列 a 中一些元素的 min 容易计算，如何计算 $\max\{a_i\}$?

利用把序列转化成 01-序列的思想，并假设 a 的元素是非负整数，可得：

$$
\begin{aligned}
&\max\{a_i\} \\
&= \sum_{t=1}^{\infty} [\max\{a_i\} \ge t] \\
&= \sum_{t=1}^{\infty} [\bigvee (a_i \ge t)] \\
&= \sum_{t=1}^{\infty} \sum_{\varnothing \neq S \subseteq \{1,2,\dots,n\}} (-1)^{|S|-1} \left[\bigwedge_{i \in S} (a_i \ge t)\right] \\
&= \sum_{t=1}^{\infty} \sum_{\varnothing \neq S \subseteq \{1,2,\dots,n\}} (-1)^{|S|-1} [\min_{i \in S}\{a_i\} \ge t] \\
&= \sum_{\varnothing \neq S \subseteq \{1,2,\dots,n\}} (-1)^{|S|-1} \sum_{t=1}^{\infty} [\min_{i \in S}\{a_i\} \ge t] \\
&= \sum_{\varnothing \neq S \subseteq \{1,2,\dots,n\}} (-1)^{|S|-1} \min_{i \in S}\{a_i\}
\end{aligned}
$$

min-max 容斥

如果序列 a 中一些元素的 min 容易计算，如何计算 $\max\{a_i\}$？
利用把序列转化成 01-序列的思想，并假设 a 的元素是非负整数，可得：

$$
\begin{aligned}
&\max\{a_i\}\\
&=\sum_{t=1}^{\infty}[\max\{a_i\}\ge t]\\
&=\sum_{t=1}^{\infty}[\bigvee(a_i\ge t)]\\
&=\sum_{t=1}^{\infty}\sum_{\varnothing\ne S\subseteq\{1,2,\ldots,n\}}(-1)^{|S|-1}\left[\bigwedge_{i\in S}(a_i\ge t)\right]\\
&=\sum_{t=1}^{\infty}\sum_{\varnothing\ne S\subseteq\{1,2,\ldots,n\}}(-1)^{|S|-1}[\min_{i\in S}\{a_i\}\ge t]\\
&=\sum_{\varnothing\ne S\subseteq\{1,2,\ldots,n\}}(-1)^{|S|-1}\sum_{t=1}^{\infty}[\min_{i\in S}\{a_i\}\ge t]\\
&=\sum_{\varnothing\ne S\subseteq\{1,2,\ldots,n\}}(-1)^{|S|-1}\min_{i\in S}\{a_i\}
\end{aligned}
$$

该证明容易拓展到实数。

对称地，有：

$$\min\{a_i\} = \sum_{\emptyset \neq S \subseteq \{1,2,\ldots,n\}} (-1)^{|S|-1} \max_{i \in S}\{a_i\}$$

还可以求第 k 大：

$$
\begin{aligned}
& k\text{th-max}\{a_i\} \\
={} & \textstyle\sum_{t=1}^{\infty} [k\text{th-max}\{a_i\} \geq t] \\
={} & \textstyle\sum_{t=1}^{\infty} \sum_{S \subseteq \{1,2,\ldots,n\}, |S| \geq k} [\bigwedge (a_i \geq t) \leftrightarrow (i \in S)] \\
={} & \textstyle\sum_{t=1}^{\infty} \sum_{S \subseteq \{1,2,\ldots,n\}, |S| \geq k} \sum_{S \subseteq T \subseteq \{1,2,\ldots,n\}} (-1)^{|T|-|S|} \left[\bigwedge_{i \in T} a_i \geq t\right] \\
={} & \textstyle\sum_{S \subseteq \{1,2,\ldots,n\}, |S| \geq k} \sum_{S \subseteq T \subseteq \{1,2,\ldots,n\}} (-1)^{|T|-|S|} \min_{i \in T}\{a_i\} \\
={} & \textstyle\sum_{T \subseteq \{1,2,\ldots,n\}} \left(\sum_{S \subseteq T, |S| \geq k} (-1)^{|T|-|S|}\right) \min_{i \in T}\{a_i\} \\
={} & \textstyle\sum_{T \subseteq \{1,2,\ldots,n\}} (-1)^{|T|-k} \binom{|T|-1}{k-1} \min_{i \in T}\{a_i\}
\end{aligned}
$$

汇总：

$$\max\{a_i\} = \sum_{\varnothing \neq S \subseteq \{1,2,\ldots,n\}} (-1)^{|S|-1} \min_{i \in S}\{a_i\}$$

$$\min\{a_i\} = \sum_{\varnothing \neq S \subseteq \{1,2,\ldots,n\}} (-1)^{|S|-1} \max_{i \in S}\{a_i\}$$

$$k\text{th-}\max\{a_i\} = \sum_{S \subseteq \{1,2,\ldots,n\}} (-1)^{|S|-k} \binom{|S|-1}{k-1} \min_{i \in S}\{a_i\}$$

$$k\text{th-}\min\{a_i\} = \sum_{S \subseteq \{1,2,\ldots,n\}} (-1)^{|S|-k} \binom{|S|-1}{k-1} \max_{i \in S}\{a_i\}$$

Problem 5 : Luogu P4707 重返现世

~~没空找~~没找到不涉及概率期望的题。

有 n 种原料。每次随机生成一种，生成 i 的概率为 $\frac{p_i}{m}$。求收集到任意 k 种不同的原料的期望时间。

对 998244353 取模。

$1 \leq n \leq 1000, \max(1, n-10) \leq k \leq n, 1 \leq m \leq 10000, p_i \geq 0, \sum_i p_i = m$。

2s, 128MiB

特殊计数序列

第二类 Stirling 数

$\left\{ {n \atop m} \right\}$ 表示将 n 个两两不同的元素，划分为 m 个互不区分的非空子集的方案数。

第二类 Stirling 数

$\left\{ n \atop m \right\}$ 表示将 n 个两两不同的元素，划分为 m 个互不区分的非空子集的方案数。

递推式 按照最后一个元素所在的子集是否只有一个元素分类。

$$\left\{ n \atop m \right\} = \left\{ n-1 \atop m-1 \right\} + m \left\{ n-1 \atop m \right\}$$

边界：$\left\{ n \atop 0 \right\} = [n = 0]$。

第二类 Stirling 数

$\left\{ n \atop m \right\}$ 表示将 n 个两两不同的元素，划分为 m 个互不区分的非空子集的方案数。

递推式 按照最后一个元素所在的子集是否只有一个元素分类。

$$\left\{ n \atop m \right\} = \left\{ n-1 \atop m-1 \right\} + m\left\{ n-1 \atop m \right\}$$

边界：$\left\{ n \atop 0 \right\} = [n=0]$。

通项公式 对子集非空的条件，使用容斥原理。

$$\left\{ n \atop m \right\} = \frac{1}{m!}\sum_{i=0}^{m}(-1)^i\binom{m}{i}(m-i)^n = \sum_{i=0}^{m}(-1)^{m-i}\frac{i^n}{i!(m-i)!}$$

第二类 Stirling 数

$\left\{ {n \atop m} \right\}$ 表示将 n 个两两不同的元素，划分为 m 个互不区分的非空子集的方案数。

递推式 按照最后一个元素所在的子集是否只有一个元素分类。

$$\left\{ {n \atop m} \right\} = \left\{ {n-1 \atop m-1} \right\} + m\left\{ {n-1 \atop m} \right\}$$

边界：$\left\{ {n \atop 0} \right\} = [n = 0]$。

通项公式 对子集非空的条件，使用容斥原理。

$$\left\{ {n \atop m} \right\} = \frac{1}{m!}\sum_{i=0}^{m}(-1)^i\binom{m}{i}(m-i)^n = \sum_{i=0}^{m}(-1)^{m-i}\frac{i^n}{i!(m-i)!}$$

同一行、同一列的第二类 Stirling 数可以快速计算，暂时不讲。

第一类 Stirling 数

$\begin{bmatrix} n \\ m \end{bmatrix}$ 表示将 n 个两两不同的元素，划分为 m 个互不区分的非空轮换的方案数。

一个轮换是一个首尾相接的环形排列，n 个元素的轮换有 $(n-1)!$ 种。

第一类 Stirling 数

$\begin{bmatrix} n \\ m \end{bmatrix}$ 表示将 n 个两两不同的元素，划分为 m 个互不区分的非空轮换的方案数。

一个轮换是一个首尾相接的环形排列，n 个元素的轮换有 $(n-1)!$ 种。

递推式 按照最后一个元素所在的轮换是否只有一个元素分类。

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

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

降低 DP 复杂度

根据组合意义，有

$$n^k=\sum_{i=0}^{n}\left\{{n \atop i}\right\}i!\binom{n}{i}=\sum_{i=0}^{n}\left\{{n \atop i}\right\}n^{\underline{i}}$$

对于一些问题，转化后 DP 中只有前 size 项非零。

应用

降低 DP 复杂度

根据组合意义，有

$$n^k=\sum_{i=0}^{n}\left\{{n \atop i}\right\}i!\binom{n}{i}=\sum_{i=0}^{n}\left\{{n \atop i}\right\}n^{\underline{i}}$$

对于一些问题，转化后 DP 中只有前 size 项非零。

上升幂、下降幂

$$x^{\overline{n}}=\sum_{m=0}^{n}\left[{n \atop m}\right]x^m$$

$$x^n=\sum_{m=0}^{n}\left\{{n \atop m}\right\}x^{\underline{m}}$$

根据下述的容斥（反演）可以推出普通幂转上升幂，下降幂转普通幂的公式。

用于容斥（反演）

有

$$\sum_{i=m}^{n}(-1)^{n-i}\begin{bmatrix}n\\i\end{bmatrix}\begin{Bmatrix}i\\m\end{Bmatrix}=[n=m]$$

$$\sum_{i=m}^{n}(-1)^{n-i}\begin{Bmatrix}n\\i\end{Bmatrix}\begin{bmatrix}i\\m\end{bmatrix}=[n=m]$$

因此，

$$g_i=\sum_{j=0}^{i}\begin{bmatrix}i\\j\end{bmatrix}f_j\Rightarrow f_i=\sum_{j=0}^{i}(-1)^{i-j}\begin{Bmatrix}i\\j\end{Bmatrix}g_j$$

$$g_i=\sum_{j=0}^{i}\begin{Bmatrix}i\\j\end{Bmatrix}f_j\Rightarrow f_i=\sum_{j=0}^{i}(-1)^{i-j}\begin{bmatrix}i\\j\end{bmatrix}g_j$$

$$g_i=\sum_{j=i}^{n}\begin{bmatrix}j\\i\end{bmatrix}f_j\Rightarrow f_i=\sum_{j=i}^{n}(-1)^{i-j}\begin{Bmatrix}j\\i\end{Bmatrix}g_j$$

$$g_i=\sum_{j=i}^{n}\begin{Bmatrix}j\\i\end{Bmatrix}f_j\Rightarrow f_i=\sum_{j=i}^{n}(-1)^{i-j}\begin{bmatrix}j\\i\end{bmatrix}g_j$$

十二重计数法 IV, VI

IV: 球之间互不相同，盒子全部相同。

$$\sum_{\{i=0\}}^{m} \left\{ {n \atop i} \right\}$$

十二重计数法 IV, VI

IV: 球之间互不相同，盒子全部相同。

$$\sum_{\{i=0\}}^{m} \left\{ {n \atop i} \right\}$$

VI: 球之间互不相同，盒子全部相同，每个盒子至少装一个球。

$$\left\{ {n \atop m} \right\}$$

$$
\begin{aligned}
&\sum_{i=0}^{n}\binom{n}{i} i^k \\
&= \sum_{i=0}^{n}\binom{n}{i} \sum_{j=0}^{k}\left\{{k \atop j}\right\} j!\binom{i}{j} \\
&= \sum_{j=0}^{k}\left\{{k \atop j}\right\} j! \sum_{i=0}^{n}\binom{n}{i}\binom{i}{j} \\
&= \sum_{j=0}^{k}\left\{{k \atop j}\right\} j! \sum_{i=j}^{n}\binom{n}{j}\binom{n-i}{i-j} \\
&= \sum_{j=0}^{k}\left\{{k \atop j}\right\} j! \binom{n}{j} 2^{n-j}
\end{aligned}
$$

对于集合 $\{1, 2, ..., N\}$，求它的子集族中，有多少个满足：

1. 任意两个子集互不相同；
2. $1, 2, ..., N$ 都在其中至少出现了 2 次。

答案对 M 取模。

$2 \leq N \leq 3000, 10^8 \leq M \leq 10^9 + 9$，$M$ 是质数。

4s, 512MiB

分拆数

分拆数 p_n 表示把非负整数 n 的拆成若干个无序的正整数的方法数。

分拆数

分拆数 p_n 表示把非负整数 n 的拆成若干个无序的正整数的方法数。

将 n 分成恰有 k 个部分的分拆，称为 k 部分拆数，记作 $p(n,k)$。

分拆数

分拆数 p_n 表示把非负整数 n 的拆成若干个无序的正整数的方法数。

将 n 分成恰有 k 个部分的分拆，称为 k 部分拆数，记作 $p(n,k)$。

按照最小的一部分是否是 1 分类，可得 $p(n,k) = p(n-1,k-1) + p(n-k,k)$，据此可以 $\mathcal{O}(n^2)$ 计算 $p(n,k)$ 及 p_n。

分拆数

分拆数 p_n 表示把非负整数 n 的拆成若干个无序的正整数的方法数。

将 n 分成恰有 k 个部分的分拆，称为 k 部分拆数，记作 $p(n,k)$。

按照最小的一部分是否是 1 分类，可得 $p(n,k)=p(n-1,k-1)+p(n-k,k)$，据此可以 $\mathcal{O}(n^2)$ 计算 $p(n,k)$ 及 p_n。

一个 $\mathcal{O}(n\sqrt{n})$ 计算 $p_1,p_2,...,p_n$ 的方法

分拆数

分拆数 p_n 表示把非负整数 n 的拆成若干个无序的正整数的方法数。

将 n 分成恰有 k 个部分的分拆，称为 k 部分拆数，记作 $p(n,k)$。

按照最小的一部分是否是 1 分类，可得 $p(n,k) = p(n-1,k-1) + p(n-k,k)$，据此可以 $\mathcal{O}(n^2)$ 计算 $p(n,k)$ 及 p_n。

一个 $\mathcal{O}(n\sqrt{n})$ 计算 $p_1, p_2, \ldots, p_n$ 的方法

用五边形数定理也可以 $\mathcal{O}(n\sqrt{n})$ 计算 $p_1, p_2, \ldots, p_n$。

分拆数

分拆数 p_n 表示把非负整数 n 的拆成若干个无序的正整数的方法数。

将 n 分成恰有 k 个部分的分拆，称为 k 部分拆数，记作 $p(n,k)$。

按照最小的一部分是否是 1 分类，可得 $p(n,k)=p(n-1,k-1)+p(n-k,k)$，据此可以 $\mathcal{O}(n^2)$ 计算 $p(n,k)$ 及 p_n。

一个 $\mathcal{O}(n\sqrt{n})$ 计算 $p_1,p_2,...,p_n$ 的方法

用五边形数定理也可以 $\mathcal{O}(n\sqrt{n})$ 计算 $p_1,p_2,...,p_n$。

分拆数增长不快，$p_{19}\le 500, p_{32}\le 10^4, p_{94}\le 10^8$。

五边形数定理

$$\prod_{i=1}^{\infty}(1-x^i)=\sum_{i=-\infty}^{\infty}(-1)^i x^{\frac{i(3i-1)}{2}}$$

可以通过构造互异分拆数之间的映射证明。

Problem 7 : SOJ1377

求长度为 n 的排列中，满足前 j 个数的逆序对数恰好为 v 的排列的个数。

答案对 2 取模。

T 组数据，$T \leq 1000, 2 \leq n \leq 10^{10}, 0 \leq v \leq n, 1 \leq j \leq n$。

$1\text{s}, 512\text{MiB}$

格点路径计数

Catalan 数

Catalan 数 C_n 等于

- $\frac{\binom{2n}{n}}{n+1} = \frac{(2n)!}{n!(n+1)!} = \binom{2n}{n} - \binom{2n}{n-1}$

Catalan 数

Catalan 数 C_n 等于

- $\frac{\binom{2n}{n}}{n+1} = \frac{(2n)!}{n!(n+1)!} = \binom{2n}{n} - \binom{2n}{n-1}$
- 长为 $2n$ 的合法括号序列数量。

Catalan 数

Catalan 数 C_n 等于

- $\frac{\binom{2n}{n}}{n+1} = \frac{(2n)!}{n!(n+1)!} = \binom{2n}{n} - \binom{2n}{n-1}$
- 长为 $2n$ 的合法括号序列数量。
- 把 n 个 $+1$ 和 n 个 -1 排成长度为 $2n$ 的序列，使得前缀和非负的方案数。

Catalan 数

Catalan 数 C_n 等于

- $\frac{\binom{2n}{n}}{n+1} = \frac{(2n)!}{n!(n+1)!} = \binom{2n}{n} - \binom{2n}{n-1}$
- 长为 $2n$ 的合法括号序列数量。
- 把 n 个 $+1$ 和 n 个 -1 排成长度为 $2n$ 的序列，使得前缀和非负的方案数。
- 在平面上，从 $(0,0)$ 走到 (n,n)，每次只能向右或上走 1，不经过直线 $y=x$ 上方的方案数。

Catalan 数

Catalan 数 C_n 等于

- $\frac{\binom{2n}{n}}{n+1} = \frac{(2n)!}{n!(n+1)!} = \binom{2n}{n} - \binom{2n}{n-1}$
- 长为 $2n$ 的合法括号序列数量。
- 把 n 个 $+1$ 和 n 个 -1 排成长度为 $2n$ 的序列，使得前缀和非负的方案数。
- 在平面上，从 $(0,0)$ 走到 (n,n)，每次只能向右或上走 1，不经过直线 $y=x$ 上方的方案数。
- 圆上有 $2n$ 个点，将这些点成对连接起来使得所得到的 n 条不相交的线段的方案数。

Catalan 数

Catalan 数 C_n 等于

- $\frac{\binom{2n}{n}}{n+1} = \frac{(2n)!}{n!(n+1)!} = \binom{2n}{n} - \binom{2n}{n-1}$
- 长为 $2n$ 的合法括号序列数量。
- 把 n 个 $+1$ 和 n 个 -1 排成长度为 $2n$ 的序列，使得前缀和非负的方案数。
- 在平面上，从 $(0,0)$ 走到 (n,n)，每次只能向右或上走 1，不经过直线 $y=x$ 上方的方案数。
- 圆上有 $2n$ 个点，将这些点成对连接起来使得所得到的 n 条不相交的线段的方案数。
- 记儿子顺序但无标号的 $n+1$ 个节点的有根树数量。

Catalan 数

Catalan 数 C_n 等于

- $\frac{\binom{2n}{n}}{n+1} = \frac{(2n)!}{n!(n+1)!} = \binom{2n}{n} - \binom{2n}{n-1}$
- 长为 $2n$ 的合法括号序列数量。
- 把 n 个 $+1$ 和 n 个 -1 排成长度为 $2n$ 的序列，使得前缀和非负的方案数。
- 在平面上，从 $(0,0)$ 走到 (n,n)，每次只能向右或上走 1，不经过直线 $y=x$ 上方的方案数。
- 圆上有 $2n$ 个点，将这些点成对连接起来使得所得到的 n 条不相交的线段的方案数。
- 记儿子顺序但无标号的 $n+1$ 个节点的有根树数量。

递推式：$C_n = \sum_{i=1}^{n} C_{i-1} C_{n-i} \quad (n \geq 1)$。

证明

考虑在平面上，从 $(0,0)$ 走到 (n,n)，每次只能向右或上走 1，不与 $y=x+1$ 相交的方案数。总路径数为 $\binom{2n}{n}$。

对于一条不合法的路径，将其与 $y=x+1$ 的第一个交点之后的路径翻转（上变成右，右变成上），最终会走到 $(n-1,n+1)$。

每一条不合法的路径都唯一地对应一条从 $(0,0)$ 走到 $(n-1,n+1)$ 的路径。因此不合法的路径数为 $\binom{2n}{n-1}$。

合法路径数即为 $\binom{2n}{n}-\binom{2n}{n-1}$。

考虑长为 $2n$ 的合法括号序列数量。

枚举第一个左括号与哪个右括号匹配。

若其与第 $2i$ 个位置上的右括号匹配，则这对括号之间的合法括号序列数量为 C_{i-1}，之后的合法括号序列数量为 C_{n-i}。

因此，有 $C_n = \sum_{i=1}^{n} C_{i-1}C_{n-i} \ \ (n \geq 1)$。

考虑长为 $2n$ 的合法括号序列数量。

枚举第一个左括号与哪个右括号匹配。

若其与第 $2i$ 个位置上的右括号匹配，则这对括号之间的合法括号序列数量为 C_{i-1}，之后的合法括号序列数量为 C_{n-i}。

因此，有 $C_n = \sum_{i=1}^{n} C_{i-1} C_{n-i} \ \ (n \geq 1)$。

使用后面提到的 (n, m)-Dyck 路计数。

C_n 等于 $(n+1, n)$-Dyck 路径数量。

斜率为 1 的直线

求从 $(0,0)$ 走到 (n,m)，每次只能向右或上走 1，不与 $y=x+t$ 相交的方案数 $(t>0, m<n+t)$。

斜率为 1 的直线

求从 $(0,0)$ 走到 (n,m)，每次只能向右或上走 1，不与 $y=x+t$ 相交的方案数 $(t>0, m<n+t)$。

总路径数为 $\binom{n+m}{n}$。

对于一条不合法的路径，将其与 $y=x+t$ 的第一个交点之后的路径翻转，最终会走到 $(m-t, n+t)$。

每一条不合法的路径都唯一地对应一条从 $(0,0)$ 走到 $(m-t,n+t)$ 的路径。因此不合法的路径数为 $\binom{n+m}{n+t}$。

合法路径数即为 $\binom{n+m}{n}-\binom{n+m}{n+t}$。

斜率为 1 的直线

求从 $(0,0)$ 走到 (n,m)，每次只能向右或上走 1，不与 $y=x+t$ 相交的方案数 $(t>0, m<n+t)$。

总路径数为 $\binom{n+m}{n}$。

对于一条不合法的路径，将其与 $y=x+t$ 的第一个交点之后的路径翻转，最终会走到 $(m-t, n+t)$。

每一条不合法的路径都唯一地对应一条从 $(0,0)$ 走到 $(m-t, n+t)$ 的路径。因此不合法的路径数为 $\binom{n+m}{n+t}$。

合法路径数即为 $\binom{n+m}{n} - \binom{n+m}{n+t}$。

思考：$m>n+t$ 时为什么不对？

斜率为 1 的直线

求从 $(0,0)$ 走到 (n,m)，每次只能向右或上走 1，不与 $y=x+t$ 相交的方案数 $(t>0, m<n+t)$。

总路径数为 $\binom{n+m}{n}$。

对于一条不合法的路径，将其与 $y=x+t$ 的第一个交点之后的路径翻转，最终会走到 $(m-t, n+t)$。

每一条不合法的路径都唯一地对应一条从 $(0,0)$ 走到 $(m-t,n+t)$ 的路径。因此不合法的路径数为 $\binom{n+m}{n+t}$。

合法路径数即为 $\binom{n+m}{n}-\binom{n+m}{n+t}$。

思考：$m>n+t$ 时为什么不对？

一条走到 $(m-t,n+t)$ 的路径不一定与 $y=x+t$ 相交。

两条斜率为 1 的直线

求从 $(0,0)$ 走到 (n,m)，每次只能向右或上走 1，不与 $y=x+l, y=x+r$ 相交的方案数 $(l<0, r>0, n+l<m<n+r)$。

两条斜率为 1 的直线

求从 $(0,0)$ 走到 (n,m)，每次只能向右或上走 1，不与 $y=x+l, y=x+r$ 相交的方案数 $(l<0, r>0, n+l<m<n+r)$。

把条件改成不与 $y=x+l+(r-l)t, t\in\mathbb{Z}$ 相交。

对于每个交点，可以选择翻转或者不翻转，翻转后乘上 -1 的系数。（可以发现，翻转不影响之后的交点。）

也就是说，一个有 k 个交点的路径，对应 2^k 种“翻转方案”，这些“翻转方案”的带权和为 $(1-1)^k$。

所以，所有“翻转方案”的带权和就是答案。

两条斜率为 1 的直线

求从 $(0,0)$ 走到 (n,m)，每次只能向右或上走 1，不与 $y=x+l, y=x+r$ 相交的方案数 $(l<0, r>0, n+l<m<n+r)$。

把条件改成不与 $y=x+l+(r-l)t, t\in\mathbb{Z}$ 相交。

对于每个交点，可以选择翻转或者不翻转，翻转后乘上 -1 的系数。(可以发现，翻转不影响之后的交点。)

也就是说，一个有 k 个交点的路径，对应 2^k 种“翻转方案”，这些“翻转方案”的带权和为 $(1-1)^k$。

所以，所有“翻转方案”的带权和就是答案。

显然，翻转偶数次的“翻转方案”权值为 1，在 $(n+(r-l)t, m-(r-l)t), t\in\mathbb{Z}$;
翻转奇数次的“翻转方案”权值为 -1，在 $(m-r+(r-l)t, n+r-(r-l)t), t\in\mathbb{Z}$。

求和，可得答案为 $\sum_{k\in\mathbb{Z}}\binom{n+m}{n+(r-l)k}-\sum_{k\in\mathbb{Z}}\binom{n+m}{n+r+(r-l)k}$。

若一条从 $(0, 0)$ 到 (n, m) 的格路，始终不经过直线 $y = \frac{m}{n}x$ 上方，则称之为一条 (n, m)-Dyck 路。

(n, m)-Dyck 路

若一条从 $(0,0)$ 到 (n,m) 的格路，始终不经过直线 $y = \frac{m}{n}x$ 上方，则称之为一条 (n,m)-Dyck 路。

若 $(n,m) = 1$，则 (n,m)-Dyck 路的数量为 $\frac{1}{n+m}\binom{n+m}{n}$。

与直线相交恰好 t 次

恰好 t 次可以转化成不少于 t 次。

直线斜率为 1

求从 $(0,0)$ 走到 (n,m) 恰与 $y=x+b$ 相交不少于 t 次的路径数量 $(0\le b\le m-n)$。

与直线相交恰好 t 次

恰好 t 次可以转化成不少于 t 次。

直线斜率为 1

求从 $(0,0)$ 走到 (n,m) 恰与 $y=x+b$ 相交不少于 t 次的路径数量 $(0\le b\le m-n)$。

把第 t 次相交前的部分全部翻折到 $y=x+b$ 的下方，再将前 $t-1$ 个交点之后的向右 1 删去。

容易验证每个走到 $(n-t+1,m)$ 的路径恰好对应 2^{t-1} 个合法路径。

答案为 $2^{t-1}\binom{n+m-t+1}{m}$。

与直线相交恰好 t 次

恰好 t 次可以转化成不少于 t 次。

直线斜率为 1

求从 $(0,0)$ 走到 (n,m) 恰与 $y=x+b$ 相交不少于 t 次的路径数量 $(0\le b\le m-n)$。

把第 t 次相交前的部分全部翻折到 $y=x+b$ 的下方，再将前 $t-1$ 个交点之后的向右 1 删去。

容易验证每个走到 $(n-t+1,m)$ 的路径恰好对应 2^{t-1} 个合法路径。

答案为 $2^{t-1}\binom{n+m-t+1}{m}$。

直线斜率为正整数 (ZR2848)

求从 $(0,0)$ 走到 (n,m) 恰与 $y=kx+b$ 相交不少于 t 次的路径数量 $(k\ge 1, b\ge 0, 0\le kn+b\le m)$。

恰好 t 次可以转化成不少于 t 次。

直线斜率为 1

求从 $(0,0)$ 走到 (n,m) 恰与 $y=x+b$ 相交不少于 t 次的路径数量 $(0\le b\le m-n)$。

把第 t 次相交前的部分全部翻折到 $y=x+b$ 的下方，再将前 $t-1$ 个交点之后的向右 1 删去。

容易验证每个走到 $(n-t+1,m)$ 的路径恰好对应 2^{t-1} 个合法路径。

答案为 $2^{t-1}\binom{n+m-t+1}{m}$。

直线斜率为正整数 (ZR2848)

求从 $(0,0)$ 走到 (n,m) 恰与 $y=kx+b$ 相交不少于 t 次的路径数量 $(k\ge 1, b\ge 0, 0\le kn+b\le m)$。

过程略去，见该题题解。答案为 $(k+1)^{t-1}\binom{n+m-t+1}{m}$。

Problem 8 : Codeforces 1770G. Koxia and Bracket

给定一个括号序列 s，你需要删除**尽可能少**的字符使得操作完的序列是个合法括号序列。

请求出所有最优删除方案的数量，答案对 998244353 取模。

$1 \leq |s| \leq 5 \times 10^5$。

5s, 256MiB

线性代数相关

LGV 引理

在有向无环图 G 上，有 n 个起点 $a_1, ..., a_n$ 和 n 个终点 $b_1, ..., b_n$，每条边有边权。

定义一条路径的权值为该路径上所有边的权值的乘积，即 $\omega(P) = \prod_{e \in P} \omega_e$。

定义 $e(a, b)$ 为所有 a 到 b 的路径的权值之和 $e(a, b) = \sum_{P: a \to b} \omega(P)$。

有

$$M = \begin{pmatrix} e(a_1, b_1) & e(a_1, b_2) & \dots & e(a_1, b_n) \\ e(a_2, b_1) & e(a_2, b_2) & \dots & e(a_2, b_n) \\ \vdots & \vdots & \ddots & \vdots \\ e(a_n, b_1) & e(a_n, b_2) & \dots & e(a_n, b_n) \end{pmatrix}$$

$$\det(M) = \sum_{S: A \to B} (-1)^{\mathrm{inv}(\sigma(S))} \prod_{i=1}^{n} \omega(S_i)$$

其中 S 是一组不相交路径，S_i 是一条从 a_i 到 $b_{\sigma(S)_i}$ 的路径，inv 表示逆序数。

Problem 9 : Codeforces 348D. Turtles

给定一个 $n \times m$ 的网格，上面有 `.` 和 #， `.` 表示可以走，# 表示障碍。

有两只乌龟要从 $(1,1)$ 走到 (n,m)，乌龟每次都可以向下或者向右走一格， 要求两只乌龟走的路径除了 $(1,1)$ 和 (n,m) 以外不交，求方案数。

对 $10^9 + 7$ 取模。

$2 \le n, m \le 3000$。

2s, 256MiB

矩阵树定理：无向图

设 G 是一个有 n 个顶点的无向图。定义度数矩阵 $D(G)$ 为：

$$D_{i,i}(G) = \deg(i), D_{i,j} = 0, i \neq j$$

设 $\#e(i,j)$ 为点 i 与点 j 相连的边数，并定义邻接矩阵 A 为：

$$A_{i,j}(G) = A_{j,i}(G) = \#e(i,j), i \neq j$$

定义 Laplace 矩阵（亦称 Kirchhoff 矩阵）L 为：

$$L(G) = D(G) - A(G)$$

记图 G 的所有生成树个数为 $t(G)$。

$$t(G) = \det L(G) \begin{pmatrix} 1, 2, ..., i-1, i+1, ..., n \\ 1, 2, ..., i-1, i+1, ..., n \end{pmatrix}$$

其中记号 $\det L(G)\binom{1,2,...,i-1,i+1,...,n}{1,2,...,i-1,i+1,...,n}$ 表示矩阵 $L(G)$ 的第 $1, ..., i-1, i+1, ..., n$ 行与第 $1, ..., i-1, i+1, ..., n$ 列构成的子矩阵。也就是说，无向图的 Laplace 矩阵的所有 $n-1$ 阶主子式都相等。

矩阵树定理：有向图

设 G 是一个有 n 个顶点的有向图。定义出度矩阵 $D^{\mathrm{out}}(G)$ 为：

$$D^{\mathrm{out}}_{i,i}(G) = \deg^{\mathrm{out}}(i), D_{i,j} = 0, i \neq j$$

类似地定义入度矩阵 $D^{\mathrm{in}}(G)$。
设 $\#e(i,j)$ 为点 i 指向点 j 的有向边数，并定义邻接矩阵 A 为：

$$A_{i,j}(G) = A_{j,i}(G) = \#e(i,j), i \neq j$$

定义 Laplace 矩阵 $L^{\mathrm{out}}, L^{\mathrm{in}}$ 为：

$$L^{\mathrm{out}}(G) = D^{\mathrm{out}}(G) - A(G)$$

$$L^{\mathrm{in}}(G) = D^{\mathrm{in}}(G) - A(G)$$

记图 G 的所有以 r 为根的根向树形图（边全部指向父亲）个数为 $t^{\mathrm{root}}(G,r)$，
记图 G 的所有以 r 为根的叶向树形图（边全部指向儿子）个数为 $t^{\mathrm{leaf}}(G,r)$。

$$t^{\mathrm{root}}(G,k) = \det L^{\mathrm{out}}(G)\begin{pmatrix} 1,2,...,k-1,k+1,...,n \\ 1,2,...,k-1,k+1,...,n \end{pmatrix}$$

$$t^{\mathrm{leaf}}(G,k) = \det L^{\mathrm{in}}(G)\begin{pmatrix} 1,2,...,k-1,k+1,...,n \\ 1,2,...,k-1,k+1,...,n \end{pmatrix}$$

Problem 10 : Codeforces 917D. Stranger Trees

给定一张 n（$2 \le n \le 100$）个节点的无向完全图和这个图的一棵生成树。对于 $i = 0, 1, ..., n-1$，求出有多少棵这个完全图的生成树，使得这些生成树与给定的生成树恰好有 i 条边重复。

答案对 $(10^9 + 7)$ 取模。

1s, 256MiB

BEST 定理

设 G 是有向欧拉图，那么 G 的不同欧拉回路总数 $\mathrm{ec}(G)$ 是

$$\mathrm{ec}(G) = t^{\mathrm{root}}(G, k) \prod_{v \in V} (\deg(v) - 1)!$$

对欧拉图 G 的任意两个节点 i, j，都有 $t^{\mathrm{root}}(G, i) = t^{\mathrm{root}}(G, j)$，且欧拉图 G 的所有节点的入度和出度相等。

Problem 11 : AGC051D - C4

有一张 4 个点 4 条边的简单无向连通图，点的编号分别为 $1,2,3,4$ ，边分别连接着 $e_1:(1,2), e_2:(2,3), e_3:(3,4), e_4:(4,1)$。

给定 4 个数 v_1, v_2, v_3, v_4 求满足以下条件的路径数量：

从 1 号点出发并到 1 号点结束，且经过第 i 条边 e_i 恰好 v_i 次。

你需要输出路径数对 998244353 取模的结果。

$1 \leq v_1, v_2, v_3, v_4 \leq 5 \times 10^5$

2s, 1024MiB

杂题

Problem 12 : AGC023E Inversions

Problem 13 : ARC118E Avoid Permutations

Problem 14 : 【THUPC 2024 决赛】排列游戏

Problem 15 : 【统一省选 2024】重塑时光

Problem 16 : 【UR #27】509 号迷宫

Problem 17 : 【NOI2023】桂花树

Problem 18 : 【NOI2021】路径交点

Problem 19 : 【2021 集训队互测 Round 2】Imbalance

Problem 20 : 【2023 集训队互测 Round 3】Permutation Counting 2

Problem 21 : Codeforces 1784E. Infinite Game

Problem 22 : Codeforces 1784D. Wooden Spoon

Problem 23 : BZOJ4671 异或图

Problem 24 : ARC124E - Pass to Next

Problem 25 : ARC162E - Strange Constraints

Problem 26 : AGC061C - First Come First Serve

Problem 27 : ARC178D - Delete Range Mex

Problem 28 : AGC056B - Range Argmax

Problem 29 : Codeforces 1809G. Prediction

Problem 30 : Codeforces 1747E. List Generation

Problem 31 : Codeforces 1605F. PalindORme

Problem 32 : Codeforces 1750F. Majority

Thanks

just#remember19

the#life#you#want24