数论

fish

2024 年 2 月 20 日

Overview

裴蜀定理

- 设 a, b 是不全为 0 的整数，对任意整数 x, y，满足 $\gcd(a, b) | ax + by$。

裴蜀定理

- 设 a, b 是不全为 0 的整数，对任意整数 x, y，满足 $\gcd(a, b) | ax + by$。
- 存在整数 x, y，使得 $ax + by = \gcd(a, b)$。

扩展欧几里得算法

用于求解方程 $ax + by = \gcd(a, b)$ 的一组特解。

表示通解

对于方程 $ax + by = c$，其有解的充要条件为 $\gcd(a, b) \mid c$。

表示通解

对于方程 $ax + by = c$，其有解的充要条件为 $\gcd(a, b) \mid c$。
调用 exgcd，得到的结果记为 x_0, y_0。

表示通解

对于方程 $ax+by=c$，其有解的充要条件为 $\gcd(a,b) \mid c$。
调用 exgcd，得到的结果记为 x_0, y_0。
一组特解为

$$x_1 = \frac{x_0 c}{\gcd(a,b)}, y_1 = \frac{y_0 c}{\gcd(a,b)}$$

表示通解

对于方程 $ax + by = c$，其有解的充要条件为 $\gcd(a, b) \mid c$。
调用 exgcd，得到的结果记为 x_0, y_0。
一组特解为

$$x_1 = \frac{x_0 c}{\gcd(a, b)}, y_1 = \frac{y_0 c}{\gcd(a, b)}$$

通解为

$$x = x_1 + k\frac{b}{\gcd(a, b)}, y = y_1 - k\frac{a}{\gcd(a, b)}$$

其中 k 为任意整数。

特解范围

若 $b \neq 0$，则必有 $|x| \leq b, |y| \leq a$。

特解范围

若 $b \neq 0$，则必有 $|x| \leq b, |y| \leq a$。

证明

归纳法。设 x_1, y_1 为所求的解，x_2, y_2 为下一层的解。有 $x_1 = y_2, y_1 = x_2 - \lfloor \frac{a}{b} \rfloor y_2$。

特解范围

若 $b \neq 0$，则必有 $|x| \leq b, |y| \leq a$。

证明

归纳法。设 x_1, y_1 为所求的解，x_2, y_2 为下一层的解。有 $x_1 = y_2, y_1 = x_2 - \lfloor \frac{a}{b} \rfloor y_2$。
$\gcd(a, b) = b$ 时，下一层会终止递归，有 $x_2 = 1, y_2 = 0$，则 $x_1 = 0, y_1 = 1$。成立。

特解范围

证明

$\gcd(a, b) \neq b$ 时，有 $|x_2| \leq (a \bmod b), |y_2| \leq b$。此时

$$
\begin{aligned}
|x_1| &= |y_2| \leq b \\
|y_1| &\leq |x_2| + \lfloor \frac{a}{b} \rfloor |y_2| \\
&\leq (a \bmod b) + \lfloor \frac{a}{b} \rfloor b \\
&= a - \lfloor \frac{a}{b} \rfloor b + \lfloor \frac{a}{b} \rfloor b \\
&= a
\end{aligned}
$$

直线下整点个数

求解如下函数

$$f(a,b,c,n)=\sum_{i=0}^{n}\lfloor\frac{ai+b}{c}\rfloor$$

直线下整点个数

求解如下函数

$$f(a,b,c,n)=\sum_{i=0}^{n}\lfloor\frac{ai+b}{c}\rfloor$$

实际上这是类欧几里得算法的一部分。由于形式简单，应用较多，故单独讲解。

直线下整点个数

若 $a \geq c$ 或 $b \geq c$，可以转化为 $a < c, b < c$ 的情况

$$f(a,b,c,n) = \sum_{i=0}^{n} \lfloor \frac{ai+b}{c} \rfloor$$

$$= \sum_{i=0}^{n} \lfloor \frac{(\lfloor \frac{a}{c} \rfloor c + (a \bmod c))i + \lfloor \frac{b}{c} \rfloor c + (b \bmod c)}{c} \rfloor$$

$$= \frac{n(n+1)}{2} \lfloor \frac{a}{c} \rfloor + (n+1) \lfloor \frac{b}{c} \rfloor + \sum_{i=0}^{n} \lfloor \frac{(a \bmod c)i + (b \bmod c)}{c} \rfloor$$

$$= \frac{n(n+1)}{2} \lfloor \frac{a}{c} \rfloor + (n+1) \lfloor \frac{b}{c} \rfloor + f(a \bmod c, b \bmod c, c, n)$$

直线下整点个数

若 $a = 0$，答案是平凡的

$$f(a, b, c, n) = (n+1)\lfloor \frac{b}{c} \rfloor$$

接下来考虑一般情况

直线下整点个数

$$
\begin{aligned}
f(a,b,c,n) &= \sum_{i=0}^{n} \lfloor \frac{ai+b}{c} \rfloor \\
&= \sum_{i=0}^{n} \sum_{j=0}^{\lfloor \frac{ai+b}{c} \rfloor - 1} 1 \\
&= \sum_{j=0}^{\lfloor \frac{an+b}{c} \rfloor - 1} \sum_{i=0}^{n} [j < \lfloor \frac{ai+b}{c} \rfloor]
\end{aligned}
$$

直线下整点个数

$$
\begin{aligned}
f(a,b,c,n) &= \sum_{i=0}^{n} \lfloor \frac{ai+b}{c} \rfloor \\
&= \sum_{i=0}^{n} \sum_{j=0}^{\lfloor \frac{ai+b}{c} \rfloor - 1} 1 \\
&= \sum_{j=0}^{\lfloor \frac{an+b}{c} \rfloor - 1} \sum_{i=0}^{n} [j < \lfloor \frac{ai+b}{c} \rfloor]
\end{aligned}
$$

由于

$$
j < \lfloor \frac{ai+b}{c} \rfloor \iff \frac{jc+c-b-1}{a} < i
$$

故消掉 i。

直线下整点个数

令 $m=\lfloor\frac{an+b}{c}\rfloor$，有

$$
\begin{aligned}
f(a,b,c,n) &= \sum_{j=0}^{m-1} n-\lfloor\frac{jc+c-b-1}{a}\rfloor \\
&= nm-f(c,c-b-1,a,m-1)
\end{aligned}
$$

直线下整点个数

令 $m = \lfloor \frac{an+b}{c} \rfloor$，有

$$
\begin{aligned}
f(a,b,c,n) &= \sum_{j=0}^{m-1} n - \lfloor \frac{jc+c-b-1}{a} \rfloor \\
&= nm - f(c, c-b-1, a, m-1)
\end{aligned}
$$

于是可以递归计算。复杂度 $O(\log n)$。

CF1912J Joy of Pokémon Observation

给定 l_1, l_2, l_3, t，求满足 $l_1x + l_2y + l_3z = t$ 的非负整数对 (x, y, z) 的数量。$l_1, l_2, l_3 \le 16, t \le 10^9$。

CF1912J Joy of Pokémon Observation

给定 l_1, l_2, l_3, t，求满足 $l_1 x + l_2 y + l_3 z = t$ 的非负整数对 (x, y, z) 的数量。$l_1, l_2, l_3 \le 16, t \le 10^9$。
将 x, y, z 在 $\bmod l_3$ 意义下考虑，可以转化为数直线下整点个数。

GYM104090 Modulo Ruins the Legend

给定 $a_1, \cdots, a_n$，求两个数 s, d，最小化$(\sum_{i=1}^{n} a_i + s + id) \bmod m$。$n \le 10^5, m \le 10^9$。

P3518 SEJ-Strongbox

有一个密码箱，0 到 $n-1$ 中的某些整数是它的密码。且满足：若 a 和 b 是它的密码，则 $(a+b) \bmod n$ 也是它的密码（a, b 可以相等）。某人试了 k 次密码 $m_1, \cdots, m_k$，前 $k-1$ 次都失败了，最后一次成功了。问，该密码箱最多有多少种不同的密码。$n \leq 10^{14}, k \leq 2.5 \times 10^5$。

P3518 SEJ-Strongbox

有一个密码箱，0 到 $n-1$ 中的某些整数是它的密码。且满足：若 a 和 b 是它的密码，则 $(a+b) \bmod n$ 也是它的密码（a, b 可以相等）。某人试了 k 次密码 $m_1, \cdots, m_k$，前 $k-1$ 次都失败了，最后一次成功了。问，该密码箱最多有多少种不同的密码。
$n \leq 10^{14}, k \leq 2.5 \times 10^5$。
由裴蜀定理可知存在一个 d，满足所有 d 的倍数恰为所有密码，且 d 为 n 的因数。

P3518 SEJ-Strongbox

有一个密码箱，0 到 $n-1$ 中的某些整数是它的密码。且满足：若 a 和 b 是它的密码，则 $(a+b) \bmod n$ 也是它的密码（a, b 可以相等）。某人试了 k 次密码 $m_1, \cdots, m_k$，前 $k-1$ 次都失败了，最后一次成功了。问，该密码箱最多有多少种不同的密码。$n \leq 10^{14}, k \leq 2.5 \times 10^5$。

由裴蜀定理可知存在一个 d，满足所有 d 的倍数恰为所有密码，且 d 为 n 的因数。

处理出 n 的所有因数，质因数，复杂度 $O(d(n)w(n))$。

线性同余方程

形如 $ax \equiv b \pmod{n}$ 的方程，称为线性同余方程。

线性同余方程

形如 $ax \equiv b \pmod{n}$ 的方程，称为线性同余方程。
将其改写为 $ax + nk = b$ 的形式，则有解的充要条件为 $\gcd(a, n) \mid b$。

线性同余方程

形如 $ax \equiv b \pmod{n}$ 的方程，称为线性同余方程。
将其改写为 $ax + nk = b$ 的形式，则有解的充要条件为 $\gcd(a, n) \mid b$。
用扩展欧几里得算法可以求出一个特解，并表示出通解

$$x = \frac{x_0 b}{\gcd(a, n)} + k \frac{n}{\gcd(a, n)}$$

线性同余方程

形如 $ax \equiv b \pmod{n}$ 的方程，称为线性同余方程。
将其改写为 $ax + nk = b$ 的形式，则有解的充要条件为 $\gcd(a, n) \mid b$。
用扩展欧几里得算法可以求出一个特解，并表示出通解

$$x = \frac{x_0 b}{\gcd(a, n)} + k\frac{n}{\gcd(a, n)}$$

于是有

$$x \equiv \frac{x_0 b}{\gcd(a, n)} \pmod{\frac{n}{\gcd(a, n)}}$$

中国剩余定理

求解线性同余方程组

$$\begin{cases} x \equiv a_1 \pmod{m_1} \\ x \equiv a_2 \pmod{m_2} \\ \quad \vdots \\ x \equiv a_k \pmod{m_k} \end{cases}$$

中国剩余定理

求解线性同余方程组

$$\begin{cases} x \equiv a_1 \pmod{m_1} \\ x \equiv a_2 \pmod{m_2} \\ \quad \vdots \\ x \equiv a_k \pmod{m_k} \end{cases}$$

考虑合并两个同余方程 $x \equiv a_1 \pmod{m_1}, x \equiv a_2 \pmod{m_2}$。

中国剩余定理

将其转化为不定方程 $x=a_1+pm_1=a_2+qm_2$，则 $m_1p-m_2q=a_2-a_1$。有解的充要条件是 $\gcd(m_1,m_2)\mid a_2-a_1$。

中国剩余定理

将其转化为不定方程 $x = a_1 + pm_1 = a_2 + qm_2$，则 $m_1p - m_2q = a_2 - a_1$。有解的充要条件是 $\gcd(m_1, m_2) \mid a_2 - a_1$。
用扩展欧几里得算法可以求出一个特解，并表示出通解

$$p = \frac{p_0(a_2 - a_1)}{\gcd(m_1, m_2)} + k\frac{m_2}{\gcd(m_1, m_2)}$$

中国剩余定理

将其转化为不定方程 $x = a_1 + pm_1 = a_2 + qm_2$，则 $m_1p - m_2q = a_2 - a_1$。有解的充要条件是 $\gcd(m_1, m_2) \mid a_2 - a_1$。
用扩展欧几里得算法可以求出一个特解，并表示出通解

$$p = \frac{p_0(a_2 - a_1)}{\gcd(m_1, m_2)} + k\frac{m_2}{\gcd(m_1, m_2)}$$

于是

$$x = a_1 + \frac{m_1p_0(a_2 - a_1)}{\gcd(m_1, m_2)} + k\frac{m_1m_2}{\gcd(m_1, m_2)}$$

$$x \equiv a_1 + \frac{m_1p_0(a_2 - a_1)}{\gcd(m_1, m_2)} \pmod{\operatorname{lcm}(m_1, m_2)}$$

中国剩余定理

将其转化为不定方程 $x = a_1 + pm_1 = a_2 + qm_2$，则 $m_1p - m_2q = a_2 - a_1$。有解的充要条件是 $\gcd(m_1, m_2) \mid a_2 - a_1$。

用扩展欧几里得算法可以求出一个特解，并表示出通解

$$p = \frac{p_0(a_2 - a_1)}{\gcd(m_1, m_2)} + k\frac{m_2}{\gcd(m_1, m_2)}$$

于是

$$x = a_1 + \frac{m_1p_0(a_2 - a_1)}{\gcd(m_1, m_2)} + k\frac{m_1m_2}{\gcd(m_1, m_2)}$$

$$x \equiv a_1 + \frac{m_1p_0(a_2 - a_1)}{\gcd(m_1, m_2)} \pmod{\operatorname{lcm}(m_1, m_2)}$$

两两合并即可。

P4621 BAKTERIJE

k 个细菌被放在一个 $n \times m$ 的矩形区域里。每个细菌都有自己的初始位置、方向与运动规则。
每秒每个细菌读取自己在这个单元格的数字 x，顺时针转 90° x 次，如果它面对矩形边界，则转 180°，最后进入自己面向的单元格。
放置一个陷阱在某个单元格，问什么时候所有细菌第一次同时在陷阱里。$n, m \leq 50, k \leq 5$。

CF338D GCD Table

给出序列 $a_1,\cdots,a_k$，判定是否存在 x,y，满足 $1\le x\le n, 1\le y\le m$，且对于 $i=1\sim k$，均有 $\gcd(x,y+i-1)=a_i$。$k\le 10^4$，$n,m,a_i\le 10^{12}$。

CF338D GCD Table

给出序列 $a_1, \cdots, a_k$，判定是否存在 x, y，满足 $1 \le x \le n, 1 \le y \le m$，且对于 $i = 1 \sim k$，均有 $\gcd(x, y + i - 1) = a_i$。$k \le 10^4$，$n, m, a_i \le 10^{12}$。
$\gcd(a, b) = c$ 的必要条件是 $c|a, c|b$。于是可以列出线性同余方程组。

CF338D GCD Table

给出序列 $a_1, \cdots, a_k$，判定是否存在 x, y，满足 $1 \le x \le n, 1 \le y \le m$，且对于 $i = 1 \sim k$，均有 $\gcd(x, y + i - 1) = a_i$。$k \le 10^4$，$n, m, a_i \le 10^{12}$。

$\gcd(a, b) = c$ 的必要条件是 $c|a, c|b$。于是可以列出线性同余方程组。

$x = k_1 \cdot \text{lcm}(a_{1\sim k}), y = k_2 \cdot \text{lcm}(a_{1\sim k}) + y_0$。

CF338D GCD Table

给出序列 $a_1, \cdots, a_k$，判定是否存在 x, y，满足 $1 \le x \le n, 1 \le y \le m$，且对于 $i = 1 \sim k$，均有 $\gcd(x, y + i - 1) = a_i$。$k \le 10^4$，$n, m, a_i \le 10^{12}$。

$\gcd(a, b) = c$ 的必要条件是 $c|a, c|b$。于是可以列出线性同余方程组。

$x = k_1 \cdot \mathsf{lcm}(a_{1\sim k}), y = k_2 \cdot \mathsf{lcm}(a_{1\sim k}) + y_0$。

还要满足恰好为 gcd。$k_1 > 1$ 一定不优，于是可以确定 x。y 的取值不影响。

带入判定即可。

卢卡斯定理

对于质数 p，有

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

卢卡斯定理

对于质数 p，有

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

证明

对于质数 p，有 $\binom{p}{x} \bmod p = [x = 0 \vee x = p]$，$(a+b)^p \equiv a^p + b^p \pmod{p}$。

卢卡斯定理

证明

考虑生成函数 $(1+x)^n$，有

$$
\begin{aligned}
\binom{n}{m} &= (1+x)^n[x^m] \\
&= (1+x)^{p\lfloor n/p \rfloor}(1+x)^{n \bmod p}[x^m] \\
&\equiv (1+x^p)^{\lfloor n/p \rfloor}(1+x)^{n \bmod p}[x^m] \pmod p \\
&\equiv (1+x^p)^{\lfloor n/p \rfloor}[x^{p\lfloor m/p \rfloor}](1+x)^{n \bmod p}[x^{m \bmod p}] \pmod p \\
&\equiv \binom{\lfloor n/p \rfloor}{\lfloor m/p \rfloor}\binom{n \bmod p}{m \bmod p} \pmod p
\end{aligned}
$$

勒让德定理

记 $v_p(x)$ 表示 x 质因数分解后 p 的指数，$S_p(x)$ 表示 p 进制下 x 各数位之和，则

$$v_p(n!) = \sum_{i=1}^{\infty} \lfloor \frac{n}{p^i} \rfloor = \frac{n - s_p(n)}{p-1}$$

勒让德定理

记 $v_p(x)$ 表示 x 质因数分解后 p 的指数，$S_p(x)$ 表示 p 进制下 x 各数位之和，则

$$v_p(n!) = \sum_{i=1}^{\infty} \lfloor \frac{n}{p^i} \rfloor = \frac{n - s_p(n)}{p-1}$$

证明

$$v_p(n!) = \sum_{i=1}^{n} v_p(i) = \sum_{i=1}^{n} \sum_{j=1}^{\infty} [p^j \mid i] = \sum_{j=1}^{\infty} \lfloor \frac{n}{p^j} \rfloor$$

勒让德定理

记 $v_p(x)$ 表示 x 质因数分解后 p 的指数，$S_p(x)$ 表示 p 进制下 x 各数位之和，则

$$v_p(n!) = \sum_{i=1}^{\infty} \lfloor \frac{n}{p^i} \rfloor = \frac{n - s_p(n)}{p-1}$$

证明

$$v_p(n!) = \sum_{i=1}^{n} v_p(i) = \sum_{i=1}^{n} \sum_{j=1}^{\infty} [p^j \mid i] = \sum_{j=1}^{\infty} \lfloor \frac{n}{p^j} \rfloor$$

后面的等号容易用归纳法证明。

CF711E ZS and The Birthday Paradox

求一年有 2^n 天，m 个人出现两人生日相同的可能性是多少。用分式表示，分子分母对 10^6+3 取模。$n, m \le 10^{18}$。

CF711E ZS and The Birthday Paradox

求一年有 2^n 天，m 个人出现两人生日相同的可能性是多少。用分式表示，分子分母对 10^6+3 取模。$n, m \le 10^{18}$。
容斥，求 $\frac{(2^n)!}{(2^n-m)!2^{nm}}$。

CF711E ZS and The Birthday Paradox

求一年有 2^n 天，m 个人出现两人生日相同的可能性是多少。用分式表示，分子分母对 10^6+3 取模。$n,m \le 10^{18}$。

容斥，求 $\frac{(2^n)!}{(2^n-m)!2^{nm}}$。

$x < 2^n, v_2(x) = v_2(2^n - x)$。

库默尔定理

$v_p(\binom{n+m}{n})$ 等于 n 与 m 在 p 进制下相加的进位次数。

库默尔定理

$v_p(\binom{n+m}{n})$ 等于 n 与 m 在 p 进制下相加的进位次数。

证明

将 n,m 写成 p 进制的形式

$$n = \sum_{i=0}^{\infty} a_i p^i$$

$$m = \sum_{i=0}^{\infty} b_i p^i$$

库默尔定理

证明

于是

$$v_p\left(\binom{n+m}{n}\right) = v_p((n+m)!) - v_p(n!) - v_p(m!)$$

$$= \sum_{i=1}^{\infty} \lfloor \frac{n+m}{p^i} \rfloor - \lfloor \frac{n}{p^i} \rfloor - \lfloor \frac{m}{p^i} \rfloor$$

$$= \sum_{i=1}^{\infty} \lfloor \frac{\sum_{j=0}^{i-1}(a_j+b_j)p^j}{p^i} + \sum_{j=i}^{\infty}(a_j+b_j)p^{j-i} \rfloor - \sum_{j=i}^{\infty} a_j p^{j-i} - \sum_{j=i}^{\infty} b_j p^{j-i}$$

$$= \sum_{i=1}^{\infty} [\sum_{j=0}^{i-1}(a_j+b_j)p^j \geq p_i]$$

P5598 混乱度

有 n 种颜色的球，其中第 i 种颜色的球有 a_i 个，同色的球不区分。
定义第 $l \sim r$ 种颜色的球的混乱度 $f(l,r)$ 为：将第 $l \sim r$ 种颜色的球排成一排的方案数对 p 取模的结果。
求 $\sum_{l=1}^{n}\sum_{r=l}^{n} f(l,r)$。$n \le 5\times 10^5, a_i \le 10^{18}, p \in \{2,3,5,7\}$。

扩展卢卡斯定理

对于模数 M 不是质数的情况，可以先将其质因数分解

$$M = \prod p_i^{k_i}$$

对每个 p^k 分别求出答案，再用中国剩余定理拼起来。

扩展卢卡斯定理

$$\binom{n}{m} \bmod p^k = \frac{n!}{m!(n-m)!} \bmod p^k$$

扩展卢卡斯定理

$$\binom{n}{m} \bmod p^k = \frac{n!}{m!(n-m)!} \bmod p^k$$

因为 $m!$ 与 $(n-m)!$ 不一定有逆元，所以尝试把分子分母中的 p 因子全部提出来。这一步的目的是使分母存在逆元。

扩展卢卡斯定理

$$\binom{n}{m} \bmod p^k = \frac{n!}{m!(n-m)!} \bmod p^k$$

因为 $m!$ 与 $(n-m)!$ 不一定有逆元，所以尝试把分子分母中的 p 因子全部提出来。这一步的目的是使分母存在逆元。
设 $f(x)$ 表示 $x!$ 移除所有 p 因子后的结果，$g(x)$ 表示 $x!$ 中因子 p 的个数。则有

$$\binom{n}{m} \bmod p^k = \frac{f(n)}{f(m)f(n-m)} p^{g(n)-g(m)-g(n-m)} \bmod p^k$$

扩展卢卡斯定理

$$
\begin{aligned}
n! &= \prod_{i=1}^{\lfloor n/p \rfloor} pi \prod_{i,(i,p)=1}^{n} i \\
&= p^{\lfloor n/p \rfloor} (\lfloor \frac{n}{p} \rfloor)! \prod_{i,(i,p)=1}^{n} i \\
&= p^{\lfloor n/p \rfloor} (\lfloor \frac{n}{p} \rfloor)! (\prod_{i,(i,p)=1}^{p^k} i)^{\lfloor n/p^k \rfloor} \prod_{i,(i,p)=1}^{n \bmod p^k} i
\end{aligned}
$$

扩展卢卡斯定理

等式右边后两项不含因子 p。则有

$$f(n) \equiv f(\lfloor \frac{n}{p} \rfloor)(\prod_{i,(i,p)=1}^{p^k} i)^{\lfloor n/p^k \rfloor} \prod_{i,(i,p)=1}^{n \bmod p^k} i \pmod{p^k}$$

$$g(n) = \lfloor \frac{n}{p} \rfloor + g(\lfloor \frac{n}{p} \rfloor)$$

扩展卢卡斯定理

等式右边后两项不含因子 p。则有

$$f(n) \equiv f(\lfloor \frac{n}{p} \rfloor)(\prod_{i,(i,p)=1}^{p^k} i)^{\lfloor n/p^k \rfloor} \prod_{i,(i,p)=1}^{n \bmod p^k} i \pmod{p^k}$$

$$g(n) = \lfloor \frac{n}{p} \rfloor + g(\lfloor \frac{n}{p} \rfloor)$$

对每个 p^k，预处理复杂度 $O(p^k)$，每次查询 $O(\log^2 n)$。

扩展卢卡斯定理

等式右边后两项不含因子 p。则有

$$f(n) \equiv f(\lfloor \frac{n}{p} \rfloor)(\prod_{i,(i,p)=1}^{p^k} i)^{\lfloor n/p^k \rfloor} \prod_{i,(i,p)=1}^{n \bmod p^k} i \pmod{p^k}$$

$$g(n) = \lfloor \frac{n}{p} \rfloor + g(\lfloor \frac{n}{p} \rfloor)$$

对每个 p^k，预处理复杂度 $O(p^k)$，每次查询 $O(\log^2 n)$。
用威尔逊定理可以除掉快速幂，优化至 $O(\log n)$。

欧拉定理

当 a 与 m 互质时，有 $a^{\varphi(m)} \equiv 1 \pmod{m}$。

欧拉定理

当 a 与 m 互质时，有 $a^{\varphi(m)} \equiv 1 \pmod m$。

证明

称 $S=\{p_1,p_2,\cdots,p_{\varphi(m)}\}$ 为 m 的简化剩余系，其中 p_i 表示从小到大第 i 个与 m 互质的数。

欧拉定理

当 a 与 m 互质时，有 $a^{\varphi(m)} \equiv 1 \pmod m$。

证明

称 $S = \{p_1, p_2, \cdots, p_{\varphi(m)}\}$ 为 m 的简化剩余系，其中 p_i 表示从小到大第 i 个与 m 互质的数。

$\forall i, j, \exists k, p_i p_j \equiv p_k \pmod m$。

欧拉定理

当 a 与 m 互质时，有 $a^{\varphi(m)} \equiv 1 \pmod{m}$。

证明

称 $S = \{p_1, p_2, \cdots, p_{\varphi(m)}\}$ 为 m 的简化剩余系，其中 p_i 表示从小到大第 i 个与 m 互质的数。

$\forall i, j, \exists k, p_i p_j \equiv p_k \pmod{m}$。

$\forall i \neq j, ap_i \not\equiv ap_j \pmod{m}$。故

$\{p_1, \cdots, p_{\varphi(m)}\} = \{ap_1 \bmod m, \cdots, ap_{\varphi(m)} \bmod m\}$。

欧拉定理

当 a 与 m 互质时，有 $a^{\varphi(m)} \equiv 1 \pmod m$。

证明

称 $S = \{p_1, p_2, \cdots, p_{\varphi(m)}\}$ 为 m 的简化剩余系，其中 p_i 表示从小到大第 i 个与 m 互质的数。
$\forall i, j, \exists k, p_i p_j \equiv p_k \pmod m$。
$\forall i \neq j, ap_i \not\equiv ap_j \pmod m$。故
$\{p_1, \cdots, p_{\varphi(m)}\} = \{ap_1 \bmod m, \cdots, ap_{\varphi(m)} \bmod m\}$。
$\prod_{i=1}^{\varphi(m)} p_i \equiv \prod_{i=1}^{\varphi(m)} ap_i \pmod m$，故 $a^{\varphi(m)} \equiv 1 \pmod m$。

阶

令 $a^x \equiv 1 \pmod{m}$ 的最小正整数 x 称为 a 模 m 的阶，记作 $\delta_m(a)$。

阶

令 $a^x \equiv 1 \pmod{m}$ 的最小正整数 x 称为 a 模 m 的阶，记作 $\delta_m(a)$。

$\gcd(a,m)=1$ 是存在阶的充要条件。故接下来只考虑 a,m 互质的情况。

阶

令 $a^x \equiv 1 \pmod m$ 的最小正整数 x 称为 a 模 m 的阶，记作 $\delta_m(a)$。

$\gcd(a,m)=1$ 是存在阶的充要条件。故接下来只考虑 a,m 互质的情况。

- $a, a_2, \cdots, a^{\delta_m(a)}$ 模意义下两两不同余。

阶

令 $a^x \equiv 1 \pmod m$ 的最小正整数 x 称为 a 模 m 的阶，记作 $\delta_m(a)$。
$\gcd(a, m) = 1$ 是存在阶的充要条件。故接下来只考虑 a, m 互质的情况。

- $a, a_2, \cdots, a^{\delta_m(a)}$ 模意义下两两不同余。
- 若 $a^x \equiv 1 \pmod m$，则$\delta_m(a) \mid x$。

阶

令 $a^x \equiv 1 \pmod m$ 的最小正整数 x 称为 a 模 m 的阶，记作 $\delta_m(a)$。
$\gcd(a, m) = 1$ 是存在阶的充要条件。故接下来只考虑 a, m 互质的情况。

- $a, a_2, \cdots, a^{\delta_m(a)}$ 模意义下两两不同余。
- 若 $a^x \equiv 1 \pmod m$，则$\delta_m(a) \mid x$。
- $\delta_m(a^k) = \frac{\delta_m(a)}{\gcd(\delta_m(a), k)}$。

求法

求 $\delta_m(a)$。

求法

求 $\delta_m(a)$。
$a^{\varphi(m)} \equiv 1 \pmod{m}$，故不断尝试除掉一些质因子即可。

原根

$\gcd(a, m) = 1$，若 $\delta_m(a) = \varphi(m)$，则称 a 是模 m 的原根。
检验原根，只需检验指数为 $\varphi(m)$ 除以每个质因子的情况即可。

原根

$\gcd(a,m)=1$，若 $\delta_m(a)=\varphi(m)$，则称 a 是模 m 的原根。
检验原根，只需检验指数为 $\varphi(m)$ 除以每个质因子的情况即可。
一个数存在原根当且仅当其为 $2,4,p^\alpha,2p^\alpha$。

求原根

若 m 存在原根，则使得 $\delta_m(a)=l$ 的 a 的个数为
$$\begin{cases} 0 & l \nmid \varphi(m) \\ \varphi(l) & l \mid \varphi(m) \end{cases}$$。

求原根

若 m 存在原根，则使得 $\delta_m(a) = l$ 的 a 的个数为 $\begin{cases} 0 & l \nmid \varphi(m) \\ \varphi(l) & l \mid \varphi(m) \end{cases}$。
于是可知原根个数为 $\varphi(\varphi(m))$。

求原根

若 m 存在原根，则使得 $\delta_m(a) = l$ 的 a 的个数为
$$\begin{cases} 0 & l \nmid \varphi(m) \\ \varphi(l) & l \mid \varphi(m) \end{cases}$$。
于是可知原根个数为 $\varphi(\varphi(m))$。
如何找到所有原根？先找到任一个 g，$g^k, (\gcd(k, \varphi(m)) = 1)$ 即为所有原根。

求原根

若 m 存在原根，则使得 $\delta_m(a) = l$ 的 a 的个数为
$$\begin{cases} 0 & l \nmid \varphi(m) \\ \varphi(l) & l \mid \varphi(m) \end{cases}$$。
于是可知原根个数为 $\varphi(\varphi(m))$。
如何找到所有原根？先找到任一个 g，$g^k, (\gcd(k, \varphi(m)) = 1)$ 即为所有原根。
对于素数 p，其最小原根是 $O(n^{1/4})$ 级别的。故暴力找的复杂度是可以接受的。

高次剩余

给 a, b, p，p 为质数，求 $0 \le x < p, x^a \equiv b \pmod p$ 的所有 x。

UOJ525 平行四边形

一个 $n \times n$ 的棋盘，放入 n 个棋子使得没有两个棋子在同行或同列，没有四个棋子构成平行四边形（包括退化）。保证 $n+1$ 为质数。$n \leq 1000$。

CF360D Levko and Sets

有两个整数数组 $a_1, \cdots, a_n$ 和 $b_1, \cdots, b_m$，与一个质数 p，现在要生成 n 个集合，第 i 个集合生成方式如下：

- 开始只有元素 1。
- 从集合中选出一个元素 c，对于所有 j，若 $c \times a_i^{b_j} \bmod p$ 不在当前集合中，将其加入集合。
- 不断重复直到集合中元素不变。

求 n 个集合并的大小。

$n \leq 10^4, m \leq 10^5, p \leq 10^9, a_i < p, b_i \leq 10^9$。

CF360D Levko and Sets

原根处理指数。

CF360D Levko and Sets

原根处理指数。
裴蜀定理。

CF360D Levko and Sets

原根处理指数。
裴蜀定理。

$$\{1, x, \cdots, x^{\delta(x)-1}\} = \{1, g^{\varphi(p)/\delta(x)}, \cdots, g^{(\delta(x)-1)(\varphi(p)/\delta(x))}\}$$

一些定义

数论函数是一种定义域为正整数的函数。

一些定义

数论函数是一种定义域为正整数的函数。
若函数 f 满足对任意 $\gcd(x, y) = 1$，有 $f(xy) = f(x)f(y)$，则称其为积性函数。
若函数 f 满足对任意 x, y，有 $f(xy) = f(x)f(y)$，则称其为完全积性函数。

简单运算

加法

$$(f+g)(n)=f(n)+g(n)$$

简单运算

加法

$$(f+g)(n)=f(n)+g(n)$$

点乘

$$(f\cdot g)(n)=f(n)g(n)$$

简单运算

加法

$$(f+g)(n)=f(n)+g(n)$$

点乘

$$(f\cdot g)(n)=f(n)g(n)$$

狄利克雷卷积

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

常见积性函数

- $\epsilon(n) = [n = 1]$
- $\mathrm{id}_k(n) = n^k$，id_1 常记作 id
- $1(n) = 1$
- $\sigma_k(n) = \sum_{d|n} d^k$，$\sigma_0$ 常记作 d，σ_1 常记作 σ
- $\varphi(n) = \sum_{i=1}^{n} [\gcd(i, n) = 1]$
- $\mu(n) = \begin{cases} 1 & n = 1 \\ 0 & \exists d > 1, d^2 | n \\ (-1)^{w(n)} & \text{otherwise} \end{cases}$
- $\chi_k(n) = [\gcd(n, k) = 1]$

贝尔级数

积性函数由其在质数幂处的取值决定。定义积性函数 f 的贝尔级数为

$$F_p(x) = \sum_{k \geq 0} f(p^k) x^k$$

贝尔级数

积性函数由其在质数幂处的取值决定。定义积性函数 f 的贝尔级数为

$$F_p(x) = \sum_{k \geq 0} f(p^k) x^k$$

积性函数卷积，对应的贝尔级数相乘。

贝尔级数

常见积性函数的贝尔级数：

- ϵ： 1
- 1： $\frac{1}{1-x}$
- id： $\frac{1}{1-px}$
- d： $\frac{1}{(1-x)^2}$
- σ： $\frac{1}{(1-x)(1-px)}$
- φ： $\frac{1-x}{1-px}$
- μ： $1-x$

狄利克雷卷积

定义两个数论函数 f, g 的狄利克雷卷积为

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

狄利克雷卷积

定义两个数论函数 f,g 的狄利克雷卷积为

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

一些性质：

狄利克雷卷积

定义两个数论函数 f, g 的狄利克雷卷积为

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

一些性质：

- 狄利克雷卷积满足交换律，结合律，分配律。

狄利克雷卷积

定义两个数论函数 f, g 的狄利克雷卷积为

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

一些性质：

- 狄利克雷卷积满足交换律，结合律，分配律。
- 两个积性函数的狄利克雷卷积也是积性函数。

狄利克雷卷积

定义两个数论函数 f, g 的狄利克雷卷积为

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

一些性质：

- 狄利克雷卷积满足交换律，结合律，分配律。
- 两个积性函数的狄利克雷卷积也是积性函数。
- 对于积性函数 f, g 和完全积性函数 h，$(f * g) \cdot h = (f \cdot h) * (g \cdot h)$。

逆函数

$\epsilon = f * g$，则称 g 为 f 的逆函数，记为 $g = f^{-1}$。

逆函数

$\epsilon = f * g$，则称 g 为 f 的逆函数，记为 $g = f^{-1}$。

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

$$g(n) = \frac{1}{f(1)}(\epsilon(n) - \sum_{d|n, d<n} g(d) f(\frac{n}{d}))$$

逆函数

$\epsilon = f * g$，则称 g 为 f 的逆函数，记为 $g = f^{-1}$。

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

$$g(n) = \frac{1}{f(1)}(\epsilon(n) - \sum_{d|n, d<n} g(d) f(\frac{n}{d}))$$

可见逆函数唯一，且有逆函数的充要条件是 $f(1) \neq 0$。

逆函数

$\epsilon = f * g$，则称 g 为 f 的逆函数，记为 $g = f^{-1}$。

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

$$g(n) = \frac{1}{f(1)}(\epsilon(n) - \sum_{d|n, d<n} g(d) f(\frac{n}{d}))$$

可见逆函数唯一，且有逆函数的充要条件是 $f(1) \neq 0$。
积性函数的逆函数也是积性函数。

逆函数

$\epsilon = f * g$，则称 g 为 f 的逆函数，记为 $g = f^{-1}$。

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

$$g(n) = \frac{1}{f(1)}(\epsilon(n) - \sum_{d|n, d<n} g(d) f(\frac{n}{d}))$$

可见逆函数唯一，且有逆函数的充要条件是 $f(1) \neq 0$。
积性函数的逆函数也是积性函数。
定义狄利克雷除法 $f/g = f * g^{-1}$。

逆函数

$\epsilon = f * g$，则称 g 为 f 的逆函数，记为 $g = f^{-1}$。

$$\epsilon(n) = \sum_{d|n} g(d) f(\frac{n}{d})$$
$$g(n) = \frac{1}{f(1)}(\epsilon(n) - \sum_{d|n, d<n} g(d) f(\frac{n}{d}))$$

可见逆函数唯一，且有逆函数的充要条件是 $f(1) \neq 0$。
积性函数的逆函数也是积性函数。
定义狄利克雷除法 $f/g = f * g^{-1}$。
1 函数的逆为 μ。

狄利克雷前缀和

给一个函数 f，求 $f * 1$。

狄利克雷前缀和

给一个函数 f，求 $f * 1$。
类似高维前缀和，把每个素数看做一维。对每个 $x = \prod_{i=1}^{k} p_i^{\alpha_i}$，将其看做高维空间内的一个点 $(\alpha_1, \alpha_2, \cdots, \alpha_k)$。
复杂度 $O(\sum_{i=1}^{k} \frac{n}{p_i}) = O(n \log \log n)$。

狄利克雷前缀和

给一个函数 f，求 $f*1$。
类似高维前缀和，把每个素数看做一维。对每个 $x=\prod_{i=1}^{k}p_i^{\alpha_i}$，将其看做高维空间内的一个点 $(\alpha_1,\alpha_2,\cdots,\alpha_k)$。
复杂度 $O(\sum_{i=1}^{k}\frac{n}{p_i})=O(n\log\log n)$。
求 $f*\mu$，即差分。一样可以 $O(n\log\log n)$。

狄利克雷前缀和

给一个函数 f，求 $f*1$。
类似高维前缀和，把每个素数看做一维。对每个 $x=\prod_{i=1}^{k}p_i^{\alpha_i}$，将其看做高维空间内的一个点 $(\alpha_1,\alpha_2,\cdots,\alpha_k)$。
复杂度 $O(\sum_{i=1}^{k}\frac{n}{p_i})=O(n\log\log n)$。
求 $f*\mu$，即差分。一样可以 $O(n\log\log n)$。
当然也有后缀和。

GYM101741 GCD

给定 $a_1, \cdots, a_n$ 与 k，求删除至多 k 个元素后最大的 gcd。
$n \le 10^5, k \le \frac{n}{2}, a_i \le 10^{18}$。

GYM101741 GCD

给定 $a_1, \cdots, a_n$ 与 k，求删除至多 k 个元素后最大的 $\gcd$。
$n \le 10^5, k \le \frac{n}{2}, a_i \le 10^{18}$。
随机化 trick。

GYM101741 GCD

给定 $a_1, \cdots, a_n$ 与 k，求删除至多 k 个元素后最大的 $\gcd$。
$n \le 10^5, k \le \frac{n}{2}, a_i \le 10^{18}$。
随机化 trick。
另类的质因数分解。

GYM101741 GCD

给定 $a_1, \cdots, a_n$ 与 k，求删除至多 k 个元素后最大的 $\gcd$。
$n \le 10^5, k \le \frac{n}{2}, a_i \le 10^{18}$。
随机化 trick。
另类的质因数分解。
狄利克雷前缀和。

积性函数卷积

对积性函数 f, g 求 $f * g$，复杂度 $O(n)$。

积性函数卷积

对积性函数 f, g 求 $f * g$，复杂度 $O(n)$。
对函数 f 与积性函数 g 求 $f * g$，复杂度 $O(n \log \log n)$。

数论分块

结论一

$$\forall a, b, c \in \mathbb{Z}, \lfloor \frac{a}{bc} \rfloor = \lfloor \frac{\lfloor \frac{a}{b} \rfloor}{c} \rfloor$$

数论分块

结论一

$$\forall a, b, c \in \mathbb{Z}, \lfloor \frac{a}{bc} \rfloor = \lfloor \frac{\lfloor \frac{a}{b} \rfloor}{c} \rfloor$$

结论二

$$\forall n \in \mathbb{N}_+, |\{\lfloor \frac{n}{d} \rfloor | d \in \mathbb{N}_+, d \leq n\}| \leq \lfloor 2\sqrt{n} \rfloor$$

数论分块

结论一

$$\forall a,b,c \in \mathbb{Z}, \lfloor \frac{a}{bc} \rfloor = \lfloor \frac{\lfloor \frac{a}{b} \rfloor}{c} \rfloor$$

结论二

$$\forall n \in \mathbb{N}_+, |\{\lfloor \frac{n}{d} \rfloor | d \in \mathbb{N}_+, d \le n\}| \le \lfloor 2\sqrt{n} \rfloor$$

结论三

将 $\lfloor \frac{n}{i} \rfloor$ 相同的 i 分为一块，则 i 所在块的右端点为 $\left\lfloor \frac{n}{\lfloor \frac{n}{i} \rfloor} \right\rfloor$。

P2260 模积和

求 $\sum_{i=1}^{n}\sum_{j=1}^{m}(n \bmod i)(m \bmod j), i \neq j$。$n, m \leq 10^9$。

逆元关系

μ 函数有一个重要性质：$1 * \mu = \epsilon$。

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

逆元关系

μ 函数有一个重要性质：$1 * \mu = \epsilon$。

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

证明

将 n 唯一分解，$n = \prod_{i=1}^{k} p_i^{\alpha_i}$，则

$$\sum_{d|n} \mu(d) = \sum_{i=0}^{k} (-1)^i \binom{k}{i} = [k = 0] = [n = 1]$$

莫比乌斯反演

$f=g*1$，可以推出 $g=f*\mu$。

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

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

莫比乌斯反演

$f = g * 1$，可以推出 $g = f * \mu$。

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

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

还有另一个方向

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

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

gcd 卷积

给定数列 a, b，求数列 f，满足 $f(n) = \sum_{\gcd(i,j)=n} a_i b_j$。

gcd 卷积

给定数列 a,b，求数列 f，满足 $f(n)=\sum_{\gcd(i,j)=n} a_i b_j$。
$g(n)=\sum_{n|\gcd(i,j)} a_i b_j$ 可以 $O(n\log\log n)$ 求出。

gcd 卷积

给定数列 a, b，求数列 f，满足 $f(n) = \sum_{\gcd(i,j)=n} a_i b_j$。
$g(n) = \sum_{n|\gcd(i,j)} a_i b_j$ 可以 $O(n \log\log n)$ 求出。
$g(n) = \sum_{n|d} f(d)$，$O(n \log\log n)$ 反演即得 f。

gcd 求和

引例

$$\sum_{i=1}^{n}\sum_{j=1}^{m}[\gcd(i,j)=1]$$

gcd 求和

引例

$$\sum_{i=1}^{n}\sum_{j=1}^{m}[\gcd(i,j)=1]$$

莫比乌斯反演可以处理这类与 gcd 求和相关的问题。

gcd 求和

利用 $\sum_{d|n}\mu(d)=[n=1]$，进行如下变换

$$
\begin{aligned}
&\sum_{i=1}^{n}\sum_{j=1}^{m}[\gcd(i,j)=1] \\
=&\sum_{i=1}^{n}\sum_{j=1}^{m}\sum_{d|i,j}\mu(d) \\
=&\sum_{d=1}^{n}\mu(d)\sum_{i=1}^{n}\sum_{j=1}^{m}[d|i][d|j] \\
=&\sum_{d=1}^{n}\mu(d)\lfloor\frac{n}{d}\rfloor\lfloor\frac{m}{d}\rfloor
\end{aligned}
$$

gcd 求和

一般形式：

$$\begin{aligned}
&\sum_{i=1}^{n}\sum_{j=1}^{m} f(\gcd(i,j)) \\
=&\sum_{i=1}^{n}\sum_{j=1}^{m}(f*\mu*1)(\gcd(i,j)) \\
=&\sum_{i=1}^{n}\sum_{j=1}^{m}\sum_{d|i,j}(f*\mu)(d) \\
=&\sum_{d=1}^{n}(f*\mu)(d)\lfloor\frac{n}{d}\rfloor\lfloor\frac{m}{d}\rfloor
\end{aligned}$$

gcd 求和

更一般的，如果 f,g 为完全积性函数：

$$\begin{aligned}
&\sum_{i=1}^{n}\sum_{j=1}^{m} f(i)g(j)h(\gcd(i,j)) \\
=&\sum_{i=1}^{n}\sum_{j=1}^{m} f(i)g(j)(h*\mu*1)(\gcd(i,j)) \\
=&\sum_{d=1}^{n}(h*\mu)(d)\sum_{i=1}^{\lfloor n/d\rfloor}\sum_{j=1}^{\lfloor m/d\rfloor} f(id)g(jd) \\
=&\sum_{d=1}^{n}(h*\mu)(d)f(d)g(d)\sum_{i=1}^{\lfloor n/d\rfloor}\sum_{j=1}^{\lfloor m/d\rfloor} f(i)g(i)
\end{aligned}$$

P2257 YY 的 GCD

求 $\sum_{i=1}^{n}\sum_{j=1}^{m}[\gcd(i,j)\in\mathbb{P}]$，$1\le n,m\le 10^7$。

P2257 YY 的 GCD

求 $\sum_{i=1}^{n}\sum_{j=1}^{m}[\gcd(i,j)\in\mathbb{P}]$，$1\le n,m\le 10^7$。
$f(n)=[n\in\mathbb{P}]$，需要求 $f*\mu$。

P2257 YY 的 GCD

求 $\sum_{i=1}^{n}\sum_{j=1}^{m}[\gcd(i,j)\in\mathbb{P}]$，$1\le n,m\le 10^7$。
$f(n)=[n\in\mathbb{P}]$，需要求 $f*\mu$。
虽然 f 不是积性函数，但也可以线筛。

P6156 简单题

求 $\sum_{i=1}^{n}\sum_{j=1}^{n}(i+j)^k\mu^2(\gcd(i,j))\gcd(i,j)$。
$n \le 5\times 10^6, k \le 10^{18}$。

P6271 一个人的数论

求 $\sum_{i=1}^{n} i^k[\gcd(i,n)=1]$。$n=\prod_{i=1}^{w} p_i^{\alpha_i}$。
$k \le 100, w \le 10^3, p_i, w_i \le 10^9$。

P6271 一个人的数论

求 $\sum_{i=1}^{n} i^k[\gcd(i,n)=1]$。$n=\prod_{i=1}^{w} p_i^{\alpha_i}$。
$k \le 100, w \le 10^3, p_i, w_i \le 10^9$。
拉插。

P3704 数字表格

f 表示斐波那契数列，求 $\prod_{i=1}^{n}\prod_{j=1}^{m} f(\gcd(i,j))$。多测。
$T \le 10^3, n, m \le 10^6$。

块筛

记 $S_f(n) = \sum_{i=1}^{n} f(i)$，求出

$$\{(\lfloor \frac{n}{i} \rfloor, S_f(\lfloor \frac{n}{i} \rfloor)) | i \in [1, n] \cap \mathbb{Z}\}$$

块筛

记 $S_f(n)=\sum_{i=1}^{n} f(i)$，求出

$$\{(\lfloor\frac{n}{i}\rfloor, S_f(\lfloor\frac{n}{i}\rfloor))|i\in[1,n]\cap\mathbb{Z}\}$$

集合大小是 $O(\sqrt{n})$ 的。

杜教筛

杜教筛核心思想在于寻找 $f = g/h$，满足 g, h 可块筛，则可以在 $O(n^{2/3})$ 复杂度内块筛 f。

杜教筛

杜教筛核心思想在于寻找 $f = g/h$，满足 g, h 可块筛，则可以在 $O(n^{2/3})$ 复杂度内块筛 f。

$$S_g(n) = \sum_{i=1}^{n} \sum_{d|i} h(d) f(\frac{i}{d})$$

$$S_g(n) = \sum_{d=1}^{n} h(d) S_f(\lfloor \frac{n}{d} \rfloor)$$

$$= S_f(n)h(1) + \sum_{d=2}^{n} h(d) S_f(\lfloor \frac{n}{d} \rfloor)$$

$$S_f(n) = \frac{1}{h(1)}(S_g(n) - \sum_{d=2}^{n} h(d) S_f(\lfloor \frac{n}{d} \rfloor))$$

杜教筛

分析复杂度。在每个 n 处复杂度都是 $O(\sqrt{n})$，则总复杂度为

$$
\begin{aligned}
&O(\sum_{i=1}^{\sqrt{n}} \sqrt{i} + \sqrt{n/i}) \\
=&O(\int_{1}^{n^{1/2}} x^{1/2}\mathrm{d}x + n^{1/2}\int_{1}^{n^{1/2}} x^{-1/2}\mathrm{d}x) \\
=&O(n^{3/4})
\end{aligned}
$$

杜教筛

分析复杂度。在每个 n 处复杂度都是 $O(\sqrt{n})$，则总复杂度为

$$
\begin{aligned}
&O(\sum_{i=1}^{\sqrt{n}} \sqrt{i} + \sqrt{n/i}) \\
=&O(\int_1^{n^{1/2}} x^{1/2}\mathrm{d}x + n^{1/2}\int_1^{n^{1/2}} x^{-1/2}\mathrm{d}x) \\
=&O(n^{3/4})
\end{aligned}
$$

如果预处理（线筛）出前 $n^{2/3}$ 项，则复杂度降为 $O(n^{2/3})$。

积式

$f = g * h$，求 $S_f(n)$。

积式

$f = g * h$，求 $S_f(n)$。

$$S_f(n) = \sum_{i=1}^{n} \sum_{d|i} g(d) h(\frac{i}{d})$$

$$S_f(n) = \sum_{d=1}^{n} g(d) S_h(\lfloor \frac{n}{d} \rfloor)$$

如果已知 g, h 的块筛，则对 S_f 求单点 $O(\sqrt{n})$，求块筛同样可以预处理前 $n^{2/3}$ 项达到 $O(n^{2/3})$。

练习

- φ
- μ
- σ_k
- $\varphi \cdot id_k$
- $\varphi \cdot \mu$

P4318 完全平方数

题意

$\mu^2(n) = \mu(n) \cdot \mu(n)$。求 $S_{\mu^2}(n)$。

P4318 完全平方数

题意

$\mu^2(n) = \mu(n) \cdot \mu(n)$。求 $S_{\mu^2}(n)$。

n 可以被唯一分解为 $x^2 y$，其中 $\mu^2(y) = 1$。

P4318 完全平方数

题意

$\mu^2(n) = \mu(n) \cdot \mu(n)$。求 $S_{\mu^2}(n)$。

n 可以被唯一分解为 x^2y，其中 $\mu^2(y) = 1$。
令 $f(n) = [\exists d, d^2 = n]$，则 $1 = f * \mu^2$。

P4318 完全平方数

题意

$\mu^2(n) = \mu(n) \cdot \mu(n)$。求 $S_{\mu^2}(n)$。

n 可以被唯一分解为 $x^2 y$，其中 $\mu^2(y) = 1$。
令 $f(n) = [\exists d, d^2 = n]$，则 $1 = f * \mu^2$。
$S_f(n)$ 可以 $O(1)$ 计算。杜教筛即可。复杂度 $O(n^{2/3})$。

P4318 完全平方数

更优秀的做法：

P4318 完全平方数

更优秀的做法：

$$\mu^2(n) = [x = 1] = \sum_{d|x} \mu(d) = \sum_{d^2|n} \mu(d)$$

$$S_{\mu^2}(n) = \sum_{i=1}^{n} \sum_{d^2|i} \mu(d) = \sum_{d=1}^{\sqrt{n}} \mu(d) \lfloor \frac{n}{d^2} \rfloor$$

复杂度 $O(\sqrt{n})$。

? ? ?

将 n 质因数分解为 $\prod_{i=1}^{k} p_i^{\alpha_i}$，$f(n) = (-1)^{\sum_{i=1}^{k} \alpha_i}$。杜教筛 S_f。

? ? ?

将 n 质因数分解为 $\prod_{i=1}^{k} p_i^{\alpha_i}$，$f(n) = (-1)^{\sum_{i=1}^{k} \alpha_i}$。杜教筛 S_f。
显然 f 为积性函数。

？？？

将 n 质因数分解为 $\prod_{i=1}^{k} p_i^{\alpha_i}$，$f(n) = (-1)^{\sum_{i=1}^{k} \alpha_i}$。杜教筛 S_f。
显然 f 为积性函数。
f 的贝尔级数为 $\sum_{i\geq 0}(-1)^i x^i = \frac{1}{1+x}$，于是 $f = \epsilon/\mu^2$。

51nod1220 约数之和

求 $\sum_{i=1}^{n}\sum_{j=1}^{n}\sigma(ij)$。$n \leq 10^9$。

51nod1220 约数之和

求 $\sum_{i=1}^{n}\sum_{j=1}^{n}\sigma(ij)$。$n \le 10^9$。

$$\sigma(xy)=\sum_{i|x}\sum_{j|y} ij[\gcd(\frac{x}{i},j)=1]$$

51nod1220 约数之和

求 $\sum_{i=1}^{n}\sum_{j=1}^{n}\sigma(ij)$。$n\le 10^9$。

$$\sigma(xy)=\sum_{i|x}\sum_{j|y} ij[\gcd(\frac{x}{i},j)=1]$$

$$\begin{aligned}
ans &= \sum_{x=1}^{n}\sum_{y=1}^{n}\sum_{i|x}\sum_{j|y}\frac{x}{i}\cdot j\sum_{d|i,j}\mu(d)\\
&= \sum_{d=1}^{n}\mu(d)(\sum_{d|i}\sum_{i|x}\frac{x}{i})(\sum_{d|j}\sum_{j|y}j)\\
&= \sum_{d=1}^{n}\mu(d)dS_{\sigma}^{2}(\lfloor\frac{n}{d}\rfloor)
\end{aligned}$$

P1829 Crash的数字表格（加强）

求 $\sum_{i=1}^{n}\sum_{j=1}^{m}\operatorname{lcm}(i,j)$。$n,m\le 10^{10}$。

P1587 循环之美

求 $\sum_{i=1}^{n}\sum_{j=1}^{m}[\gcd(i,j)=1][\gcd(j,k)=1]$。
$n,m \le 10^9, k \le 2000$。

SP20173 DIVCNT2

求 $\sum_{i=1}^{n} d(i^2)$。$n \le 10^{11}$。

Powerful Number

将 n 质因数分解，若每个质数的指数均大于等于 2，则称 n 为 Powerful Number。

Powerful Number

将 n 质因数分解，若每个质数的指数均大于等于 2，则称 n 为 Powerful Number。

结论

n 以内的 PN 不超过 $O(\sqrt{n})$。

Powerful Number

将 n 质因数分解，若每个质数的指数均大于等于 2，则称 n 为 Powerful Number。

结论

n 以内的 PN 不超过 $O(\sqrt{n})$。

证明

一个 PN 可以被分解为 a^2b^3 的形式。

$$O(\sum_{a=1}^{\sqrt{n}} \sqrt[3]{\frac{n}{a^2}}) = O(n^{1/3} \int_1^{n^{1/2}} x^{-2/3}\mathrm{d}x) = O(\sqrt{n})$$

素数拟合

给一个积性函数 f，求 $S_f(n)$。

素数拟合

给一个积性函数 f，求 $S_f(n)$。
可以构造一个易块筛的积性函数 g，满足 g 在所有素数处都与 f 相等。

素数拟合

给一个积性函数 f，求 $S_f(n)$。
可以构造一个易块筛的积性函数 g，满足 g 在所有素数处都与 f 相等。
求 $h = f/g$，此时 h 也是积性函数。

素数拟合

给一个积性函数 f，求 $S_f(n)$。
可以构造一个易块筛的积性函数 g，满足 g 在所有素数处都与 f 相等。
求 $h = f/g$，此时 h 也是积性函数。
$f(p) = g(p)h(1) + g(1)h(p)$，因为 $g(p) = f(p)$，故 $h(p) = 0$。
于是 h 只在 PN 处有值。

素数拟合

给一个积性函数 f，求 $S_f(n)$。
可以构造一个易块筛的积性函数 g，满足 g 在所有素数处都与 f 相等。
求 $h = f/g$，此时 h 也是积性函数。
$f(p) = g(p)h(1) + g(1)h(p)$，因为 $g(p) = f(p)$，故 $h(p) = 0$。
于是 h 只在 PN 处有值。

$$S_f(n) = \sum_{d \in \mathrm{PN}} h(d) S_g(\lfloor \frac{n}{d} \rfloor)$$

素数拟合

接下来求 h。因为 h 有积性，故只需求出其在质数幂处的值。

素数拟合

接下来求 h。因为 h 有积性，故只需求出其在质数幂处的值。

$$f(p^k) = \sum_{i=0}^{k} h(p^i)g(p^{k-i})$$

$$h(p^k) = f(p^k) - \sum_{i=0}^{k-1} h(p^i)g(p^{k-i})$$

可以在找 PN 的过程中暴力求出。

素数拟合

接下来求 h。因为 h 有积性，故只需求出其在质数幂处的值。

$$f(p^k) = \sum_{i=0}^{k} h(p^i)g(p^{k-i})$$

$$h(p^k) = f(p^k) - \sum_{i=0}^{k-1} h(p^i)g(p^{k-i})$$

可以在找 PN 的过程中暴力求出。

这部分的复杂度为 $O(\sum_{p\in\mathbb{P},p\leq\sqrt{n}} \log_p^2(n)) = O(\frac{\sqrt{n}}{\log n})$，不为瓶颈。

素数拟合

一般复杂度瓶颈在于块筛 g。常为 $O(n^{2/3})$。

素数拟合

一般复杂度瓶颈在于块筛 g。常为 $O(n^{2/3})$。
特别地，如果 $S_g(n)$ 能在 $O(\sqrt{n})$ 复杂度内求出，则复杂度优化至

$$O(\sum_{d\in \mathrm{PN}}\sqrt{\frac{n}{d}}) = O(\sum_{a,b}\sqrt{\frac{n}{a^2b^3}}) = O(n^{1/2}\sum_{a}a^{-1}) = O(\sqrt{n}\log n)$$

P5325 【模板】Min_25 筛

定义积性函数 f，$f(p^k) = p^k(p^k - 1)$。求 $S_f(n)$。$n \le 10^{10}$。

P5325 【模板】Min_25 筛

定义积性函数 f，$f(p^k) = p^k(p^k - 1)$。求 $S_f(n)$。$n \leq 10^{10}$。
$g = \mathsf{id} \cdot \varphi$。则 g 与 f 素数拟合。
$\mathsf{id} \cdot \varphi = \mathsf{id} \cdot (\mathsf{id}/1) = \mathsf{id}_2/\mathsf{id}$。容易块筛。

? ? ?

定义积性函数 f，$f(p^c) = p^{ck_1} + p^{ck_2}$。求 $S_f(n)$。
$n \leq 10^{12}, k_1, k_2 \leq 10$。

？？？

定义积性函数 f，$f(p^c) = p^{ck_1} + p^{ck_2}$。求 $S_f(n)$。
$n \le 10^{12}, k_1, k_2 \le 10$。
$g = \mathrm{id}_{k_1} * \mathrm{id}_{k_2}$。则 g 与 f 素数拟合。

? ? ?

定义积性函数 f，$f(p^c) = p^{ck_1} + p^{ck_2}$。求 $S_f(n)$。
$n \le 10^{12}, k_1, k_2 \le 10$。
$g = \mathrm{id}_{k_1} * \mathrm{id}_{k_2}$。则 g 与 f 素数拟合。

$$S_g(n) = \sum_{d=1}^{n} \mathrm{id}_{k_1}(d) S_{\mathrm{id}_{k_2}}(\lfloor \frac{n}{d} \rfloor)$$

? ? ?

定义积性函数 f，$f(p^c) = p^{ck_1} + p^{ck_2}$。求 $S_f(n)$。
$n \le 10^{12}, k_1, k_2 \le 10$。
$g = \mathrm{id}_{k_1} * \mathrm{id}_{k_2}$。则 g 与 f 素数拟合。

$$S_g(n) = \sum_{d=1}^{n} \mathrm{id}_{k_1}(d) S_{\mathrm{id}_{k_2}}(\lfloor \frac{n}{d} \rfloor)$$

利用拉格朗日插值可以 $O(k)$ 求出 id_k 的前缀和，于是可以 $O((k_1+k_2)\sqrt{n})$ 求出两个 id 函数的块筛。

? ? ?

定义积性函数 f，$f(p^c) = p^{ck_1} + p^{ck_2}$。求 $S_f(n)$。
$n \le 10^{12}, k_1, k_2 \le 10$。
$g = \mathrm{id}_{k_1} * \mathrm{id}_{k_2}$。则 g 与 f 素数拟合。

$$S_g(n) = \sum_{d=1}^{n} \mathrm{id}_{k_1}(d) S_{\mathrm{id}_{k_2}}(\lfloor \frac{n}{d} \rfloor)$$

利用拉格朗日插值可以 $O(k)$ 求出 id_k 的前缀和，于是可以 $O((k_1 + k_2)\sqrt{n})$ 求出两个 id 函数的块筛。
求 $S_g(n)$ 的复杂度是 $O(\sqrt{n})$ 的。故总复杂度为 $O(\sqrt{n}(k_1 + k_2 + \log n))$。

SP20174 DIVCNT3

求 $\sum_{i=1}^{n} d(i^3)$。$n \le 10^{11}$。