1.7 组合数学

- n 个无标号小球放入 m 个无标号盒子中，设答案为 $f_{n,m}$，转移即讨论第一个盒子是否放球：

$$f_{0,i}=1, f_{i,j}=f_{i,j-1}+f_{i-j,j}$$

- 若要求每个盒子中至少放一个，则答案为 $f_{n-m,m}$，枚举 m 求和即 **分拆数。**
- **结论1** $\sum\limits_{i=0}^{m}\binom{k+i}{k}=\binom{k+m+1}{k+1}$

> $$\begin{aligned}\sum_{i=0}^{m}\binom{k+i}{k}&=\binom{k+1}{k+1}+\binom{k+1}{k}+\sum_{i=2}^{m}\binom{k+i}{k}\\&=\binom{k+2}{k+1}+\binom{k+2}{k}+\sum_{i=3}^{m}\binom{k+i}{k}\\&=\ldots\\&=\binom{k+m+1}{k+1}\end{aligned}$$

- **结论2** 对于斐波拉契数列 $F_n(F_0=0,F_1=1,F_n=F_{n-1}+F_{n-2})$， 有：

$$F_n=\sum_{k=0}^{n-1}\binom{n-1-k}{k}$$

证明 易知 $F_0=0,F_1=1$，对于 $n\geq 2$，考虑归纳证明

$$\begin{aligned}F_n&=F_{n-1}+F_{n-2}\\&=\binom{n-2}{0}+\sum_{k=1}^{n-2}\binom{n-2-k}{k}+\sum_{k=1}^{n-2}\binom{n-2-k}{k-1}\\&=\binom{n-1}{0}+\sum_{k=1}^{n-2}\binom{n-1-k}{k}+\binom{0}{n-1}\\&=\sum_{k=0}^{n-1}\binom{n-1-k}{k}\end{aligned}$$

- **范德蒙德卷积** $\sum_{i=0}^{k}\binom{n}{i}\binom{m}{k-i}=\binom{n+m}{k}$
  - **推论1** $\sum_{i=0}^{\min\{n,m\}}\binom{n}{i}\binom{m}{i}=\binom{n+m}{n}=\binom{n+m}{m}$（利用 $\binom{n}{i}=\binom{n}{n-i}$）。
  - **推论2** $\sum_{i=0}^{n}\binom{n}{i}\binom{m}{i+k}=\sum_{i=0}^{n}\binom{n}{n-i}\binom{m}{i+k}=\binom{n+m}{n+k}$

鸽巢原理

- 将 $n+1$ 个物体，划分为 n 组，至少有一组有两个或以上的物体。
- 将 n 个物体，划分为 k 组，至少存在一个分组，含有至少 $\lceil\frac{n}{k}\rceil$ 个物体。

典例 2022牛客多校10I

题目大意

- 给定数组 a,b，长度分别为 n,m，询问能够找到一组 i,j,k,l，满足 $i\neq j,k\neq l,|a_i-a_j|=|b_k-b_l|$。
- $2\leq n,m\leq 10^6,0\leq a_i,b_i\leq 10^7$。

题解

- 若 a,b 均有重复元素，显然有解，否则将 a,b 排序后去重。
- 原式去掉绝对值后移项可变为 $a_i+b_l=a_j+b_k$，由鸽巢原理，a_j+b_k 的值两两不同的情况最多只有 2×10^7 种，暴力枚举即可。

容斥原理

- 若是带着容斥系数 DP，可利用组合数递推关系进行转移。

一般化形式

- 对于两个集合函数 $f(S),g(S)$，以下两个等式可进行等价变换：

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

证明 以下证明左边推导到右边，右边到左边证明过程基本一致。

$$\begin{aligned}\sum_{T\subseteq S}(-1)^{|S|-|T|}f(T)&=\sum_{T\subseteq S}\sum_{Q\subseteq T}(-1)^{|S|-|T|}g(Q)\\&=\sum_{Q\subseteq S}g(Q)\sum_{Q\subseteq T\subseteq S}(-1)^{|S|-|T|}\\&=\sum_{Q\subseteq S}g(Q)\sum_{Q\subseteq T\subseteq S,T'=T-Q\subseteq S-Q}(-1)^{|S-Q|-|T'|}\end{aligned}$$

> 另设
>
> $$\begin{aligned} F(S) &= \sum_{T\subseteq S}(-1)^{|S|-|T|} \\ &= \sum_{i=0}^{|S|}(-1)^{|S|-i}\binom{|S|}{i} \\ &= (1-1)^{|S|} = [S=\varnothing] \end{aligned}$$
>
> 代入得:
>
> $$\begin{aligned} g(S) &= \sum_{Q\subseteq S} g(Q)F(S-Q) \\ &= \sum_{Q\subseteq S} g(Q)[S=Q] = g(S) \end{aligned}$$

- 上述证明过程会得到以下等式，也可以作为一种变换技巧:

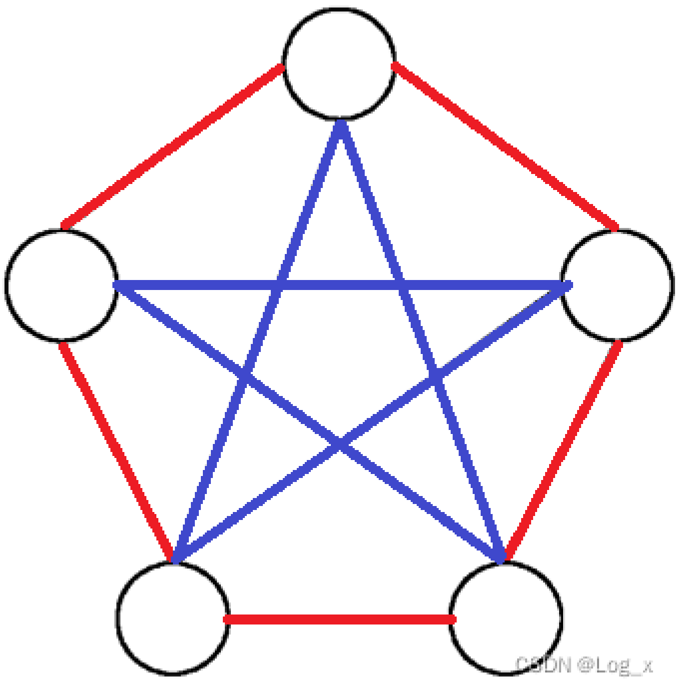
$$f(S)=\sum_{T\subseteq S}\sum_{Q\subseteq T}(-1)^{|T|-|Q|}f(Q)=\sum_{T\subseteq S}(-1)^{|S|-|T|}\sum_{Q\subseteq T}f(Q)$$

- 取 $f'(S)=f(U-S),g'(S)=g(U-S)$，可有以下推论:

$$f'(S)=\sum_{S\subseteq T}g'(T)\Leftrightarrow g'(S)=\sum_{S\subseteq T}(-1)^{|T|-|S|}f'(T)$$

Ramsey 数

- 定义 $r(m,n)$ 表示满足以下条件的完全图的最少点数 n:
  - 将图的所有边染为红蓝两种颜色之一。
  - 无论如何染色，图中至少存在一个边全为红色的 K_m 或者边全为蓝色的 K_n。
- 由定义可知 $r(n,m)=r(m,n),r(1,n)=1$。
- 利用鸽巢原理容易证明 $r(3,3)\le 6$，$r(3,3)>5$ 的反例如下图，实际上任取一个五元环即可。



- 可以证明 $r(2,n)=n$:
  - 证明 $r(2,n)>n-1$，显然我们可以将 K_{n-1} 的所有边都染成蓝色而符合条件。
  - 证明 $r(2,n)\le n$，显然 K_n 所有边全为蓝色和其中至少存在一个边全为红色的 K_2 ，两个条件总是至少有一个成立。

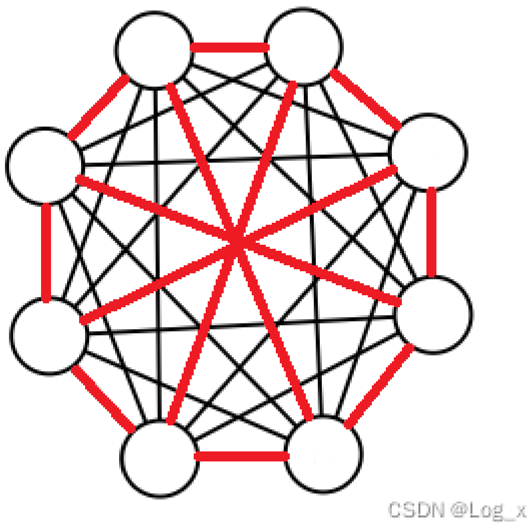
- **结论1** $r(n,m) \le r(n-1,m)+r(n,m-1)$

> **证明** 设 $p=r(n-1,m)+r(n,m-1)$，任取 p 个点中的一个点 x，设 R_x 表示与 x 相连的边均为红色的点集，B_x 表示与 x 相连的边均为蓝色的点集，易知 $|R_x|+|B_x|=p-1$。
>
> 由鸽巢原理，一定存在以下两个条件之一成立：
>
> 1. $|R_x| \ge r(n-1,m)$
> 2. $|B_x| \ge r(n,m-1)$
>
> 对于条件 1，在 R_x 中任取 $r(n-1,m)$ 个点，由 $r(n-1,m)$ 的定义至少存在一个边全为红色的 K_{n-1} 或边全为蓝色的 K_m。若存在一个边全为蓝色的 K_m，证明已经完成；若存在一个边全为红色的 K_{n-1}，将 x 加入即可得到边全为红色的 K_n。
>
> 对于条件2，同理可证。

- **推论1** 若 $r(n-1,m)$ 和 $r(n,m-1)$ 均为偶数，则 $r(n,m) \le r(n-1,m)+r(n,m-1)-1$

> **证明** 沿用 **结论1** 中证明的定义，此时 $|R_x|+|B_x|=p-2$。
>
> 由鸽巢定理，一定存在以下三个条件之一成立：
>
> 1. $|R_x| \ge r(n-1,m)$
> 2. $|B_x| \ge r(n,m-1)$
> 3. $|R_x|=r(n-1,m)-1$ 且 $|B_x|=r(n,m-1)-1$
>
> 前两个条件已经被证明，该推论可能不成立当且仅当对于所有点都满足条件 3，然而若所有点都满足条件 3 时由握手定理，图中的红边总数为 $\frac{(r(n-1,m)-1)(p-1)}{2}$，由奇偶性该式一定不为整数，故至少存在一个点不满足条件 3。

- 部分 Ramsey 数的证明：
  - $r(3,4)=9$
    - 由 **推论1** 可知 $r(3,4) \le r(2,4)+r(3,3)-1 \le 4+6-1=9$，以下是证明 $r(3,4)>8$ 的例子：



    - 记中间四条红边为特殊边。对于边全为红色的 K_3，容易验证不含或只含一条特殊边的红环边数均大于3，而若含两条及以上特殊边，因为特殊边间没有公共点，肯定不会形成三元环。对于边全为蓝色的 K_4，我们需要保证选到的点中不存在两点是一条特殊边的两个端点，但此时一定会存在两点在除特殊边外的红八元环上相邻。
  - $r(3,5)=14$
    - 由 **推论** 可知 $r(3,5) \le r(2,5)+r(3,4)=5+9=14$，下面是证明 $r(3,5)>13$ 的例子：

        - 构造 13 个结点，标号 $0 \sim 12$。
        - 设 $A=\{1,5,8,12\}$，(i,j) 为红色当且仅当 $j-i \in A$。
    - 首先肯定不会边全为红色的 K_3，因为集合 A 中任意两个元素之和均不属于 A。
    - 注意到该构造的定义实际上是对称的，将一个 13 元环所有距离属于集合 A 的点对相连，故若存在至少一个边全为蓝色的 K_5，一定存在一个边全为蓝色的 K_5 包含结点 0。
    - 对剩下选取的结点进行分类讨论，不难排除所有可能的方案。
- **结论2** $r(n,m) > (n-1)(m-1)$

> **证明** 构造一个 $n-1$ 行 $m-1$ 列的点阵，同行的点之间连红边，否则连蓝边，显然图中不存在边全为红色的 K_{n-1} 和边全为蓝色的 K_{m-1}。

- 将 Ramsey 数的概念推广到 k 染色，不难得到 $r(x_1,x_2,\dots,x_k)$ 的定义。
- **结论3** $r(x_1,x_2,\dots,x_k) \le r(x_1,x_2,\dots,x_{k-2},r(x_{k-1},x_k))$

> **证明** 根据定义讨论即可。

- **Sum-Free 划分** 将集合 $\{1,2,\dots,n\}$ 划分成 k 个集合，使得每个集合中不存在两个数 x,y（x 和 y 可相等），使得 $x+y$ 也在这一集合中。
- **结论4** 设 s_k 表示关于参数 k 最小的不能进行 **Sum-Free 划分** 的 n，则 $s_k < r(3,\dots,3)(k\uparrow 3)$

> **证明** 设 $X=r(3,\dots,3)(k\uparrow 3)$，将 $\{1,2,\dots,X-1\}$ 划分成 k 个集合 $R_1,R_2,\dots,R_k$， 对于任意结点编号 $i,j \in \{1,2,\dots,X\}$，若 $|i-j| \in R_l$，则将 (i,j) 染为颜色 l，由 X 的定义图中至少存在一个边同色的 K_3，设编号为 $i<j<k$，则有 $|j-i|+|k-j|=|k-i|$，与 **Sum-Free 划分** 的定义矛盾。

二项式反演

- **结论** 给定 $k \in \mathbb{N}$，则存在以下关系式

$$g_k=\sum_{i=k}^{n}\binom{i}{k}f_i \Leftrightarrow f_k=\sum_{i=k}^{n}(-1)^{i-k}\binom{i}{k}g_i$$

> **证明**
>
> $$\begin{aligned}\sum_{i=k}^{n}(-1)^{i-k}\binom{i}{k}g_i &= \sum_{i=k}^{n}(-1)^{i-k}\binom{i}{k}\sum_{j=i}^{n}\binom{j}{i}f_j \\ &= \sum_{j=k}^{n}f_j\sum_{i=k}^{j}(-1)^{i-k}\binom{i}{k}\binom{j}{i} \\ &= \sum_{j=k}^{n}f_j\sum_{i=k}^{j}(-1)^{i-k}\binom{j}{k}\binom{j-k}{i-k} \\ &= \sum_{j=k}^{n}\binom{j}{k}f_j\sum_{i=0}^{j-k}(-1)^{i}\binom{j-k}{i} \\ &= \sum_{j=k}^{n}\binom{j}{k}f_j(1-1)^{j-k} \\ &= \sum_{j=k}^{n}\binom{j}{k}f_j[j=k]=f_k\end{aligned}$$

第二类斯特林数

- 记作 $S(n,k)$ 或者 $\left\{\begin{matrix}n\\k\end{matrix}\right\}$ 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。

- 易知 $\begin{Bmatrix} n \\ 0 \end{Bmatrix} = [n=0], \begin{Bmatrix} n \\ k \end{Bmatrix} = \begin{Bmatrix} n-1 \\ k-1 \end{Bmatrix} + k\begin{Bmatrix} n-1 \\ k \end{Bmatrix}$。
- **结论** 长度为 n 且最大数不超过 m 的单调不降正整数数列的各数乘积之和为 $\begin{Bmatrix} n+m \\ m \end{Bmatrix}$。

> **证明** 相当于将 $n+m$ 个不同的球放入 m 个相同的盒子中，每个球至少有一个球。可将放入每个球的过程分为选择一个有球/无球的盒子和放入两步，则需要选择一个有球盒子的球恰好有 n 个，第 i 个选择有球盒子的球选择的方案数为当前有球的盒子数， 可等价为数列中的第 i 个数。

快速计算一整行的第二类斯特林数

- 设 $G_{n,k}$ 为将 n 个两两不同的元素，划分为 k 个两两不同（允许为空）的集合的方案数，$F_{n,k}$ 为将 n 个两两不同的元素，划分为 k 个两两不同的非空集合的方案数，由容斥原理得：

$$\begin{Bmatrix} n \\ k \end{Bmatrix} = \frac{F_{n,k}}{k!} = \frac{1}{k!}\sum_{i=0}^{k}(-1)^{k-i}\binom{k}{i}G_{n,i} = \sum_{i=0}^{k}\frac{(-1)^{k-i}}{(k-i)!}\frac{i^n}{i!}$$

- 显然是卷积的形式。

快速计算一整列的第二类斯特林数

- 由于非空集合的限制，设指数型生成函数 $H(x) = \sum\limits_{i\geq 1}\frac{x^i}{i!}$，则

$$\begin{Bmatrix} i \\ k \end{Bmatrix} = \frac{i!}{k!}[x^i]H^k(x)$$

- 通过多项式快速幂计算即可。

次幂的拆解

- 因为

$$G_{n,k} = \sum_{i=0}^{k}\binom{k}{i}F_{n,i}$$

$$k^n = \sum_{i=0}^{\min\{k,n\}}\begin{Bmatrix} n \\ i \end{Bmatrix}\binom{k}{i}i!$$

- 若 n 比较小，就能实现将 k^n 拆开后分项计算。

典例

题目大意

- 给定一个 DAG，点数为 n，边数为 m，设长度为 l 的路径的权值为 l^k，求 DAG 中所有路径的权值和，对 998244353 取模，$n,m \leq 2\times 10^5$，$0 \leq k \leq 500$。

解法

- 设以 u 为终点的路径集合为 P_u，另 $f_{u,i} = \sum\limits_{p\in P_u}\binom{|p|}{i}$。
- 初始时 $f_{u,i} = \binom{0}{i}$，根据组合数递推公式，对于有向边 $u \to v$，有转移

$$f_{v,i} \leftarrow f_{u,i-1} + f_{u,i}$$

- 最后答案为 $\sum\limits_{u=1}^{n}\sum\limits_{i=0}^{k}\begin{Bmatrix}k\\i\end{Bmatrix}i!f_{u,i}$，时间复杂度 $\mathcal{O}(nk)$。

第一类斯特林数

- 记作 $s(n,k)$ 或者 $\begin{bmatrix}n\\k\end{bmatrix}$，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。
- 易知 $\begin{bmatrix}n\\0\end{bmatrix}=[n=0],\begin{bmatrix}n\\k\end{bmatrix}=\begin{bmatrix}n-1\\k-1\end{bmatrix}+(n-1)\begin{bmatrix}n-1\\k\end{bmatrix}$。
- 设生成函数 $F_n(x)=\sum\limits_{k=0}^{n}\begin{bmatrix}n\\k\end{bmatrix}x^k$，根据递推式不难写出

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

- 容易通过分治 **NTT** 求出。

上升幂、下降幂、普通幂的相互转化

- 记上升幂 $x^{\overline{n}}=\prod\limits_{i=0}^{n-1}(x+i)$，下降幂 $x^{\underline{n}}=\prod\limits_{i=0}^{n-1}(x-i)=\frac{x!}{(x-n)!}=A_x^n$，容易根据定义得到：

$$(-x)^{\overline{n}}=\prod_{i=0}^{n-1}(-x+i)=(-1)^n\prod_{i=0}^{n-1}(x-i)=(-1)^nx^{\underline{n}}$$
$$(-x)^{\underline{n}}=\prod_{i=0}^{n-1}(-x-i)=(-1)^n\prod_{i=0}^{n-1}(x+i)=(-1)^nx^{\overline{n}}$$

- 关于两者与普通幂之间的转换有如下结论，证明过程以顺序编号（1）~（4）指代：

$$x^{\overline{n}}=\sum_{k=0}^{n}\begin{bmatrix}n\\k\end{bmatrix}x^k \tag{1}$$

$$x^n=\sum_{k=0}^{n}(-1)^{n-k}\begin{Bmatrix}n\\k\end{Bmatrix}x^{\overline{k}} \tag{2}$$

$$x^n=\sum_{k=0}^{n}\begin{Bmatrix}n\\k\end{Bmatrix}x^{\underline{k}} \tag{3}$$

$$x^{\underline{n}}=\sum_{k=0}^{n}(-1)^{n-k}\begin{bmatrix}n\\k\end{bmatrix}x^k \tag{4}$$

> **证明** （1）已经在 **第一类斯特林数** 一节内容中用生成函数证明，考虑到：
>
> $$(-x)^{\overline{n}}=(-1)^nx^{\underline{n}}=\sum_{k=0}^{n}(-1)^k\begin{bmatrix}n\\k\end{bmatrix}x^k$$
>
> 移项后即可得到（4）。
>
> （3）容易根据 $x^{\underline{k}}$ 和 x^n 的组合意义得到，利用类似的技巧，可以得到：
>
> $$(-x)^n=\sum_{k=0}^{n}\begin{Bmatrix}n\\k\end{Bmatrix}(-x)^{\underline{k}}=\sum_{k=0}^{n}(-1)^k\begin{Bmatrix}n\\k\end{Bmatrix}x^{\overline{k}}$$
>
> 整理移项后即可得到（2），证毕。

贝尔数

- 设 B_n 表示基为 n 的集合的划分数目，则 $B_n = \sum_{i=0}^{n} \begin{Bmatrix} n \\ i \end{Bmatrix}$。
- 考虑第 n 个元素归属的划分，B_n 有递推公式：

$$B_n = \sum_{i=0}^{n-1} \binom{n-1}{i} B_i = (n-1)! \sum_{i=0}^{n-1} \frac{B_i}{i!} \frac{1}{(n-1-i)!}$$

- 可以通过分治 **NTT** 快速求得 $B_i(1 \le i \le n)$。
- 另外，沿用 **第二类斯特林数** 一节内容中 $H(x)$ 的定义，$\exp H(x) = \sum_{i \ge 0} \frac{H^i(x)}{i!}$ 即为贝尔数的生成函数，也可通过这一方式来计算。

卡特兰数

- **通项公式** $h_n = C_{2n}^n - C_{2n}^{n+1} = \frac{C_{2n}^n}{n+1} = \frac{(2n)!}{n!(n+1)!}$
    - 在平面直角坐标系上，每一步只能往上走或往右走，从 $(0,0)$ 走到 (n,n) 并且两个端点外不接触直线 $y=x$ 的路线数量为 $2h_{n-1}$，不跨过直线 $y=x$ 的路线数量为 h_n，证明过程即通项公式的证明。
- **递推公式** $h_0 = h_1 = 1$，$h_n = \frac{(4n-2)h_{n-1}}{n+1}$，$h_n = \sum_{i=0}^{n-1} h_i h_{n-i-1}$
    - 入栈顺序为 $1,2,\ldots,n$，合法出栈顺序的方案数为 h_n。
    - 考虑枚举最后一个出栈的元素为 k，则小于 k 的元素必定在 k 入栈之前出栈，大于 k 的元素必定在 k 出栈之前出栈，可证得该递推式。
- **相关问题**
    - n 对括号的合法匹配数为 h_n。
    - n 个节点构成的不同二叉树的数量为 h_n。
    - 圆上 $2n$ 个点，两两配对连线不相交的方案数为 h_n。
    - n 条边的凸多边形三角剖分划分方案数 h_{n-2}（枚举其中一条边对应的三角剖分）。

分拆数

- 令 f_n 表示将 n 进行分拆的方案数，有上界 $f_n \le e^{\sqrt{\frac{20}{3}n}}$。
- 例如，$1+1+1+1 = 1+1+2 = 1+3 = 2+2 = 4$，所以 $f_4 = 5$。
- 下面给出几种常见的计算 $f_i(1 \le i \le n)$ 的方法。

根号分治

- 设 $S = \lfloor \sqrt{n} \rfloor$。
- 对于 $> S$ 的数，设 $g_{i,j}$ 表示选了 i 个数总和为 $i(S+1)+j$ 的方案数，转移有两种：
    1. 新加入一个数，初始值为 $S+1$：$g_{i,j} += g_{i-1,j}$。
    2. 给之前选的所有数 `+1`：$g_{i,j} += g_{i,j-i}$。
- 第一维只有 $\mathcal{O}(\sqrt{n})$ 级别，时间复杂度 $\mathcal{O}(n\sqrt{n})$。
- 求出 $> S$ 的数的 DP 数组后，对于 $\le S$ 的数，直接完全背包即可，总的时间复杂度 $\mathcal{O}(n\sqrt{n})$。

```
    s = sqrt(n);
    g[0] = 1;
    for (int i = 1; i <= s; ++i)
        for (int j = 0, jm = n - i * (s + 1); j <= jm; ++j)
        {
            if (j >= i)
                add(g[j], g[j - i]);
            add(f[i * (s + 1) + j], g[j]);
        }
    f[0] = 1;
    for (int i = 1; i <= s; ++i)
        for (int j = i; j <= n; ++j)
            add(f[j], f[j - i]);
```

生成函数

- 令 $f_0 = 1$，考虑 f_i 的生成函数：

$$\begin{aligned}F(x) &= \sum_{i=0}^{\infty} f_i x^i \\ &= \prod_{i=1}^{\infty} \frac{1}{1-x^i} \\ &= \exp\left(\sum_{i=1}^{\infty} \ln \frac{1}{1-x^i}\right)\end{aligned}$$

- 根据泰勒展开式

$$\ln \frac{1}{1-x^i} = -\ln(1-x^i) = \sum_{j=1}^{\infty} \frac{x^{ij}}{j}$$

- 所以

$$F(x) = \exp\left(\sum_{i=1}^{\infty}\sum_{j=1}^{\infty} \frac{x^{ij}}{j}\right)$$

- 可以 $\mathcal{O}(n \ln n)$ 预处理出内部系数，再 $\exp$ 回去，总时间复杂度 $\mathcal{O}(n \log n)$。

五边形数定理

- 即函数 $\Phi(x)$ 满足：

$$\Phi(x) = \prod_{i=1}^{\infty}(1-x^i) = \sum_{k=0}^{\infty}(-1)^k x^{\frac{k(3k-1)}{2}}$$

- 具体证明见 [visit_world博客中的证明]()。
- 由于 $\Phi(x)F(x) = 1$，可以直接多项式求逆，时间复杂度 $\mathcal{O}(n \log n)$。
- 注意到 $\Phi(x) \pmod{x^{n+1}}$ 中系数不为 0 的项只有 $\mathcal{O}(\sqrt{n})$ 个，也可以暴力进行求逆，时间复杂度 $\mathcal{O}(n\sqrt{n})$。

高阶前缀和

- 考虑一个序列 $a_1, a_2, a_3, \ldots$，设 $f(i,j)$ 表示序列 a 中求 j 次前缀和后 a_i 中包含多少个 a_1，显然 $\forall i \geq 1, f(i,1) = f(1,i) = 1$，容易得到转移：$f(i,j) = f(i,j-1) + f(i-1,j)$
- 利用坐标变换 $\begin{cases} x = i+j-2 \\ y = j-1 \end{cases}$ 得到 $g(x,y) = f(i,j)$，容易得到转移：$g(x,y) = g(x-1,y-1) + g(x-1,y)$
- 容易验证 $g(x,y) = \binom{x}{y}$，即 $\forall i \geq 1, j \geq 1, f(i,j) = \binom{i+j-2}{j-1}$。

高阶差分

与组合数的关系

- 定义 $\Delta^p h(n)$ 表示序列 h 经过 p 次差分后第 n 项的值，即

$$\Delta^{p+1} h(n-1) = \Delta^p h(n) - \Delta^p h(n-1)$$
$$\Delta^0 h(n) = h(n)$$

- **结论1** 容易证明差分可使多项式降次，因此可对 p 归纳，若 $h(n) = \sum\limits_{i=0}^{p} a_i n^i$，则 $\Delta^{p+1} h(n)$ 恒为 0。
- **结论2** 若 $\forall i \neq p, \Delta^i h(0) = 0$ 且 $\Delta^p h(0) = 1$， $h(n)$ 为 n 的 p 次多项式，则 $h(n) = \binom{n}{p}$。

> **证明** 容易得到 $\forall 0 \leq i < p$，$h(i) = 0$ 且 $h(p) = 1$，可设 $h(n) = c\prod\limits_{i=0}^{p-1}(n-i)$代入 $h(p) = 1$，解得 $c = \frac{1}{p!}$，即 $h(n) = \binom{n}{p}$。

- **结论3** 根据结论 1 并推广结论 2，已知 $\Delta^i h(0)(0 \leq i \leq p)$ 且 $h(n)$ 为 n 的 p 次多项式，则：$h(n) = \sum\limits_{i=0}^{p} \binom{n}{i} \Delta^i h(0)$该结论可用于快速求一些数列的通项。

区间加等差数列

- 将其差分两次后得到：

0	a_1	a_1+d	a_1+2d	$\ldots$	$a_1+(n-1)d$	**0**	**0**	**0**
0	a_1	d	d	$\ldots$	d	$-a_1-(n-1)d$	0	0
0	a_1	$d-a_1$	0	$\ldots$	0	$-a_1-nd$	$a_1+(n-1)d$	0

- 每次操作打上标记后，最后求两次前缀和即可（注意不要把 $a_1+(n-1)d$ 遗漏，其它高阶差分问题也要注意这样的边界问题）。

二维矩阵差分

- 设 $\Delta h(i,j) = h(i,j) - h(i-1,j) - h(i,j-1) + h(i-1,j-1)$。
- 倘若我们需要给一个矩阵的某个子方矩阵打上下面这样的标记：

$$\begin{matrix} 0 & 0 & 0 & 0 & \ldots & 0 & 0 \\ 0 & d & d & d & \ldots & d & 0 \\ 0 & d & 2d & 2d & \ldots & 2d & 0 \\ 0 & d & 2d & 3d & \ldots & 3d & 0 \\ \vdots & \vdots & \vdots & \vdots & \ddots & \vdots & \vdots \\ 0 & d & 2d & 3d & \ldots & nd & 0 \\ 0 & 0 & 0 & 0 & \ldots & 0 & 0 \end{matrix}$$

- 做一次二维矩阵差分后，可得到：

$$
\begin{matrix}
0 & 0 & 0 & 0 & \dots & 0 & 0 \\
0 & d & 0 & 0 & \dots & 0 & -d \\
0 & 0 & d & 0 & \dots & 0 & -d \\
0 & 0 & 0 & d & \dots & 0 & -d \\
\vdots & \vdots & \vdots & \vdots & \ddots & \vdots & \vdots \\
0 & 0 & 0 & 0 & \dots & d & -d \\
0 & -d & -d & -d & \dots & -d & nd
\end{matrix}
$$

- 通过对行、列、主对角线分别差分最后把标记合并即可。
- 类似地，倘若我们需要给一个矩阵的某个子方矩阵打上下面这样的标记：

$$
\begin{matrix}
0 & 0 & 0 & 0 & \dots & 0 & 0 \\
0 & d & d & d & \dots & d & 0 \\
0 & d & 3d & 3d & \dots & 3d & 0 \\
0 & d & 3d & 6d & \dots & 6d & 0 \\
\vdots & \vdots & \vdots & \vdots & \ddots & \vdots & \vdots \\
0 & d & 3d & 6d & \dots & \frac{n(n+1)}{2}d & 0 \\
0 & 0 & 0 & 0 & \dots & 0 & 0
\end{matrix}
$$

- 做一次二维矩阵差分后，可得到：

$$
\begin{matrix}
0 & 0 & 0 & 0 & \dots & 0 & 0 \\
0 & d & 0 & 0 & \dots & 0 & -d \\
0 & 0 & 2d & 0 & \dots & 0 & -2d \\
0 & 0 & 0 & 3d & \dots & 0 & -3d \\
\vdots & \vdots & \vdots & \vdots & \ddots & \vdots & \vdots \\
0 & 0 & 0 & 0 & \dots & nd & -nd \\
0 & -d & -2d & -3d & \dots & -nd & \frac{n(n+1)}{2}d
\end{matrix}
$$

- 通过对行、列、主对角线分别做区间加等差数列的差分，最后把标记合并即可。