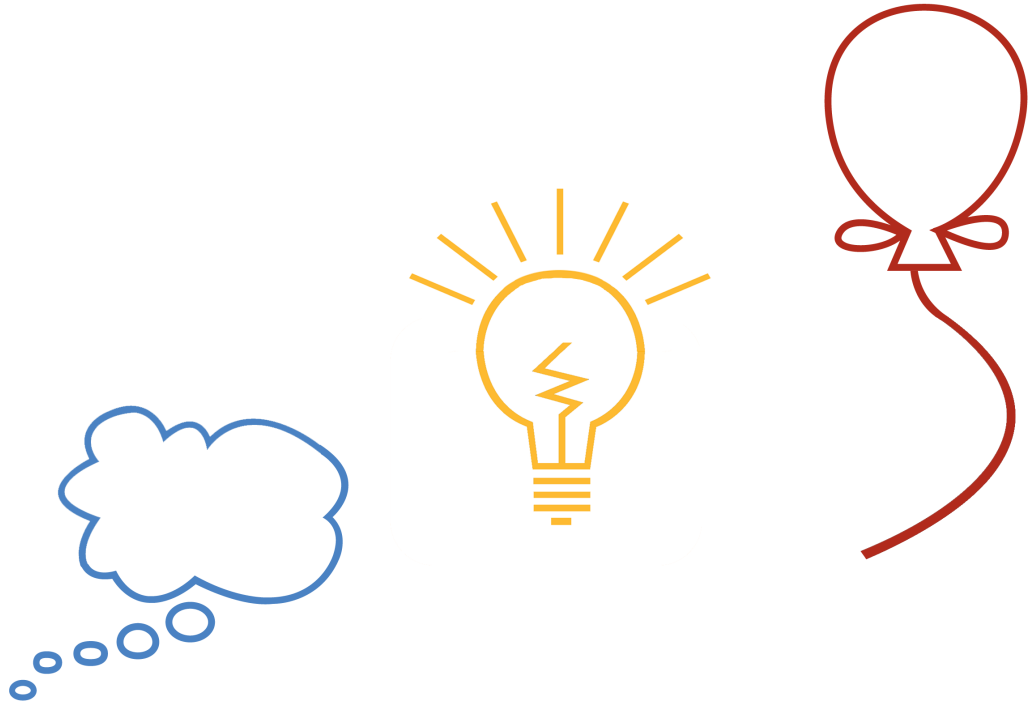
ACM 中的数学

hz

2022-01-19

参考：《具体数学》，oi-wiki，答案对 998244353 取模



组合数

$\binom{n}{k}$ 表示从 n 个不同的元素中，选出 k 个不同元素的方案数，组合数有以下简单恒等式：

$$\binom{n}{k} = \binom{n-1}{k-1} + \binom{n-1}{k}$$

$$\binom{n}{k} = \frac{n!}{k!(n-k)!}$$

$$\binom{n}{k} = \binom{n}{n-k}$$

$$\binom{n}{k} = \frac{n-k}{k}\binom{n}{k-1}$$

$$(1+x)^n = \binom{n}{0}x^0 + \binom{n}{1}x^1 + \binom{n}{2}x^2 + \cdots + \binom{n}{n}x^n$$

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

组合数还有一些其他的恒等式，但大部分可以用生成函数简单证明

组合数

广义组合数：从 n 个不同的元素中选出 a_1 个数染成颜色 1，a_2 个数染成颜色 2，...，a_m 个数染成颜色 m，$\sum a_i = n$，方案数为

$$\binom{n}{a_1, a_2, \ldots, a_m} = \frac{n!}{a_1! a_2! \ldots a_m!}$$

隔板法：n 个相同的元素划分成 m 份，每份大小不为 0，方案数 $\binom{n-1}{m-1}$，每份大小可以为 0，方案数 $\binom{n+m-1}{m-1}$

二项式反演：

$$\sum_k \binom{n}{k}\binom{k}{m}(-1)^{k-m} = [n=m]$$

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

直接展开即可证明，其组合意义会在“容斥”一章讲到

（广义）卡塔兰数

卡塔兰数 C_n 有很多组合意义，例如：

- 从 $(0,0)$ 走到 $(2n,0)$，每步的向量为 $(1,1)$ 或 $(1,-1)$，y 坐标始终不小于 0 的路径数
- 包含 n 组括号的合法括号序列的个数
- $\{1,\ldots,n\}$ 的进出栈方案数
- n 个节点组成不同构二叉树的方案数
- 将 $n+2$ 边的凸多边三角剖分的方案数
- $2\times n$ 杨表的数量（实际上卡塔兰数是勾长公式的一个特例，感兴趣的同学请自行了解）

C_n 的递推公式为：

$$\begin{aligned} C_n &= C_0\,C_{n-1} + C_1\,C_{n-2} + \cdots + C_{n-1}\,C_0 \\ &= \sum_{i=0}^{n-1} C_i\,C_{n-1-i} \end{aligned}$$

（广义）卡塔兰数

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

通项公式证明：

考虑组合意义：“从 $(0,0)$ 走到 $(2n,0)$，每步的向量为 $(1,1)$ 或 $(1,-1)$，y 坐标始终不小于 0 的路径数”

如果不考虑 y 坐标始终不小于 0 的条件，方案数为 $\binom{2n}{n}$
构造双射，可证明某一时刻 $y=-1$ 的路径数为 $\binom{2n}{n+1}$

广义卡塔兰数：从 $(0,0)$ 走到 $(2n,m)$，每步的向量为 $(1,1)$ 或 $(1,-1)$，y 坐标始终不小于 0 的路径数，证明方法相同

反射容斥：从 $(0,0)$ 走到 $(2n,m)$，每步的向量为 $(1,1)$ 或 $(1,-1)$，y 坐标始终在 $[L,R]$ 的路径数（较难，感兴趣的同学请自行了解）

斯特林数

第一类斯特林数：$\begin{bmatrix} n \\ k \end{bmatrix}$ 表示把 n 个数分成 k 个圆排列的方案数

考虑第 n 个数所属圆排列，有递推式：$\begin{bmatrix} n \\ k \end{bmatrix} = (n-1)\begin{bmatrix} n-1 \\ k \end{bmatrix} + \begin{bmatrix} n-1 \\ k-1 \end{bmatrix}$

第二类斯特林数：$\begin{Bmatrix} n \\ k \end{Bmatrix}$ 表示把 n 个数分成 k 个集合的方案数

考虑第 n 个数所属集合，有递推式：$\begin{Bmatrix} n \\ k \end{Bmatrix} = k\begin{Bmatrix} n-1 \\ k \end{Bmatrix} + \begin{Bmatrix} n-1 \\ k-1 \end{Bmatrix}$

在 ACM 中，斯特林数主要用于上升幂、通常幂、下降幂之间的转换：

上升幂：$x^{\overline{k}} = \frac{(x+k-1)!}{(x-1)!} = x \times (x+1) \times (x+2) \times \cdots \times (x+k-1)$

普通幂：$x^k = x \times x \times x \times \cdots \times x$

下降幂：$x^{\underline{k}} = \frac{x!}{(x-k)!} = x \times (x-1) \times (x-2) \times \cdots \times (x-k+1)$

斯特林数

与普通幂相比，上升 / 下降幂有一些优秀的性质：

$$(x+1)^{\overline{k}} = x^{\overline{k}} + k(x+1)^{\overline{k-1}}$$

$$(x+1)^{k} = \binom{k}{0}x^0 + \binom{k}{1}x^1 + \binom{k}{2}x^2 + \cdots + \binom{k}{k}x^k$$

$$(x+1)^{\underline{k}} = x^{\underline{k}} + kx^{\underline{k-1}}$$

（一般来说，下降幂更为常用，此外，下降幂和组合数之间的关联也很有用）

$$x^{\underline{k}} = \binom{x}{k} \times k!$$

斯特林数

在 ACM 中，斯特林数主要用于上升幂、通常幂、下降幂之间的转换：

$$x^{\overline{k}} = \sum_{i=0}^{k} \begin{bmatrix} k \\ i \end{bmatrix} x^i \qquad x^{\underline{k}} = \sum_{i=0}^{k} (-1)^{k-i} \begin{bmatrix} k \\ i \end{bmatrix} x^i$$

$$x^k = \sum_{i=0}^{k} \begin{Bmatrix} k \\ i \end{Bmatrix} x^{\underline{i}} \qquad x^k = \sum_{i=0}^{k} (-1)^{k-i} \begin{Bmatrix} k \\ i \end{Bmatrix} x^{\overline{i}}$$

前两条公式可以用生成函数证明，后两条公式可以考虑组合意义

对第三条公式进行二项式反演，可以得到第二类斯特林数的另一种表达式：

$$x^n = \sum_{k=0}^{n} \begin{Bmatrix} n \\ k \end{Bmatrix} x^{\underline{k}} = \sum_{k=0}^{x} \binom{x}{k} \begin{Bmatrix} n \\ k \end{Bmatrix} k!$$

$$\Longleftrightarrow \begin{Bmatrix} n \\ k \end{Bmatrix} \cdot k! = \sum_{i=0}^{k} \binom{k}{i} (-1)^{k-i} i^n$$

斯特林数

$$x^{\overline{k}} = \sum_{i=0}^{k} \begin{bmatrix} k \\ i \end{bmatrix} x^i \qquad x^{\underline{k}} = \sum_{i=0}^{k} (-1)^{k-i} \begin{bmatrix} k \\ i \end{bmatrix} x^i$$

$$x^k = \sum_{i=0}^{k} \begin{Bmatrix} k \\ i \end{Bmatrix} x^{\underline{i}} \qquad x^k = \sum_{i=0}^{k} (-1)^{k-i} \begin{Bmatrix} k \\ i \end{Bmatrix} x^{\overline{i}}$$

以上式子还说明了第一类斯特林数与第二类斯特林数之间存在一定的联系，称作斯特林反演

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

$$f_n = \sum_{i=0}^{n} \begin{Bmatrix} n \\ i \end{Bmatrix} g_i \iff g_n = \sum_{i=0}^{n} \begin{bmatrix} n \\ i \end{bmatrix} (-1)^{n-i} f_i$$

$$f_n = \sum_{i=0}^{n} \begin{bmatrix} n \\ i \end{bmatrix} g_i \iff g_n = \sum_{i=0}^{n} \begin{Bmatrix} n \\ i \end{Bmatrix} (-1)^{n-i} f_i$$

感兴趣的同学请自行了解

拉格朗日插值

通过 $n+1$ 个不同点的点值 (x_i, y_i) 可以唯一确定一个 n 阶多项式，其表达式为：

$$f(x) = \sum_i \left[\left(\prod_{j \neq i} \frac{x - x_j}{x_i - x_j} \right) \cdot y_i \right]$$

线性递推及优化

线性递推：给定前 m 项的值 $a_1, a_2, \ldots, a_m$ 及递推式 $\forall i > m,\ a_i = \sum_{j=1}^{m} b_j a_{i-j}$，求 a_n

最经典的例子：斐波那契数列 $\forall n > 2,\ F_n = F_{n-1} + F_{n-2}$

矩阵乘法优化：

$$\begin{pmatrix} a_{n-m+1} \\ a_{n-m+2} \\ a_{n-m+3} \\ \vdots \\ a_{n-1} \\ a_n \end{pmatrix} = \begin{pmatrix} 0 & 1 & 0 & \cdots & 0 & 0 \\ 0 & 0 & 1 & \cdots & 0 & 0 \\ 0 & 0 & 0 & \cdots & 0 & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots & \vdots \\ 0 & 0 & 0 & \cdots & 0 & 1 \\ b_m & b_{m-1} & b_{m-2} & \cdots & b_2 & b_1 \end{pmatrix} \begin{pmatrix} a_{n-m} \\ a_{n-m+1} \\ a_{n-m+2} \\ \vdots \\ a_{n-2} \\ a_{n-1} \end{pmatrix}$$

记中间的矩阵为 M，通过矩阵快速幂可以快速求出 a_n，复杂度 $O(m^3 \log n)$

$$\begin{pmatrix} a_{n-m+1} \\ \vdots \\ a_n \end{pmatrix} = \mathbf{M} \begin{pmatrix} a_{n-m} \\ \vdots \\ a_{n-1} \end{pmatrix} = \mathbf{M}^{n-m} \begin{pmatrix} a_1 \\ \vdots \\ a_m \end{pmatrix}$$

线性递推及优化

将下标最大元素为 a_n 的向量记作 $\mathbf{v}_n$，由之前的结论：$\mathbf{v}_n = \mathbf{M}^{n-m}\mathbf{v}_m$

Cayley-Hamilton 定理：

若矩阵 $\mathbf{M}$ 的特征多项式为: $f(x)$，则 $f(\mathbf{M}) = \mathbf{O}$ （全 0 矩阵）

对于线性递推 $a_i = \sum_{j=1}^{m} b_j a_{i-j}$，特征多项式
$f(x) = x^m - (\sum_{j=1}^{m} b_j x^{m-j})$

只需要快速幂计算 $x^{n-m} \bmod f(x) = c_0 + c_1 x^1 + \cdots + c_{m-1} x^{m-1}$ ，
$\mathbf{v}_n = \mathbf{M}^{n-m}\mathbf{v}_m = (c_0\mathbf{M}^0 + c_1\mathbf{M}^1 + \cdots + c_{m-1}\mathbf{M}^{m-1})\mathbf{v}_m$

复杂度 $O(m^2 \log n)$，还可以使用 fft 多项式取模算法进一步优化

在一些特殊的情况下，还可以通过矩阵对角化优化矩阵乘法

应用：自然数幂和

矩阵乘法、斯特林数、拉格朗日插值

容斥原理

在 ACM 中，最基础的容斥表达式如下，这也被称作子集反演，通过带入证明

$$f_S = \sum_{S \subseteq T} g_T \iff g_S = \sum_{S \subseteq T} (-1)^{|T|-|S|} f_T$$

$$f_S = \sum_{T \subseteq S} g_T \iff g_S = \sum_{T \subseteq S} (-1)^{|T|-|S|} f_T$$

如果 $\forall |S| = |T|, f_S = f_T = f'_{|S|}$ ，那就得到了二项式反演的表达式：

$$f'_i = \sum_{j=i}^{n} \binom{n-i}{n-j} g'_j \iff g'_i = \sum_{j=i}^{n} (-1)^{j-i} \binom{n-i}{n-j} f'_j$$

$$f'_i = \sum_{j=0}^{i} \binom{i}{j} g'_j \iff g'_i = \sum_{j=0}^{i} (-1)^{j-i} \binom{i}{j} f'_j$$

容斥原理

此外，还有莫比乌斯反演：（μ 是一个函数，稍后会提到）

$$f(n)=\sum_{d|n}g(d)\iff g(n)=\sum_{d|n}\mu(\frac{n}{d})f(d)$$

min-max 反演：

$$\max\{S\}=\sum_{T\subseteq S}(-1)^{|T|-1}\min\{T\}$$

都可以带入证明

容斥原理

简单容斥：$1 \ldots n$ 中 2 或 3 或 5 的倍数数量：
$\lfloor \frac{n}{2} \rfloor + \lfloor \frac{n}{3} \rfloor + \lfloor \frac{n}{5} \rfloor - \lfloor \frac{n}{6} \rfloor - \lfloor \frac{n}{10} \rfloor - \lfloor \frac{n}{15} \rfloor + \lfloor \frac{n}{30} \rfloor$

错排数问题：给定 n，求满足 $\forall 1 \leq i \leq n,\ p_i \neq i$ 的排列个数 D_n

设 g_S 表示恰好只有位置 S 满足 $p_i = i$ 的排列个数，f_S 表示位置 S 满足 $p_i = i$ 的排列个数，那么有 $f_S = (n - |S|)!, f_S = \sum_{S \subseteq T} g_T$

因此 $D_n = g_\varnothing = \sum_T (-1)^{|T|} f_T = \sum_{i=0}^{n} (-1)^i (n-i)!$

求字符集为 26, 最短周期为 n 的字符串个数

设 $g(n)$ 表示最短周期为 n 的字符串个数，$f(n)$ 表示长度为 n 的字符串个数，那么有 $f(n) = 26^n, f(n) = \sum_{d|n} g(d)$

因此 $g(n) = \sum_{d|n} \mu(\frac{n}{d}) 26^d$

置换群与 polya 计数

置换：

集合 $S = \{a_1, a_2, \ldots, a_n\}$ 上的置换可以表示为：

$$f = \begin{pmatrix} a_1 & a_2 & \ldots & a_n \\ a_{p_1} & a_{p_2} & \ldots & a_{p_n} \end{pmatrix}$$

其中 $p_1, p_2, \ldots p_n$ 是 $\{1, 2, \ldots, n\}$ 的一个排列

置换 f 可以理解为一个映射，将 a_i 映射到 a_{p_i}

任意一个置换都可以分解为若干不相交的循环置换的乘积，例如

$$\begin{pmatrix} a_1 & a_2 & a_3 & a_4 & a_5 \\ a_3 & a_1 & a_2 & a_5 & a_4 \end{pmatrix} = (a_1, a_3, a_2)(a_4, a_5)$$

定义置换乘法：

$$\begin{pmatrix} a_1 & a_2 & \ldots & a_n \\ a_{p_1} & a_{p_2} & \ldots & a_{p_n} \end{pmatrix} \circ \begin{pmatrix} a_{p_1} & a_{p_2} & \ldots & a_{p_n} \\ a_{q_1} & a_{q_2} & \ldots & a_{q_n} \end{pmatrix} = \begin{pmatrix} a_1 & a_2 & \ldots & a_n \\ a_{q_1} & a_{q_2} & \ldots & a_{q_n} \end{pmatrix}$$

置换群与 polya 计数

根据定义，群由一个集合以及一个定义在该群上的二元运算组成，记作 $(G, \cdot)$，群元素个数为 $|G|$ 并具有以下性质：

- 封闭性：$\forall a, b \in G,\ a \cdot b \in G$
- 结合律：$(a \cdot b) \cdot c = a \cdot (b \cdot c)$
- 单位元：$\exists! e \in G,\ \forall a \in G,\ e \cdot a = a \cdot e = a$
- 逆元：$\forall a \in G, \exists! b \in G,\ a \cdot b = b \cdot a = e, b = a^{-1}$

定义群 $(G, \cdot)$ 的子群 $(H, \cdot)$ 满足：

- $H \subseteq G$
- $\forall x, y \in H, x \cdot y \in H$
- $\forall x \in H, x^{-1} \in H$

拉格朗日定理：$|H|$ 是 $|G|$ 的约数

这里只介绍置换群：置换群由集合 S 上的若干置换组成，满足以上群性质

置换群与 polya 计数

轨道-稳定子定理：

$\forall a \in S$ 定义 a 的轨道为 $O_a = \{g(a) \mid g \in G\}$
通过构造双射，可以证明 $\forall x \in O_a$, $\sum_{g \in G} [g(a) = x] = \sum_{g \in G} [g(a) = a]$

Burnside 引理：

用于计算置换群的不同轨道数，或是说：“在一个置换群的作用下，本质不同的元素个数”

$$\sum_{g \in G} |X^g| = \sum_{a \in S} |H_a| = \sum_{a} |G|/|O_a| = |G| \cdot |X/G|$$

符号说明：

- X^g 表示置换 g 的不动点数量（$|X^g| = \sum_{a \in S} [g(a) = a]$）
- H_a 表示元素 a 的不变子群大小（$|H_a| = \sum_{g \in G} [g(a) = a]$）
- X/G 表示不同轨道数

Polya 定理讲的是染色情况下的计数，和 Burnside 引理没有太大区别

置换群与 polya 计数

举例：用两种颜色对 4 个节点染色，旋转意义下不同的方案数
（通过枚举可以看出不同方案有 0000, 0001, 0011, 0101, 0111, 1111）

首先，$S = \{0000, 0001, 0010, 0011, \ldots 1110, 1111\}$ 共 16 个

$$G = \left\{ \begin{pmatrix} 1 & 2 & 3 & 4 \\ 1 & 2 & 3 & 4 \end{pmatrix}, \begin{pmatrix} 1 & 2 & 3 & 4 \\ 2 & 3 & 4 & 1 \end{pmatrix}, \begin{pmatrix} 1 & 2 & 3 & 4 \\ 3 & 4 & 1 & 2 \end{pmatrix}, \begin{pmatrix} 1 & 2 & 3 & 4 \\ 4 & 1 & 2 & 3 \end{pmatrix} \right\}$$

轨道：

$$\begin{aligned}
&\{0,0,0,0\} \\
&\{1,0,0,0\}, \{0,1,0,0\}, \{0,0,1,0\}, \{0,0,0,1\} \\
&\{1,1,0,0\}, \{0,1,1,0\}, \{0,0,1,1\}, \{1,0,0,1\} \\
&\{1,0,1,0\}, \{0,1,0,1\} \\
&\{1,1,1,0\}, \{0,1,1,1\}, \{1,0,1,1\}, \{1,1,0,1\} \\
&\{1,1,1,1\}
\end{aligned}$$

不动点：$\{X^g\} = \{16, 2, 4, 2\}$，轨道数：$|X/G| = \frac{1}{4}(16 + 2 + 4 + 2) = 6$

图计数结论

树的拓扑序计数：$cnt(T) = \frac{n!}{\prod size_i}$，$size$ 表示该节点的子树大小

prufer 序列：

可以在“n 个节点的有标号无根树”和“长度为 $n-2$，每一项为 $1 \ldots n$ 的整数的数列”之间构造双射
（因此 n 个节点的有标号无根树的数量为 n^{n-2}）

扩展 prufer 序列：

同样通过构造双射，可以证明：

n 个结点已结成 m 个连通块，每个联通块的大小为 $\{a_1, a_2, \ldots, a_m\}$，再增加 $m-1$ 条边连成一颗树的方案数是：

$$n^{m-2} \prod_{i=1}^{m} a_i$$

矩阵树定理、LGV 引理：记结论

生成函数

多项式基础：

$$f(x) = a_0 + a_1 x^1 + a_2 x^2 + \dots$$
$$g(x) = b_0 + b_1 x^1 + b_2 x^2 + \dots$$
$$f(x)g(x) = a_0 b_0 + (a_0 b_1 + a_1 b_0) x^1 + (a_0 b_2 + a_1 b_1 + a_2 b_0) x^2 + \dots$$

$$[x^n] f(x) = a_n$$
$$[x^n] f(x)g(x) = \sum_{i=0}^{n} a_i b_{n-i}$$

给定 $f(x), g(x)$，通过 fft 算法，可以在 $O(n \log n)$ 复杂度算出 $f(x) \cdot g(x)$

都在 $\bmod x^n$ 下运算

生成函数

多项式基础：

在 $\bmod x^n$ 下运算，所有多项式都可以表示为：
$a_0 + a_1 x^1 + a_2 x^2 + \ldots a_{n-1} x^{n-1}$，有单位元 1，大部分元素有逆元

有以下常见级数，由于是在 $\bmod x^n$ 下运算，不用考虑其收敛性

$$\frac{1}{1-x} = 1 + x + x^2 + \ldots$$

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

$$\exp(x) = 1 + \frac{x}{1!} + \frac{x^2}{2!} + \ldots$$

$$\ln \frac{1}{1-x} = \sum_{k \geq 1} \frac{1}{k} x^k$$

上述等式中的 x 替换成其他多项式仍成立，还有相应的算法可以快速求 $f^{-1}, \ln(f), \exp(f)$

生成函数

数列 $\{a_0, a_1, a_2, \dots\}$ 的普通型生成函数 (OGF) 是 $a_0 + a_1 x + a_2 x^2 + \dots$，指数型生成函数是 $a_0 + a_1 \frac{x}{1!} + a_2 \frac{x^2}{2!} + \dots$

证明范德蒙德卷积 $\sum_k \binom{r}{k}\binom{s}{n-k} = \binom{r+s}{n}$

证明隔板法 $[x^m]\left(\frac{x}{1-x}\right)^n, [x^m]\left(\frac{1}{1-x}\right)^n$

证明二项式反演 $G = F * e^x \iff F = G * e^{-x}$

整数拆分问题 $\frac{1}{1-x} \cdot \frac{1}{1-x^2} \cdot \dots \cdot \frac{1}{1-x^n}$

用红蓝绿三种颜色给 n 个元素染色，红色蓝色个数均为偶数个，元素间（有 / 没有）区别

OGF : $\left(\frac{1}{1-x^2}\right)^2 \frac{1}{1-x}$ ，EGF : $\left(\frac{e^x+e^{-x}}{2}\right)^2 e^x$

BZOJ3456: n 个点有标号连通图计数，EGF ln

生成函数

生成函数符号化（解析组合）：

分为有标号计数和无标号计数两部分

生成函数是组合对象的投影

OGF 乘法：无标号笛卡尔积
EGF 乘法：有标号笛卡尔积

无标号 MSET 构造

EXP：有标号 SET 构造

连通图 <—> 图
树 <—> 森林
不可约多项式 <—> 多项式

裴蜀定理，扩展欧几里得算法，中国剩余定理

给定 $A, B, C \in \mathbb{Z}$，解方程 $xA + yB = C$

裴蜀定理：方程有解 $\iff \gcd(a, b) \mid C$

扩展欧几里得定理：

$$\begin{cases} xA + yB = C \\ A = pB + D \end{cases} \Longrightarrow (px + y)B + xD = C$$

中国剩余定理：$n_1, n_2, \cdots n_k$ 两两互质，方程

$$\begin{cases} x \equiv a_1 \pmod{n_1} \\ x \equiv a_2 \pmod{n_2} \\ \cdots \\ x \equiv a_k \pmod{n_k} \end{cases}$$

在 $\bmod \prod n_i$ 下有唯一解

乘法群

$\forall a, m \in \mathbb{N},\ (\exists b,\ ab \equiv 1\ (\mathrm{mod}\ m)) \iff ba + xm = 1$ 有解
$\iff \gcd(a, m) = 1$

$1 \ldots m$ 中所有与 m 互质的数在模 m 乘法下成一群，其大小为
$\varphi(m) = \sum_{i=1}^{m} [\gcd(i, m) = 1]$

费马小定理：$\forall a, p, p \in \mathrm{prime}\ , a^{p-1} \equiv 1\ (\mathrm{mod}\ m)$

欧拉定理：$\forall a, m, \gcd(a, m) = 1\ , a^{\varphi(m)} \equiv 1\ (\mathrm{mod}\ m)$

费马小定理是 m 为质数时的特殊情况，这两条定理可用于求逆元

扩展欧拉定理：$\forall a, m \gcd(a, m) \neq 1$ 时,

$$a^c \equiv \begin{cases} a^c & c < \varphi(m) \\ a^{(c\ \mathrm{mod}\ \varphi(m)) + \varphi(m)} & c \geq \varphi(m) \end{cases}$$

原根、离散对数、二次剩余

当 $m = p, 2p, 4p,\ p \in \text{prime}$ 时，模 m 乘法群是循环群，循环群可以由单个生成元生成，将这些生成元称作原根

设原根为 g，那么剩余系 $\{a \mid 1 \le a \le m, \gcd(a, m) = 1\}$ 可以表示为 $\{g^0, g^1, \ldots, g^{\varphi(m)-1}\}$

由循环群性质，模 m 乘法群原根的个数为 $\varphi(\varphi(m))$，但原根的分布是一个困难的问题，一般只能通过随机找到

离散对数：$\forall a, m, \gcd(a, m) = 1$，$g$ 为模 m 的原根，求 b 使得 $g^b \equiv a \pmod{m}$

BSGS 算法利用哈希表可以在 $O(\sqrt{m})$ 时间内求出离散对数

二次剩余：$\forall a, p, p \in \text{prime}$，是否存在 b，$b^2 \equiv a \pmod{p}$ ？

是否存在二次剩余的判别式：勒让德符号 $\left(\frac{a}{p}\right) = a^{\frac{p-1}{2}}$

若 $\left(\frac{a}{p}\right) = 1$，$a$ 是模 p 的二次剩余，若 $\left(\frac{a}{p}\right) = -1$，$a$ 不是

Cipolla 算法通过扩域可以在 $O(\log m)$ 时间内求出二次剩余

Lucas 定理，exLucas 定理

Lucas 定理：$O(p)$ 求 $\binom{n}{m} \bmod p$：

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

exLucas 定理：求 $\binom{n}{m} \bmod m$：
根据中国剩余定理，只需考虑 $\binom{n}{m} \bmod p^k$ 然后分解 $n! = p^x \cdot y$，其中 $\gcd(y, p) = 1$
注意到：

$$\max_x \{p^x \mid n!\} = \lfloor \frac{n}{p} \rfloor + \lfloor \frac{n}{p^2} \rfloor + \dots$$

$$\prod_{i=1}^{p^k} \frac{i}{\gcd(i, p^\infty)} = \prod_{i=(t-1)\cdot p^k+1}^{t\cdot p^k} \frac{i}{\gcd(i, p^\infty)}$$

可以 $O(\log n)$ 得到 x，$O(p^k)$ 得到 $y \bmod p^k$

积性函数与狄利克雷卷积

定义积性函数：f 为一个定义域为 $\mathbb{N}$ 的函数满足：

- $f(1) = 1$
- $\forall x, y \in \mathbb{N}, \gcd(x, y) = 1, f(x \cdot y) = f(x) \cdot f(y)$

根据上述定义，只需要知道 $\forall p \in \text{prime}, k \in \mathbb{N}^+$ 时 $f(p^k)$ 的值，就可以描述一个积性函数：$f(n) = f(p_1^{k_1}) f(p_2^{k_2}) \dots f(p_m^{k_m})$

常见积性函数举例：

- $\varepsilon(n) = \begin{cases} 1 & n = 1 \\ 0 & n \neq 1 \end{cases}$
- $\mathrm{id}(n) = 1$
- $\mathrm{id}_k(n) = n^k$
- $d(n) = \sum_{i=1}^{n} [i \mid n]$
- $d_k(n) = \sum_{i=1}^{n} [i \mid n] \, i^k$
- $\varphi(n) = \sum_{i=1}^{n} [\gcd(i, n) = 1]$
- $\mu(n)$
- ...

积性函数与狄利克雷卷积

积性：（设 $n = \prod_{i=1}^{m} p_i^{k_i}$）

$$d(n) = \sum_{i=1}^{n} [i \mid n] = \prod_{i=1}^{m} (k_i + 1)$$

$$d_k(n) = \sum_{i=1}^{n} [i \mid n]\, i^k = \prod_{i=1}^{m} (1 + p_i + \ldots p_i^{k_i})$$

$$\varphi(n) = \prod_{i=1}^{m} (p_i - 1) p_i^{k_i - 1}$$

$$\mu(n) = \prod_{i=1}^{m} \begin{cases} 1 & k = 0 \\ -1 & k = 1 \\ 0 & k > 1 \end{cases}$$

φ 的证明需要用到中国剩余定理

积性函数与狄利克雷卷积

狄利克雷卷积：两个定义域为 $\mathbb{N}$ 的函数的狄利克雷卷积为：

$$(f*g)(n)=\sum_{d|n}f(d)g(\frac{n}{d})$$

两个积性函数狄利克雷卷积的结果仍为积性函数：

$$\begin{aligned}(f*g)(p^k)&=\sum_{i=0}^{k}f(p^i)g(p^{k-i})\\ \prod_{i=1}^{m}(f*g)(p_i^{k_i})&=\prod_{i=1}^{m}\sum_{j=0}^{k_i}f(p_i^j)g(p_i^{k_i-j})\\ &=\sum_{0\le j_i\le k_i}(\prod_{i=1}^{m}f(p_i^{j_i}))(\prod_{i=1}^{m}g(p_i^{k_i-j_i}))\\ &=\sum_{d|n}f(d)g(\frac{n}{d})\end{aligned}$$

积性函数与狄利克雷卷积

积性函数表示：用 p^k 处的值表示积性函数，狄利克雷卷积可以用多项式乘法表示：

$$f(p^k) = \langle a_0, a_1, a_2 \ldots, \rangle$$
$$g(p^k) = \langle b_0, b_1, b_2 \ldots, \rangle$$
$$(f * g)(p^k) = \langle a_0 b_0, a_0 b_1 + a_1 b_0, a_0 b_2 + a_1 b_1 + a_2 b_0 \ldots, \rangle$$

从这个角度重新观察一下常见积性函数：

$$\varepsilon(p^k) = \langle 1, 0, 0, 0, \ldots \rangle$$
$$\mathrm{id}(p^k) = \langle 1, 1, 1, 1, \ldots \rangle$$
$$\mathrm{id}_k(p^k) = \langle 1, p^k, p^{2k}, p^{3k}, \ldots \rangle$$
$$d_k(p^k) = \langle 1, 1+p, 1+p+p^2, 1+p+p^2+p^3, \ldots \rangle$$
$$\varphi(p^k) = \langle 1, p-1, (p-1)p, (p-1)p^2, \ldots \rangle$$
$$\mu(p^k) = \langle 1, -1, 0, 0, \ldots \rangle$$

这也说明了这些函数之间的一些关系（可以发现 id 对应着前缀和，μ 对应着差分）：

$$\varepsilon * f = f \qquad d_k = \mathrm{id}_k * id$$
$$\mathrm{id} * \mu = \varepsilon \qquad \varphi = \mu * \mathrm{id}_1$$

积性函数与狄利克雷卷积

根据调和级数：

$$\sum_{i=1}^{n} \frac{n}{i} \approx n \ln n$$

计算两个已知函数的狄利克雷卷积的前 n 项的时间复杂度为 $O(n \log n)$

对于 $f(p^k)$ 形式简单的积性函数，可以通过线性筛 $O(n)$ 算出其前 n 项

在线性筛中，每个合数 m 只会从 $\frac{m}{\text{minp}(m)}$ 被遍历一次，$\text{minp}(m)$ 为 m 的最小质因子

整除分块与莫比乌斯反演

整除分块：

$$\{\lfloor \frac{n}{i} \rfloor \mid i \in \mathbb{N}, 1 \leq i \leq n\}$$

只有不超过 $2\sqrt{n}$ 种取值

另外，刚才的结论提供了两个重要的等式，在莫比乌斯反演中有很大作用：

$$\sum_{d|n} \mu(d) = [n = 1]$$

$$\sum_{d|n} \varphi(d) = n$$

整除分块与莫比乌斯反演

莫比乌斯反演：尝试使用 μ 和 φ 化简式子

举例：求 $\sum_{i=1}^{N}\sum_{i=1}^{M}\gcd(i,j)$

$$\begin{aligned}
\sum_{i=1}^{N}\sum_{i=1}^{M}\gcd(i,j) &= \sum_{i=1}^{N}\sum_{i=1}^{M}\sum_{d}[\gcd(i,j)=d]\,d \\
&= \sum_{d\geq 1} d \sum_{i=1}^{\lfloor \frac{N}{d} \rfloor}\sum_{i=1}^{\lfloor \frac{M}{d} \rfloor}[\gcd(i,j)=1] \\
&= \sum_{d\geq 1} d \sum_{i=1}^{\lfloor \frac{N}{d} \rfloor}\sum_{i=1}^{\lfloor \frac{M}{d} \rfloor}\sum_{t\mid \gcd(i,j)}\mu(t) \\
&= \sum_{d\geq 1} d \sum_{t\geq 1}\mu(t)\lfloor \frac{N}{dt} \rfloor \lfloor \frac{M}{dt} \rfloor \\
&= \sum_{x\geq 1}\varphi(x)\lfloor \frac{N}{x} \rfloor \lfloor \frac{M}{x} \rfloor
\end{aligned}$$

杜教筛

杜教筛用于计算函数前缀和，该函数通过简单函数之间的狄利克雷卷积得到

用 $(\Sigma f)(n)$ 表示 $f(1)+f(2)+\ldots f(n)$

发现 Σf，Σg 与 $\Sigma f * g$ 之间存在一定联系：

$$
\begin{aligned}
(\Sigma f * g)(n) &= \sum_{i \le n} (f*g)(i) = \sum_{xy \le n} f(x) \cdot g(y) \\
&= \sum_{x=1}^{n} f(x) \cdot (\Sigma g)(\lfloor \frac{n}{x} \rfloor) \\
&= \sum_{x=1}^{\sqrt{n}} \left\{ f(x) \cdot (\Sigma g)(\lfloor \frac{n}{x} \rfloor) + g(x) \cdot (\Sigma f)(\lfloor \frac{n}{x} \rfloor) \right\} \\
&\quad - \sum_{i=1}^{\sqrt{n}} \sum_{j=1}^{\sqrt{n}} f(i) g(j)
\end{aligned}
$$

可用于计算 $\Sigma\mu, \Sigma\varphi$，更强的结论是：只要维护两个积性函数的 $\sqrt{n}$ 个前缀和，就可以计算两个积性函数乘、除得到的积性函数对应位置的值

min25 筛

min25 筛用于求积性函数的前缀和，另外要求 $f(p^k)$ 容易计算、$\forall p \in \text{prime},\ f(p)$ 可以用 p 的多项式表示

min25 筛分为两部分，第一部分计算：

$$H_k = \sum_{i=1}^{n} [i \in \text{prime}]\ i^k$$

第二部分计算：

$$\sum_{j=1}^{\lfloor \frac{n}{j} \rfloor} f(i)$$