数论

harryzhr

2025 年 2 月 7 日

Preface

数论包含的知识非常多，~~但是大部分都没啥用~~
所以下面就挑一些~~有用的~~有意思的来讲了

辗转相除法

更相减损术

$$\gcd(a, b) = \gcd(a - b, b)$$

欧几里得算法

$$\gcd(a, b) = \gcd(b, a \bmod b)$$

注意到当 $a > b$ 时 $a \bmod b < \frac{1}{2}a$，所以至多递归 $O(\log a)$ 层。

扩展欧几里得算法

扩展欧几里得算法

用于求不定方程

$$ax + by = d$$

的一组整数解，该方程有解当且仅当 $\gcd(a, b) \mid d$，下面假设 $d = \gcd(a, b)$。

设

$$ax + by = \gcd(a, b)$$
$$bx_1 + (a \bmod b)y_1 = \gcd(b, a \bmod b)$$

假设我们已经通过递归求出 x_1 和 y_1 则

$$x = y_1, \ \ y = x_1 - \left\lfloor \frac{a}{b} \right\rfloor y_1$$

递归边界是 $b = 0$ 时，此时取 $x = 1, y = 0$ 即可。

扩展欧几里得算法的推导

由欧几里得定理可知：$\gcd(a,b)=\gcd(b,a \bmod b)$
所以 $ax+by=bx_1+(a \bmod b)y_1$
又因为

$$a \bmod b = a-\left(\left\lfloor\frac{a}{b}\right\rfloor\times b\right)$$

所以

$$ax+by=bx_1+\left(a-\left(\left\lfloor\frac{a}{b}\right\rfloor\times b\right)\right)y_1$$

整理得

$$ax+by=ay_1+bx_1-\left\lfloor\frac{a}{b}\right\rfloor\times by_1=ay_1+b\left(x_1-\left\lfloor\frac{a}{b}\right\rfloor y_1\right)$$

对比两侧系数即得到

$$x=y_1,\quad y=x_1-\left\lfloor\frac{a}{b}\right\rfloor y_1$$

类欧几里得算法

洛谷 P5170 【模板】类欧几里得算法

给定 n, a, b, c ，分别求

$$\sum_{i=0}^{n}\left\lfloor\frac{ai+b}{c}\right\rfloor,\sum_{i=0}^{n}\left\lfloor\frac{ai+b}{c}\right\rfloor^{2},\sum_{i=0}^{n}i\left\lfloor\frac{ai+b}{c}\right\rfloor$$

答案对 998244353 取模。多组数据。
$1\leqslant T\leqslant 10^5,\ 0\leqslant n,\ a,\ b,\ c\leqslant 10^9,\ c\neq 0$ 。

令

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

- $a\geq c$ 或者 $b\geq c$，将 a,b 对 c 取模

$$\begin{aligned}
f(a,b,c,n)&=\sum_{i=0}^{n}\left\lfloor\frac{ai+b}{c}\right\rfloor\\
&=\sum_{i=0}^{n}\left\lfloor\frac{\left(\left\lfloor\frac{a}{c}\right\rfloor c+a\bmod c\right)i+\left(\left\lfloor\frac{b}{c}\right\rfloor c+b\bmod c\right)}{c}\right\rfloor\\
&=\frac{n(n+1)}{2}\left\lfloor\frac{a}{c}\right\rfloor+(n+1)\left\lfloor\frac{b}{c}\right\rfloor+\sum_{i=0}^{n}\left\lfloor\frac{(a\bmod c)i+(b\bmod c)}{c}\right\rfloor\\
&=\frac{n(n+1)}{2}\left\lfloor\frac{a}{c}\right\rfloor+(n+1)\left\lfloor\frac{b}{c}\right\rfloor+f(a\bmod c,b\bmod c,c,n)
\end{aligned}$$

现在只需考虑 $a<c$ 且 $b<c$ 的情况。

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

- $a<c$ 且 $b<c$

把 $\left\lfloor\frac{ai+b}{c}\right\rfloor$ 用求和式展开

$$\sum_{i=0}^{n}\left\lfloor\frac{ai+b}{c}\right\rfloor=\sum_{i=0}^{n}\sum_{j=0}^{\left\lfloor\frac{ai+b}{c}\right\rfloor-1}1$$

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

- $a<c$ 且 $b<c$

把 $\left\lfloor\frac{ai+b}{c}\right\rfloor$ 用求和式展开

$$\sum_{i=0}^{n}\left\lfloor\frac{ai+b}{c}\right\rfloor=\sum_{i=0}^{n}\sum_{j=0}^{\left\lfloor\frac{ai+b}{c}\right\rfloor-1}1$$

交换和式

$$=\sum_{j=0}^{\left\lfloor\frac{an+b}{c}\right\rfloor-1}\sum_{i=0}^{n}\left[j<\left\lfloor\frac{ai+b}{c}\right\rfloor\right]$$

把 $j < \left\lfloor \frac{ai+b}{c} \right\rfloor$ 这个条件转化一下，

$$
\begin{aligned}
j < \left\lfloor \frac{ai+b}{c} \right\rfloor &\iff j+1 \le \left\lfloor \frac{ai+b}{c} \right\rfloor \iff j+1 \le \frac{ai+b}{c} \\
&\iff jc+c \le ai+b \iff jc+c-b-1 < ai \\
&\iff \left\lfloor \frac{jc+c-b-1}{a} \right\rfloor < i
\end{aligned}
$$

令 $m = \left\lfloor \frac{an+b}{c} \right\rfloor$，那么原式化为

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

递归即可。此时 c 和 a 交换了位置，复杂度同欧几里得算法，为 $O(\log n)$。

模板题

我们还需要求

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

$$h(a,b,c,n)=\sum_{i=0}^{n}\left\lfloor\frac{ai+b}{c}\right\rfloor^2$$

推导 g

- $a \geq c$ 或 $b \geq c$

$$g(a,b,c,n) = g(a \bmod c, b \bmod c, c, n) + \left\lfloor \frac{a}{c} \right\rfloor \frac{n(n+1)(2n+1)}{6} + \left\lfloor \frac{b}{c} \right\rfloor \frac{n(n+1)}{2}$$

- $a < c$ 且 $b < c$

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

$$g(a,b,c,n) = \sum_{i=0}^{n} i \left\lfloor \frac{ai+b}{c} \right\rfloor = \sum_{j=0}^{m-1} \sum_{i=0}^{n} \left[j < \left\lfloor \frac{ai+b}{c} \right\rfloor \right] \cdot i$$

方便起见，设 $t = \left\lfloor \frac{jc+c-b-1}{a} \right\rfloor$，

$$= \sum_{j=0}^{m-1} \sum_{i=0}^{n} [i > t] \cdot i = \sum_{j=0}^{m-1} \frac{1}{2}(t+n+1)(n-t) = \frac{1}{2}\left[mn(n+1) - \sum_{j=0}^{m-1} t^2 - \sum_{j=0}^{m-1} t \right]$$

$$= \frac{1}{2}[mn(n+1) - h(c, c-b-1, a, m-1) - f(c, c-b-1, a, m-1)]$$

推导 h

- $a \geq c$ 或 $b \geq c$

$$h(a,b,c,n)=\sum_{i=0}^{n}\left\lfloor\frac{ai+b}{c}\right\rfloor^2=\sum_{i=0}^{n}\left(\left\lfloor\frac{a}{c}\right\rfloor i+\left\lfloor\frac{b}{c}\right\rfloor+\left\lfloor\frac{a \bmod c\times i+b \bmod c}{c}\right\rfloor\right)^2$$

$$=\sum_{i=0}^{n}\left(\left\lfloor\frac{a}{c}\right\rfloor^2 i^2+\left\lfloor\frac{b}{c}\right\rfloor^2+\left\lfloor\frac{a \bmod c\times i+b \bmod c}{c}\right\rfloor^2+2\left\lfloor\frac{a}{c}\right\rfloor\left\lfloor\frac{b}{c}\right\rfloor i\right.$$

$$\left.+2\left\lfloor\frac{a}{c}\right\rfloor\left\lfloor\frac{a \bmod c\times i+b \bmod c}{c}\right\rfloor i+2\left\lfloor\frac{b}{c}\right\rfloor\left\lfloor\frac{a \bmod c\times i+b \bmod c}{c}\right\rfloor\right)$$

$$=\frac{n(n+1)(2n+1)}{6}\left\lfloor\frac{a}{c}\right\rfloor^2+(n+1)\left\lfloor\frac{b}{c}\right\rfloor^2+n(n+1)\left\lfloor\frac{a}{c}\right\rfloor\left\lfloor\frac{b}{c}\right\rfloor$$

$$+2\left\lfloor\frac{a}{c}\right\rfloor g(a \bmod c,b \bmod c,c,n)+2\left\lfloor\frac{b}{c}\right\rfloor f(a \bmod c,b \bmod c,c,n)$$

$$+h(a \bmod c,b \bmod c,c,n)$$

- $a < c$ 且 $b < c$

令 $m = \left\lfloor \frac{an+b}{c} \right\rfloor$，$t = \left\lfloor \frac{cj+c-b-1}{a} \right\rfloor$

$$h(a,b,c,n) = \sum_{i=0}^{n} \left\lfloor \frac{ai+b}{c} \right\rfloor^2$$

将 n^2 写成 $(2\sum_{i=1}^{n} i) - n$

$$h(a,b,c,n) = \sum_{i=0}^{n} \left(\left(2 \sum_{j=1}^{\left\lfloor \frac{ai+b}{c} \right\rfloor} j \right) - \left\lfloor \frac{ai+b}{c} \right\rfloor \right)$$

$$= \left(2 \sum_{i=0}^{n} \sum_{j=0}^{\left\lfloor \frac{ai+b}{c} \right\rfloor - 1} j + 1 \right) - f(a,b,c,n)$$

$$= \left(2 \sum_{j=0}^{m-1} \sum_{i=0}^{n} (j+1) \left[j < \left\lfloor \frac{ai+b}{c} \right\rfloor \right] \right) - f(a,b,c,n)$$

其中

$$\sum_{j=0}^{m-1}\sum_{i=0}^{n}(j+1)\left[j<\left\lfloor\frac{ai+b}{c}\right\rfloor\right]=\sum_{j=1}^{m-1}(j+1)\sum_{i=0}^{n}\left[j<\left\lfloor\frac{ai+b}{c}\right\rfloor\right]$$

$$=\sum_{j=0}^{m-1}(j+1)\sum_{i=0}^{n}[i>t]=\sum_{j=0}^{m-1}(j+1)(n-t)$$

$$=\frac{m(m+1)}{2}n-f(c,c-b-1,a,m-1)-g(c,c-b-1,a,m-1)$$

所以 $h(a,b,c,n)=$

$$m(m+1)n-2f(c,c-b-1,a,m-1)-2g(c,c-b-1,a,m-1)-f(a,b,c,n)$$

其中

$$\sum_{j=0}^{m-1}\sum_{i=0}^{n}(j+1)\left[j<\left\lfloor\frac{ai+b}{c}\right\rfloor\right]=\sum_{j=1}^{m-1}(j+1)\sum_{i=0}^{n}\left[j<\left\lfloor\frac{ai+b}{c}\right\rfloor\right]$$

$$=\sum_{j=0}^{m-1}(j+1)\sum_{i=0}^{n}[i>t]=\sum_{j=0}^{m-1}(j+1)(n-t)$$

$$=\frac{m(m+1)}{2}n-f(c,c-b-1,a,m-1)-g(c,c-b-1,a,m-1)$$

所以 $h(a,b,c,n)=$

$$m(m+1)n-2f(c,c-b-1,a,m-1)-2g(c,c-b-1,a,m-1)-f(a,b,c,n)$$

f,g,h 相互递归调用即可。

欧拉定理及其扩展

欧拉定理

若 $\gcd(a,p)=1$，则

$$a^c \equiv a^{c \bmod \varphi(p)} \pmod p$$

费马小定理即欧拉定理在 p 为质数时的特殊情况

欧拉定理及其扩展

欧拉定理

若 $\gcd(a,p)=1$，则

$$a^c \equiv a^{c \bmod \varphi(p)} \pmod p$$

费马小定理即欧拉定理在 p 为质数时的特殊情况

扩展欧拉定理

$$a^c \equiv \begin{cases} a^{c \bmod \varphi(p)} & \gcd(a,m)=1 \\ a^c & \gcd(a,m)\neq 1 \land c<\varphi(p) \\ a^{c \bmod \varphi(p)+\varphi(p)} & \gcd(a,m)\neq 1 \land c\geq \varphi(p) \end{cases} \pmod p$$

例题

LOJ525. 多项式

给定 k，求出一个次数不超过 6×10^4，最高次项非 0 的多项式 $f(x)$，使得
$\forall i \in [0, k-1], f(i) \bmod k = 0$
$k \leq 3 \times 10^4$

例题

LOJ525. 多项式

给定 k，求出一个次数不超过 6×10^4，最高次项非 0 的多项式 $f(x)$，使得
$\forall i\in[0,k-1], f(i) \bmod k = 0$
$k\le 3\times 10^4$

根据扩展欧拉定理，$x^{\varphi}(k)\equiv x^{2\varphi(k)} \bmod k$
构造多项式 $x^{2\varphi(k)}-x^{\varphi(k)}$ 即可

例题

「六省联考 2017」相逢是问候

给定长为 n 的初始序列 a，以及常数 c, p。m 次操作：

- 将一个区间内的所有数替换为 c^{a_i}
- 区间求和 $\bmod p$

$n, m \leq 5 \times 10^4, p \leq 10^8$

例题

「六省联考 2017」相逢是问候

给定长为 n 的初始序列 a，以及常数 c, p。m 次操作：

- 将一个区间内的所有数替换为 c^{a_i}
- 区间求和 $\bmod\ p$

$n, m \le 5 \times 10^4, p \le 10^8$

记 $f(a, k, p)$ 表示 a 经过 k 次 $a = c^a$ 后 $\bmod\ p$ 的值。可以递归处理。
因为 $\varphi(p)$ 只会递归 $O(\log p)$ 层，所以这个值在 $k > 2\log p$ 时是不变的。

例题

「六省联考 2017」相逢是问候

给定长为 n 的初始序列 a，以及常数 c, p。m 次操作：

- 将一个区间内的所有数替换为 c^{a_i}
- 区间求和 $\bmod p$

$n, m \leq 5 \times 10^4, p \leq 10^8$

记 $f(a, k, p)$ 表示 a 经过 k 次 $a = c^a$ 后 $\bmod p$ 的值。可以递归处理。
因为 $\varphi(p)$ 只会递归 $O(\log p)$ 层，所以这个值在 $k > 2\log p$ 时是不变的。
所以对于没有操作 $2\log p$ 次的位置暴力修改，线段树维护：区间和，这个区间是否所有位置都修改过超过 $2\log p$ 次询问。
复杂度是 $O(n\log^2 p)$

前置知识

~~整除分块和莫比乌斯反演相信大家已经听了很多遍了，所以这里略过~~

- 积性函数
- Dirichlet 卷积
- 线性筛素数、线性筛 μ, φ

积性函数

常见的积性函数有

- 单位函数：$\varepsilon(n) = [n = 1]$（完全积性）
- 恒等函数：$\mathrm{id}_k(n) = n^k$（完全积性），当 $k = 1$ 时，简记为 $id(n) = n$
- 常数函数：$1(n) = 1$（完全积性）
- 除数函数：$\sigma_k(n) = \sum_{d|n} d^k$，$\sigma_0(n)$ 即因数个数，简记为 $\mathrm{d}(n)$；$\sigma_1(n)$ 即因数之和，简记为 $\sigma(n)$
- 欧拉函数：$\varphi(n) = \sum_{d=1}^{n} [\gcd(d, n) = 1]$
- 莫比乌斯函数：

$$\mu(x) = \begin{cases} 1 & (x = 1) \\ (-1)^k & (x \text{ 没有平方数因子，且 } x \text{ 的质因子个数为 } k) \\ 0 & (x \text{ 有平方数因子}) \end{cases}$$

数论分块（整除分块）

性质

$\left\lfloor \frac{n}{i} \right\rfloor$ 的不同的取值只有 $O(\sqrt{n})$ 种。

证明. 对于 $i \in [1, \sqrt{n}]$，取值最多 $\sqrt{n}$ 个；对于 $i \in [\sqrt{n}, n]$，$\left\lfloor \frac{n}{i} \right\rfloor \le \sqrt{n}$，取值最多 $\sqrt{n}$ 个。

数论分块（整除分块）

性质

$\left\lfloor \frac{n}{i} \right\rfloor$ 的不同的取值只有 $O(\sqrt{n})$ 种。

证明. 对于 $i \in [1, \sqrt{n}]$，取值最多 $\sqrt{n}$ 个；对于 $i \in [\sqrt{n}, n]$，$\left\lfloor \frac{n}{i} \right\rfloor \leq \sqrt{n}$，取值最多 $\sqrt{n}$ 个。

然后我们需要对于一个 i ，求出满足 $\left\lfloor \frac{n}{j} \right\rfloor = \left\lfloor \frac{n}{i} \right\rfloor$ 的最大的 j 是多少。设 $\left\lfloor \frac{n}{i} \right\rfloor = k$

$$\left\lfloor \frac{n}{j} \right\rfloor = k \implies k \leq \frac{n}{j} < k+1 \implies \frac{1}{k+1} < \frac{j}{n} \leq \frac{1}{k} \implies j \leq \frac{n}{k} \implies j \leq \left\lfloor \frac{n}{\left\lfloor \frac{n}{i} \right\rfloor} \right\rfloor$$

j 取这个 $\left\lfloor \frac{n}{\left\lfloor \frac{n}{i} \right\rfloor} \right\rfloor$ 即可

狄利克雷卷积的性质

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

证明：设 $n=\prod p_i^{k_i}, n'=\prod p_i$，$n$ 的质因数个数为 m

$$\sum_{d|n}\mu(d)=\sum_{d|n'}\mu(d)=\sum_{i=0}^{m}\binom{m}{i}(-1)^i=(1+(-1))^m$$

当且仅当 $m=0$ 即 $n=1$ 时 $\sum_{d|n}\mu(d)=1$

狄利克雷卷积的性质

$$\sum_{d|n}\varphi(d)=n \iff \varphi*1=\mathrm{id}$$

证明：因为 φ 是积性函数，只需要证明 $n=p^k$(p为质数) 的时候 $\sum_{d|n}\varphi(d)=n$ 成立即可。

$$\sum_{i=0}^{k}\varphi(p^i)=1+\sum_{i=1}^{k}(p-1)p^{i-1}=p^k$$

狄利克雷卷积的性质

$$\sum_{d|n}\varphi(d)=n \iff \varphi*1=\mathrm{id}$$

证明：因为 φ 是积性函数，只需要证明 $n=p^k$(p为质数) 的时候 $\sum_{d|n}\varphi(d)=n$ 成立即可。

$$\sum_{i=0}^{k}\varphi(p^i)=1+\sum_{i=1}^{k}(p-1)p^{i-1}=p^k$$

两边同时再卷上 μ，还可以得到

$$\varphi*1*\mu=\mathrm{id}*\mu \iff \varphi=\mathrm{id}*\mu \iff \varphi(n)=\sum_{d|n}d\mu(\frac{n}{d})$$

狄利克雷卷积的性质

$$d = 1 * 1 \iff d(n) = \sum_{d|n} 1$$

$$\sigma = \mathrm{id} * 1 \iff \sigma(n) = \sum_{d|n} d$$

这两个都可以用定义得到

莫比乌斯反演

$$g = f * 1$$

两边同时卷一个 μ

$$g * \mu = f * 1 * \mu = f * \epsilon = f$$

莫比乌斯反演

$$g = f * 1$$

两边同时卷一个 μ

$$g * \mu = f * 1 * \mu = f * \epsilon = f$$

$$g = f * 1 \iff g * \mu = f$$

用式子表示，就是

$$g(n) = \sum_{i|n} f(i) \iff f(n) = \sum_{i|n} \mu\left(\frac{n}{i}\right) g(i)$$

这个式子就是莫比乌斯反演

热身题

洛谷 P1829 [国家集训队]Crash 的数字表格 / JZPTAB

T 组数据，求

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

$n, m \leq 10^7, T \leq 10^4$

热身题

洛谷 P1829 [国家集训队]Crash 的数字表格 / JZPTAB

T 组数据，求

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

$n,m \le 10^7, T \le 10^4$

把 $\operatorname{lcm}(i,j)$ 转化成 $\frac{ij}{\gcd(i,j)} = \gcd(i,j) \times \frac{i}{\gcd(i,j)} \times \frac{j}{\gcd(i,j)}$，枚举 $\gcd(i,j)$

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

整除分块之后只用考虑后面 $f(n,m)=\sum_{i=1}^{n}\sum_{j=1}^{m}[\gcd(i,j)=1]ij$ 怎么求

记 $S(n)=\sum_{i=1}^{n} i=\frac{i(i+1)}{2}$

$$\begin{aligned} f(n,m) &= \sum_{i=1}^{n}\sum_{j=1}^{m} ij \sum_{k|\gcd(i,j)} \mu(k) \\ &= \sum_{k=1}^{n} \mu(k)k^2 \sum_{i=1}^{n/k}\sum_{j=1}^{m/k} ij \\ &= \sum_{k=1}^{n} \mu(k)k^2 S(\lfloor \frac{n}{k} \rfloor) S(\lfloor \frac{m}{k} \rfloor) \end{aligned}$$

预处理 $x^2\mu(x)$ 的前缀和即可整除分块
现在是一个整除分块套一个整除分块的形式，多组数据可能过不太了，继续优化

$$原式 = \sum_{d=1}^{n} d \sum_{k=1}^{n/d} \mu(k)k^2 S(\lfloor \frac{n}{dk} \rfloor) S(\lfloor \frac{m}{dk} \rfloor)$$

先枚举 $T = dk$

$$= \sum_{T=1}^{n} S(\lfloor \frac{n}{T} \rfloor) S(\lfloor \frac{m}{T} \rfloor) T \sum_{k|T} \mu(k)k$$

$$原式 = \sum_{d=1}^{n} d \sum_{k=1}^{n/d} \mu(k) k^2 S(\lfloor \frac{n}{dk} \rfloor) S(\lfloor \frac{m}{dk} \rfloor)$$

先枚举 $T = dk$

$$= \sum_{T=1}^{n} S(\lfloor \frac{n}{T} \rfloor) S(\lfloor \frac{m}{T} \rfloor) T \sum_{k|T} \mu(k) k$$

我们需要在线性复杂度内筛出 $T \sum_{k|T} \mu(k) k$ 的前缀和，也就是线性筛出 $f(n) = \sum_{i|n} \mu(i) i$，容易证明这个函数是积性的。

因为 $\mu(x)$ 在 x 有平方因子时答案为 0，所以如果 $f(i \times p)$ 在 $p|i$ 时，不会产生新的质因子，$f(i \times p) = f(i)$。

然后就是简单的整除分块了，复杂度 $O(n + T\sqrt{n})$

热身题

「SDOI2014」数表

有一张 $n \times m$ 的数表，其第 i 行第 j 列（$1 \leq i \leq n, 1 \leq j \leq m$）的数值为能同时整除 i 和 j 的所有自然数之和。给定 a，计算数表中不大于 a 的数之和。

$n, m \leq 10^5$，2×10^4 组数据

热身题

「SDOI2014」数表

有一张 $n \times m$ 的数表，其第 i 行第 j 列（$1 \le i \le n, 1 \le j \le m$）的数值为能同时整除 i 和 j 的所有自然数之和。给定 a，计算数表中不大于 a 的数之和。
$n, m \le 10^5$，2×10^4 组数据

$\sigma(x)$ 表示 x 的因数之和，如果不考虑 $\le a$ 的限制

$$
\begin{aligned}
ans &= \sum_{i=1}^{n} \sum_{j=1}^{m} \sigma(\gcd(i,j)) \\
&= \sum_{d=1}^{n} \sigma(d) \sum_{i=1}^{n/d} \sum_{j=1}^{m/d} [\gcd(i,j)=1] \\
&= \sum_{d=1}^{n} \sigma(d) \sum_{i=1}^{n/d} \mu(i) \left\lfloor \frac{n}{di} \right\rfloor \left\lfloor \frac{m}{di} \right\rfloor
\end{aligned}
$$

和上面相同的套路，枚举 $T = di$

$$ans = \sum_{T=1}^{n} \left\lfloor \frac{n}{T} \right\rfloor \left\lfloor \frac{m}{T} \right\rfloor \sum_{d|T} \sigma(d)\mu(\frac{T}{d})$$

和上面相同的套路，枚举 $T=di$

$$ans=\sum_{T=1}^{n}\left\lfloor\frac{n}{T}\right\rfloor\left\lfloor\frac{m}{T}\right\rfloor\sum_{d|T}\sigma(d)\mu(\frac{T}{d})$$

设 $f(x)=\sum_{d|x}\sigma(d)\mu(\frac{x}{d})$
现在的问题是，只有 $\sigma(d)\le a$ 时才能对 $f(x)$ 造成贡献。
将询问按照 a 递增排序，每次 a 增大时，把新的 d 的贡献加进去。这一部分的复杂度是调和级数的 $O(n\log n)$。

和上面相同的套路，枚举 $T=di$

$$ans=\sum_{T=1}^{n}\left\lfloor\frac{n}{T}\right\rfloor\left\lfloor\frac{m}{T}\right\rfloor\sum_{d|T}\sigma(d)\mu(\frac{T}{d})$$

设 $f(x)=\sum_{d|x}\sigma(d)\mu(\frac{x}{d})$
现在的问题是，只有 $\sigma(d)\le a$ 时才能对 $f(x)$ 造成贡献。
将询问按照 a 递增排序，每次 a 增大时，把新的 d 的贡献加进去。这一部分的复杂度是调和级数的 $O(n\log n)$。
现在我们需要单点修改 f，查询区间和，使用树状数组。
一共 $O(n\log n)$ 次修改，$O(q\sqrt{n}))$ 次查询，总复杂度 $O(n\log^2 n+q\sqrt{n}\log n)$

杜教筛

对于数论函数 f，杜教筛可以在低于线性时间的复杂度内计算 $S(n) = \sum_{i=1}^{n} f(i)$。

洛谷 P4213 【模板】杜教筛（Sum）

求 $\sum_{i=1}^{n} \varphi(i)$ 和 $\sum_{i=1}^{n} \mu(i)$
$1 \leq n < 2^{31}$。

杜教筛

对于数论函数 f，杜教筛可以在低于线性时间的复杂度内计算 $S(n)=\sum_{i=1}^{n}f(i)$。

洛谷 P4213 【模板】杜教筛（Sum）

求 $\sum_{i=1}^{n}\varphi(i)$ 和 $\sum_{i=1}^{n}\mu(i)$
$1\le n<2^{31}$。

我们考虑一个更一般的情况，对于数论函数 f，计算 $S(n)=\sum_{i=1}^{n}f(i)$。

杜教筛

对于数论函数 f，杜教筛可以在低于线性时间的复杂度内计算 $S(n)=\sum_{i=1}^{n} f(i)$。

洛谷 P4213 【模板】杜教筛（Sum）

求 $\sum_{i=1}^{n}\varphi(i)$ 和 $\sum_{i=1}^{n}\mu(i)$
$1\le n<2^{31}$。

我们考虑一个更一般的情况，对于数论函数 f，计算 $S(n)=\sum_{i=1}^{n} f(i)$。
我们找一个积性函数 g，考虑其与 f 的狄利克雷卷积：

$$\sum_{i=1}^{n}(f*g)(i)=\sum_{i=1}^{n}\sum_{d\mid i} g(d)f\left(\frac{i}{d}\right)=\sum_{d=1}^{n} g(d)\left(\sum_{i=1}^{\lfloor\frac{n}{d}\rfloor} f(i)\right)$$

$$=\sum_{i=1}^{n} g(i)S\left(\left\lfloor\frac{n}{i}\right\rfloor\right)$$

$S(n)=g(1)S(n)$ 可以写成 $\sum_{i=1}^{n} g(i)S\left(\left\lfloor\frac{n}{i}\right\rfloor\right)-\sum_{i=2}^{n} g(i)S\left(\left\lfloor\frac{n}{i}\right\rfloor\right)$，即

$$\sum_{i=1}^{n}(f*g)(i)-\sum_{i=2}^{n} g(i)S(\left\lfloor\frac{n}{i}\right\rfloor)$$

$S(n)=g(1)S(n)$ 可以写成 $\sum_{i=1}^{n} g(i)S\left(\left\lfloor\frac{n}{i}\right\rfloor\right)-\sum_{i=2}^{n} g(i)S\left(\left\lfloor\frac{n}{i}\right\rfloor\right)$，即

$$\sum_{i=1}^{n}(f*g)(i)-\sum_{i=2}^{n} g(i)S(\left\lfloor\frac{n}{i}\right\rfloor)$$

令 $f*g=h$，则:

$$g(1)S(n)=\sum_{i=1}^{n} h(i)-\sum_{i=2}^{n} g(i)S(\left\lfloor\frac{n}{i}\right\rfloor)$$

如果 h 的前缀和是好求的，g 的前缀和也是好求的，那么使用数论分块就可以递归快速求出 $S(n)$。

$S(n)=g(1)S(n)$ 可以写成 $\sum_{i=1}^{n} g(i)S\left(\left\lfloor\frac{n}{i}\right\rfloor\right)-\sum_{i=2}^{n} g(i)S\left(\left\lfloor\frac{n}{i}\right\rfloor\right)$，即

$$\sum_{i=1}^{n}(f*g)(i)-\sum_{i=2}^{n} g(i)S(\left\lfloor\frac{n}{i}\right\rfloor)$$

令 $f*g=h$，则:

$$g(1)S(n)=\sum_{i=1}^{n} h(i)-\sum_{i=2}^{n} g(i)S(\left\lfloor\frac{n}{i}\right\rfloor)$$

如果 h 的前缀和是好求的，g 的前缀和也是好求的，那么使用数论分块就可以递归快速求出 $S(n)$。
由 $\left\lfloor\frac{\lfloor\frac{n}{a}\rfloor}{b}\right\rfloor=\left\lfloor\frac{n}{ab}\right\rfloor$ 可知，不同的 $\left\lfloor\frac{n}{i}\right\rfloor$ 只有 $O(\sqrt{n})$ 种，于是我们只需记忆化求出所有 $O(\sqrt{n})$ 个 $S(\left\lfloor\frac{n}{i}\right\rfloor)$ 的值即可。

时间复杂度

我们认为计算 h 和 g 的前缀和的时间复杂度均为 $O(1)$。设计算 $S(n)$ 的复杂度为 $T(n)$，由于我们有记忆化，所以有

$$
\begin{aligned}
T(n) &= \Theta(\sqrt{n}) + \sum_{k=1}^{\lfloor\sqrt{n}\rfloor} O(\sqrt{k}) + \sum_{k=2}^{\lfloor\sqrt{n}\rfloor} O\left(\sqrt{\frac{n}{k}}\right) \\
&= O\left(\int_0^{\sqrt{n}} \left(\sqrt{x} + \sqrt{\frac{n}{x}}\right) \mathrm{d}x\right) \\
&= O\left(n^{3/4}\right).
\end{aligned}
$$

若我们可以预处理出一部分 $S(k)$, 其中 $k=1,2,\ldots,m$, $m \geq \lfloor\sqrt{n}\rfloor$. 设预处理的时间复杂度为 $T_0(m)$，则此时的 $T(n)$ 为:

$$\begin{aligned} T(n) &= T_0(m) + \sum_{k \in R(n); k>m} T(k) \\ &= T_0(m) + \sum_{k=1}^{\lfloor n/m \rfloor} O\left(\sqrt{\frac{n}{k}}\right) \\ &= O\left(T_0(m) + \int_0^{n/m} \sqrt{\frac{n}{x}} \mathrm{d}x\right) \\ &= O\left(T_0(m) + \frac{n}{\sqrt{m}}\right). \end{aligned}$$

若 $T_0(m)=O(m)$（如线性筛），由均值不等式可知：当 $m=\Theta\left(n^{2/3}\right)$ 时，$T(n)$ 取得最小值 $O\left(n^{2/3}\right)$。

筛 μ

$$\mu * 1 = \epsilon$$

带到我们的式子里：

$$1(1)S(n) = \sum_{i=1}^{n} \epsilon(i) - \sum_{i=2}^{n} 1(i) S\left(\left\lfloor \frac{n}{i} \right\rfloor\right)$$

即：

$$S(n) = 1 - \sum_{i=2}^{n} 1(i) S\left(\left\lfloor \frac{n}{i} \right\rfloor\right)$$

$1(i)$ 的前缀和就是 i。杜教筛时间复杂度 $O(n^{\frac{2}{3}})$

筛 φ

法一：

$$\sum_{i=1}^{n}\sum_{j=1}^{n}[\gcd(i,j)=1]=\sum_{i=1}^{n}\sum_{j=1}^{n}\sum_{d|i,d|j}\mu(d)$$
$$=\sum_{d=1}^{n}\mu(d)\left\lfloor\frac{n}{d}\right\rfloor^2$$

由于题目所求的是 $\sum_{i=1}^{n}\sum_{j=1}^{i}[\gcd(i,j)=1]$，所以我们排除掉 $i=1,j=1$ 的情况，并将结果除以 2 即可。
观察到，只需求出莫比乌斯函数的前缀和，就可以快速计算出欧拉函数的前缀和了。
时间复杂度 $O(n^{\frac{2}{3}})$。

筛 φ

法二：

$$\varphi * 1 = \mathrm{id}$$

带到我们的式子里：

$$1(1)S(n) = \sum_{i=1}^{n} \mathrm{id}(i) - \sum_{i=2}^{n} 1(i) S\left(\left\lfloor \frac{n}{i} \right\rfloor\right)$$

即：

$$S(n) = \frac{n(n+1)}{2} - \sum_{i=2}^{n} 1(i) S\left(\left\lfloor \frac{n}{i} \right\rfloor\right)$$

$1(i)$ 的前缀和就是 i。杜教筛时间复杂度 $O(n^{\frac{2}{3}})$

例题

洛谷 P3768 简单的数学题

给定 n 和模数 p，求

$$\left(\sum_{i=1}^{n}\sum_{j=1}^{n} i \cdot j \cdot \gcd(i,j)\right) \bmod p$$

$1 \le n \le 10^{10}, 5 \times 10^8 \le p \le 1.1 \times 10^9$ 且 p 为质数，4s。

根据套路，使用 $\gcd(i,j)=\sum_{d|i,d|j}\varphi(d)$

$$
\begin{aligned}
&\sum_{i=1}^{n}\sum_{j=1}^{n} ij\gcd(i,j)\\
=&\sum_{i=1}^{n}\sum_{j=1}^{n} ij\sum_{d|i,d|j}\varphi(d)\\
=&\sum_{d=1}^{n}\varphi(d)\sum_{d|i}\sum_{d|j} ij\\
=&\sum_{d=1}^{n}\varphi(d)d^2\sum_{i=1}^{\lfloor\frac{n}{d}\rfloor}\sum_{j=1}^{\lfloor\frac{n}{d}\rfloor} ij\\
=&\sum_{d=1}^{n}\varphi(d)d^2\left(\sum_{i=1}^{\lfloor\frac{n}{d}\rfloor} i\right)^2
\end{aligned}
$$

记 $F(n) = (\sum_{i=1}^{n} i)^2 = \left(\frac{n(n+1)}{2}\right)^2$

$$原式 = \sum_{i=1}^{n} \varphi(i) i^2 F\left(\left\lfloor \frac{n}{i} \right\rfloor\right)$$

后面的那部分整除分块即可，我们需要快速计算 $\varphi(i)i^2$ 的前缀和，考虑杜教筛：

$$\sum_{i=1}^{n} \varphi(i) i^2$$

令 $f = \varphi \cdot \mathrm{id}^2$，我们需要找到合适的 g。

记 $F(n)=(\sum_{i=1}^{n} i)^2=\left(\frac{n(n+1)}{2}\right)^2$

$$\text{原式}=\sum_{i=1}^{n}\varphi(i)i^2F\left(\left\lfloor\frac{n}{i}\right\rfloor\right)$$

后面的那部分整除分块即可，我们需要快速计算 $\varphi(i)i^2$ 的前缀和，考虑杜教筛：

$$\sum_{i=1}^{n}\varphi(i)i^2$$

令 $f=\varphi\cdot\mathrm{id}^2$，我们需要找到合适的 g。
为了 $(f*g)$ 求前缀和方便，由于 $i\cdot\frac{n}{i}=n$，取 $g=\mathrm{id}^2$，以消掉 i。

$$(f*g)(n)=\sum_{d|n}\varphi(d)d^2\cdot\left(\frac{n}{d}\right)^2=\sum_{d|n}\varphi(d)n^2=n^2\sum_{d|n}\varphi(d)=n^3$$

$$f=\varphi\cdot\mathrm{id}^2, g=\mathrm{id}^2$$

带到我们的式子里:

$$g(1)S(n)=\sum_{i=1}^{n}(f*g)(i)-\sum_{i=2}^{n}\mathrm{id}^2(i)S\left(\left\lfloor\frac{n}{i}\right\rfloor\right)$$

$$f=\varphi\cdot\mathrm{id}^2, g=\mathrm{id}^2$$

带到我们的式子里：

$$g(1)S(n)=\sum_{i=1}^{n}(f*g)(i)-\sum_{i=2}^{n}\mathrm{id}^2(i)S\left(\left\lfloor\frac{n}{i}\right\rfloor\right)$$

$$\sum_{i=1}^{n}i^2=\frac{n(n+1)(2n+1)}{6}$$

$$\sum_{i=1}^{n}i^3=\frac{n^2(n+1)^2}{4}$$

于是 $(f*g)$ 和 g 的前缀和都可以 $O(1)$ 求。复杂度 $O(n^{\frac{2}{3}})$

点乘

定义**点乘**运算，对于两函数 A, B ，其点乘 $A \cdot B$ 为一个新的函数，满足 $(A \cdot B)(n) = A(n)B(n)$ 。
当 C 是完全积性函数时，有 $(A \cdot C) * (B \cdot C) = (A * B) \cdot C$。

点乘

定义**点乘**运算，对于两函数 A, B ，其点乘 $A \cdot B$ 为一个新的函数，满足 $(A \cdot B)(n) = A(n)B(n)$ 。
当 C 是完全积性函数时，有 $(A \cdot C) * (B \cdot C) = (A * B) \cdot C$。
下面是一些常见的基本和组：

- $\mu \cdot id_k$

$$(\mu \cdot id_k) * id_k = (\mu \cdot id_k) * (1 \cdot id_k) = (\mu * 1) \cdot id_k = \epsilon$$

- $\varphi \cdot id_k$

$$(\varphi \cdot id_k) * id_k = (\varphi \cdot id_k) * (1 \cdot id_k) = (\varphi * 1) \cdot id_k = id_{k+1}$$

例题

51Nod2026 Gcd and Lcm

已知

$$f(x)=\sum_{d|x}\mu(d)\cdot d$$

求下面式子的值：

$$\sum_{i=1}^{n}\sum_{j=1}^{n}f(\gcd(i,j))\times f(\mathrm{lcm}(i,j))$$

$n\leq 10^9$

对于所有积性函数，都有 $f(\gcd(i,j))f(\operatorname{lcm}(i,j)) = f(i)f(j)$
这样问题变为：

$$\sum_{i=1}^{n}\sum_{j=1}^{n} f(i)f(j)$$
$$= \left(\sum_{i=1}^{n} f(i)\right)^2$$

现在只需要求 $f(i)$ 的前缀和

对于所有积性函数，都有 $f(\gcd(i,j))f(\mathrm{lcm}(i,j)) = f(i)f(j)$
这样问题变为：

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

$$= \left(\sum_{i=1}^{n} f(i)\right)^2$$

现在只需要求 $f(i)$ 的前缀和

$$\sum_{i=1}^{n} f(i) = \sum_{i=1}^{n}\sum_{d|i} \mu(i)i$$

$$= \sum_{i=1}^{n} \mu(i)i\left\lfloor \frac{n}{i} \right\rfloor$$

根据基本和组杜教筛求出 $\mu(i)i$ 的前缀和，再套一层整除分块即可。

例题

「NOI2016」循环之美

牛牛认为，如果在 k 进制下，一个数的小数部分是纯循环的，那么它就是美的。现在，牛牛想知道：对于已知的十进制数 n 和 m，在 k 进制下，有多少个数值上互不相等的纯循环小数，可以用分数 $\frac{x}{y}$ 表示，其中 $1 \le x \le n, 1 \le y \le m$，且 x, y 是整数。一个数是纯循环的，当且仅当其可以写成以下形式：

$$a.\dot{c}_1 c_2 c_3 \dots c_{p-1} \dot{c}_p$$

其中，a 是一个整数，$p \ge 1$；对于 $1 \le i \le p$，c_i 是 k 进制下的一位数字。
例如，在十进制下，$0.45454545 \dots = 0.\dot{4}\dot{5}$ 是纯循环的，它可以用 $\frac{5}{11}$、$\frac{10}{22}$ 等分数表示；在十进制下，$0.1666666 \dots = 0.1\dot{6}$ 则不是纯循环的，它可以用 $\frac{1}{6}$ 等分数表示。
需要特别注意的是，我们认为一个整数是纯循环的，因为它的小数部分可以表示成 0 的循环或是 $k-1$ 的循环；而一个小数部分非 0 的有限小数不是纯循环的。
$1 \le n \le 10^9$，$1 \le m \le 10^9$，$2 \le k \le 2 \times 10^3$。

首先要求数值互不相同，就相当于要求分子分母互质（不互质就会重复）
考虑怎么转化纯循环小数这个条件：
设 $\frac{x}{y}$ 循环节为 l 位，这意味着 $\frac{x}{y}$ 向左平移 l 位后，小数点后的部分不变，即：

$$xk^l \equiv x \pmod y$$

x, y 互质，因此只要 k, y 互质即可。
于是只需要求

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

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

这里可以整除分块了，考虑分别计算：

$$
f(x)=\sum_{i=1}^{x}[\gcd(i,k)=1],\ \ g(x,k)=\sum_{i=1}^{x}[\gcd(i,k)=1]\mu(i)
$$

注意到 k 非常小，由 $\gcd(i,k)=\gcd(i \bmod k,k)$，我们只需要对 $1\sim k$ 中每个数 x $O(k^2)$ 预处理 $F_k(x)$ 表示 $\le x$ 的数中与 k 互质的数的个数，即可 $\mathcal{O}(1)$ 求解单个 f。

注意到 k 非常小，由 $\gcd(i,k)=\gcd(i \bmod k, k)$，我们只需要对 $1\sim k$ 中每个数 x $O(k^2)$ 预处理 $F_k(x)$ 表示 $\le x$ 的数中与 k 互质的数的个数，即可 $\mathcal{O}(1)$ 求解单个 f。但是 g 不行，因为 $\mu(i)\neq\mu(i+k)$，两者的关系很复杂。

$$
\begin{aligned}
g(x,k) &= \sum_{i=1}^{x}[\gcd(i,k)=1]\mu(i) \\
&= \sum_{i=1}^{x}\mu(i)\sum_{d\mid\gcd(i,k)}\mu(d) \\
&= \sum_{d\mid k}\mu(d)\sum_{i=1}^{x/d}\mu(id)
\end{aligned}
$$

若 $\gcd(i,d)\neq 1$ 则 $\mu(id)=0$，对答案没有贡献，因此当 $gcd(i,d)=1$，可以积性函数展开后面的 μ。

$$\begin{aligned} g(x,k) &= \sum_{d|k}^{x} \mu^2(d) \sum_{i=1}^{x/d} \mu(i)[\gcd(i,d)=1] \\ &= \sum_{d|k} \mu^2(d) g\left(\frac{x}{d}, d\right) \end{aligned}$$

于是可以递归求解所有的 g，初始值 $g(x,1)=\sum_{i=1}^{x}\mu(i)$ 杜教筛即可。
复杂度 $\mathcal{O}(\sqrt{n}\sigma_0(k)+n^{\frac{2}{3}})$

Powerful Number

定义：对于正整数 n，记 n 的质因数分解为 $n=\prod_{i=1}^{m} p_i^{e_i}$。$n$ 是 Powerful Number (简称 PN）当且仅当 $\forall 1 \le i \le m, e_i > 1$。

Powerful Number

定义：对于正整数 n，记 n 的质因数分解为 $n=\prod_{i=1}^{m} p_i^{e_i}$。$n$ 是 Powerful Number (简称 PN) 当且仅当 $\forall 1 \le i \le m, e_i > 1$。PN 有如下性质：

1. 所有 PN 都可以表示成 a^2b^3 的形式。
   证明：若 e_i 是偶数，则将 $p_i^{e_i}$ 合并进 a^2 里；若 e_i 为奇数，则先将 p_i^3 合并进 b^3 里，再将 $p_i^{e_i-3}$ 合并进 a^2 里。
2. n 以内的 PN 有 $O(\sqrt{n})$ 个。
   证明：考虑枚举 a，再考虑满足条件的 b 的个数，有 PN 的个数约等于

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

Powerful Number

定义：对于正整数 n，记 n 的质因数分解为 $n=\prod_{i=1}^{m} p_i^{e_i}$。$n$ 是 Powerful Number (简称 PN）当且仅当 $\forall 1 \le i \le m, e_i > 1$。PN 有如下性质：

1. 所有 PN 都可以表示成 a^2b^3 的形式。
   证明：若 e_i 是偶数，则将 $p_i^{e_i}$ 合并进 a^2 里；若 e_i 为奇数，则先将 p_i^3 合并进 b^3 里，再将 $p_i^{e_i-3}$ 合并进 a^2 里。
2. n 以内的 PN 有 $O(\sqrt{n})$ 个。
   证明：考虑枚举 a，再考虑满足条件的 b 的个数，有 PN 的个数约等于

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

那么如何求出 n 以内所有的 PN 呢？线性筛找出 $\sqrt{n}$ 内的所有素数，再 DFS 搜索各素数的指数即可。由于 n 以内的 PN 至多有 $O(\sqrt{n})$ 个，所以至多搜索 $O(\sqrt{n})$ 次。

例题

CodeChef TABRARRAY

给定一棵 n 个点的以 1 为根的有根树，以及一个常数 K。你需要往每个节点上填上一个正整数，同时，要求 $a_{fa_i} \bmod a_i = 0$，且 $\prod_{i=1}^{n} a_i \le K$。
求填数的方案数 $\bmod 998244353$。
$n \le 10^4, K \le 10^{14}$

先假设 $a_1 = \operatorname{lcm}_{i=2}^{n} a_i$，若当前的 $prod = \prod_{i=1}^{n} a_i = x$，那么对答案的贡献是 $\lfloor \frac{K}{x} \rfloor$。

此时注意到一个质因数在 $prod$ 里的出现次数要么为 0，要么 ≥ 2，也就是说，此时的 $prod$ 是一个 Powerful Number。这样的 $prod$ 只有 $O(\sqrt{K})$ 个。

先假设 $a_1 = \operatorname{lcm}_{i=2}^{n} a_i$，若当前的 $prod = \prod_{i=1}^{n} a_i = x$，那么对答案的贡献是 $\lfloor \frac{K}{x} \rfloor$。

此时注意到一个质因数在 $prod$ 里的出现次数要么为 0，要么 ≥ 2，也就是说，此时的 $prod$ 是一个 Powerful Number。这样的 $prod$ 只有 $O(\sqrt{K})$ 个。

而显然每个质因子是独立的，树形 DP。设 $f_{u,i,j}$ 表示以 u 为根的子树内，u 上填的数 $= i$，子树内所有填的数之和是 j，要求 fa_u 填的数 $\geq u$ 填的数的方案数。前缀和优化转移做到 $O(n \log^3 K)$。注意因为我们钦定了 $a_1 = \operatorname{lcm}_{i=2}^{n} a_i$，所以一号点的转移略微有点特殊。

然后我们在 dfs 找 PN 的时候一起把方案数算出来就行了。总复杂度 $O(n \log^3 K + \sqrt{K})$。

Powerful Number 筛

Powerful Number 筛要求存在一个函数 g 满足：

- g 是积性函数。
- g 易求前缀和。
- 对于质数 p，$g(p)=f(p)$。

假设现在要求积性函数 f 的前缀和

$$F(n)=\sum_{i=1}^{n} f(i)$$

首先，构造出一个易求前缀和的积性函数 g，且满足对于素数 p，$g(p)=f(p)$。记 $G(n)=\sum_{i=1}^{n} g(i)$。
然后，构造函数 $h=f/g$，这里的 $/$ 表示狄利克雷卷积除法。根据狄利克雷卷积的性质可以得知 h 也为积性函数，因此 $h(1)=1$。$f=g*h$，这里 $*$ 表示狄利克雷卷积。
对于素数 p，$f(p)=g(1)h(p)+g(p)h(1)=h(p)+g(p) \implies h(p)=0$。根据 $h(p)=0$ 和 h 是积性函数可以推出对于非 PN 的数 n 有 $h(n)=0$，即 h 仅在 PN 处取有效值。

现在，根据 $f = g * h$ 有

$$
\begin{aligned}
F(n) = \sum_{i=1}^{n} f(i) &= \sum_{i=1}^{n} \sum_{d|i} h(d) g\left(\frac{i}{d}\right) \\
&= \sum_{d=1}^{n} \sum_{i=1}^{\lfloor \frac{n}{d} \rfloor} h(d) g(i) \\
&= \sum_{d=1}^{n} h(d) \sum_{i=1}^{\lfloor \frac{n}{d} \rfloor} g(i) \\
&= \sum_{d=1}^{n} h(d) G\left(\left\lfloor \frac{n}{d} \right\rfloor\right) \\
&= \sum_{\substack{d=1 \\ d \text{ is PN}}}^{n} h(d) G\left(\left\lfloor \frac{n}{d} \right\rfloor\right)
\end{aligned}
$$

$O(\sqrt{n})$ 找出所有 PN，计算出所有 h 的有效值。对于 h 有效值的计算，只需要计算出所有 $h(p^c)$ 处的值，就可以根据 h 为积性函数推出 h 的所有有效值。

现在对于每一个有效值 d，计算 $h(d)G\left(\left\lfloor\frac{n}{d}\right\rfloor\right)$ 并累加即可得到 $F(n)$。

下面考虑计算 $h(p^c)$，一共有两种方法：

- 直接推出 $h(p^c)$ 仅与 p, c 有关的计算公式，再根据公式计算 $h(p^c)$
- 根据 $f = g * h$ 有 $f(p^c) = \sum_{i=0}^{c} g(p^i)h(p^{c-i})$，移项可得 $h(p^c) = f(p^c) - \sum_{i=1}^{c} g(p^i)h(p^{c-i})$，现在就可以枚举素数 p 再枚举指数 c 求解出所有 $h(p^c)$。

$O(\sqrt{n})$ 找出所有 PN，计算出所有 h 的有效值。对于 h 有效值的计算，只需要计算出所有 $h(p^c)$ 处的值，就可以根据 h 为积性函数推出 h 的所有有效值。

现在对于每一个有效值 d，计算 $h(d)G\left(\left\lfloor\frac{n}{d}\right\rfloor\right)$ 并累加即可得到 $F(n)$。

下面考虑计算 $h(p^c)$，一共有两种方法：

- 直接推出 $h(p^c)$ 仅与 p, c 有关的计算公式，再根据公式计算 $h(p^c)$
- 根据 $f = g * h$ 有 $f(p^c) = \sum_{i=0}^{c} g(p^i)h(p^{c-i})$，移项可得 $h(p^c) = f(p^c) - \sum_{i=1}^{c} g(p^i)h(p^{c-i})$，现在就可以枚举素数 p 再枚举指数 c 求解出所有 $h(p^c)$。

复杂度 $O(\sqrt{n}\log n)$，而且这个上界比较宽松

例题

洛谷 P5325 【模板】Min_25 筛

给定积性函数 f，其在质数的幂次的取值为 $f(p^k)=p^k(p^k-1)$，求

$$\sum_{i=1}^{n} f(i)$$

$n \le 10^{10}$

例题

洛谷 P5325 【模板】Min_25 筛

给定积性函数 f，其在质数的幂次的取值为 $f(p^k)=p^k(p^k-1)$，求

$$\sum_{i=1}^{n} f(i)$$

$n\le 10^{10}$

易得 $f(p)=p(p-1)=\mathrm{id}(p)\cdot\varphi(p)$，构造 $g(n)=\mathrm{id}(n)\cdot\varphi(n)$。
使用杜教筛求 $G(n)$，这是基本和组。
之后 $h(p^k)$ 的取值可以枚举计算，这种方法不再赘述。

例题

洛谷 P5325 【模板】Min_25 筛

给定积性函数 f，其在质数的幂次的取值为 $f(p^k) = p^k(p^k - 1)$，求

$$\sum_{i=1}^{n} f(i)$$

$n \leq 10^{10}$

易得 $f(p) = p(p-1) = \mathrm{id}(p) \cdot \varphi(p)$，构造 $g(n) = \mathrm{id}(n) \cdot \varphi(n)$。
使用杜教筛求 $G(n)$，这是基本和组。
之后 $h(p^k)$ 的取值可以枚举计算，这种方法不再赘述。
此外，此题还可以直接求出 $h(p^k)$ 仅与 p, k 有关的公式

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

$$\iff p^k(p^k-1)=\sum_{i=0}^{k} p^{2k-2i-1}(p-1)h(p^i)$$

$$\iff p^k(p^k-1)=h(p^k)+\sum_{i=0}^{k-1} p^{2k-2i-1}(p-1)h(p^i)$$

$$\iff h(p^k)=p^k(p^k-1)-\sum_{i=0}^{k-1} p^{2k-2i-1}(p-1)h(p^i)$$

$$\iff h(p^k)-p^2h(p^{k-1})=p^k(p^k-1)-p^{k+1}(p^{k-1}-1)-p(p-1)h(p^{k-1})$$

$$\iff h(p^k)-ph(p^{k-1})=p^{k+1}-p^k$$

$$\iff \frac{h(p^k)}{p^k}-\frac{h(p^{k-1})}{p^{k-1}}=p-1$$

再根据 $h(p)=0$，通过累加法即可推出 $h(p^k)=(k-1)(p-1)p^k$。

例题

SPOJ UDIVSUM - The Sum of Unitary Divisors

对于一个自然数 n，如果 $d \mid n$，且 $\gcd\left(d, \frac{n}{d}\right) = 1$，则称 d 为 n 的“元因子”。

我们定义函数 $\sigma^*(n)$ 表示 n 的元因子之和。例如 $\sigma^*(1) = 1$，$\sigma^*(2) = 3$，$\sigma^*(6) = 12$。

$$S(n) = \sum_{i=1}^{n} \sigma^*(i)$$

现给定 n，求 $S(n) \bmod 2^{64}$。

σ^* 的积性容易验证，则仅考虑其在质数幂处的值

$$\sigma^*(p^k)=\sum_{i=0}^{k}p^i[\gcd(p^i,p^{k-i})=1]=\sum_{i=0}^{k}p^i[\min(i,k-i)=0]=p^k+[k>0]$$

由于该积性函数具有很好的性质：$\sigma^*(p)=p+1=\sigma(p)$ ，故考虑 Powerful Number 筛：
令积性函数 h 满足 $\sigma^*=\sigma*h$

$$\sigma^*(p^k)=\sum_{i=0}^{k}\sigma(p^i)h(p^{k-i})$$

易得

$$h(p^k)=\begin{cases}-p, k=2\\0, k\neq 2\end{cases}\quad ,k>0$$

由 Powerful Number 筛，有：

$$
\begin{aligned}
S(n) &= \sum_{i=1}^{n} \sigma^*(i) \\
&= \sum_{i=1}^{n} \sum_{d|i} h(d)\sigma(\frac{i}{d}) \\
&= \sum_{d=1}^{n} h(d) \sum_{i=1}^{n} [d \mid i]\sigma(\frac{i}{d}) \\
&= \sum_{d \in PN} h(d) \sum_{i=1}^{n/d} \sigma(i) \\
&= \sum_{d \in PN} h(d) S_1(n/d)
\end{aligned}
$$

而 $S_1(n) = \sum_{i=1}^{n} \sigma(i) = \sum_{i=1}^{n} \sum_{d|i} d = \sum_{d=1}^{n} d \cdot (n/d)$
对右边整除分块即可 $O(\sqrt{n})$ 时间内求解 $S_1(n)$，总复杂度 $O(\sqrt{n} \log n)$

Stern-Brocot 树

Stern-Brocot 树是一种维护最简分数的数据结构。

Stern-Borcot 树可以在迭代构造第 k 阶 Stern-Brocot 序列的过程中得到。第 0 阶 Stern-Brocot 序列由两个简单的分数组成：

$$\frac{0}{1}, \frac{1}{0}.$$

此处的 $\frac{1}{0}$ 严格意义上并不算是有理分数，可以理解为表示 ∞ 的最简分数。

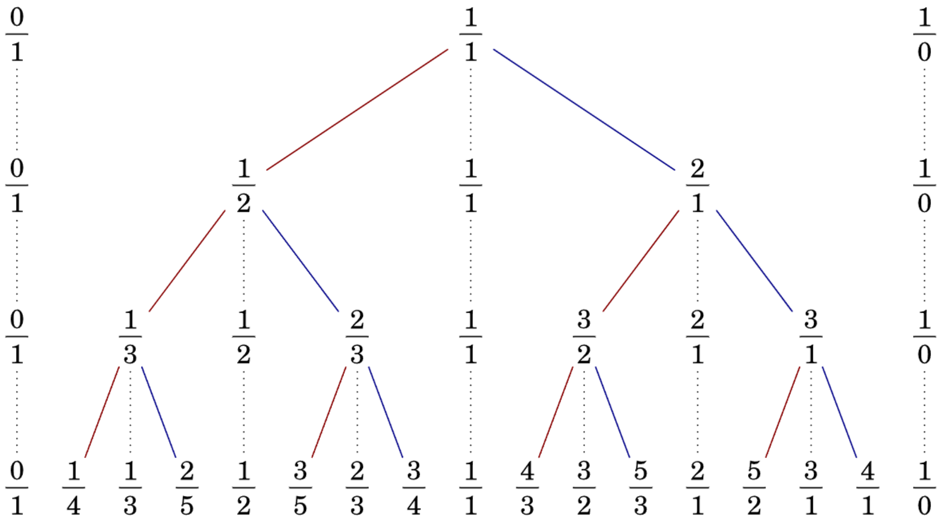
在第 k 阶 Stern-Brocot 序列相邻的两个分数 $\frac{a}{b}$ 和 $\frac{c}{d}$ 中间插入 $\frac{a+c}{b+d}$，就得到第 $k+1$ 阶 Stern-Brocot 序列。前几次迭代的结果如下：

$$\frac{0}{1}, \frac{1}{1}, \frac{1}{0}$$

$$\frac{0}{1}, \frac{1}{2}, \frac{1}{1}, \frac{2}{1}, \frac{1}{0}$$

$$\frac{0}{1}, \frac{1}{3}, \frac{1}{2}, \frac{2}{3}, \frac{1}{1}, \frac{3}{2}, \frac{2}{1}, \frac{3}{1}, \frac{1}{0}$$

用树来表示，就是 Stern-Brocot 树



每个点上有一个" 三元组"(a, b, c)，$\left(\frac{0}{1}, \frac{1}{1}, \frac{1}{0}\right)$ 作为根节点，且在每个节点 $\left(\frac{a}{b}, \frac{p}{q}, \frac{c}{d}\right)$ 后 $\left(\frac{a}{b}, \frac{a+p}{b+q}, \frac{p}{q}\right)$ 为其左儿子，$\left(\frac{p}{q}, \frac{p+c}{q+d}, \frac{c}{d}\right)$ 为其右儿子

每个点上有一个" 三元组"(a, b, c)，$\left(\frac{0}{1}, \frac{1}{1}, \frac{1}{0}\right)$ 作为根节点，且在每个节点 $\left(\frac{a}{b}, \frac{p}{q}, \frac{c}{d}\right)$ 后 $\left(\frac{a}{b}, \frac{a+p}{b+q}, \frac{p}{q}\right)$ 为其左儿子，$\left(\frac{p}{q}, \frac{p+c}{q+d}, \frac{c}{d}\right)$ 为其右儿子

根据构造，Stern-Brocot 树显然是分数的**二叉搜索树**，即树上节点的中序遍历得到的分数序列是递增的。除此之外，他还满足如下两个重要性质：

- 最简性：Stern-Brocot 树构造出来的所有分数都是最简分数
- 完全性：Stern-Brocot 树能构造出来任意一个正的最简分数

每个点上有一个" 三元组"(a, b, c)，$\left(\frac{0}{1}, \frac{1}{1}, \frac{1}{0}\right)$ 作为根节点，且在每个节点 $\left(\frac{a}{b}, \frac{p}{q}, \frac{c}{d}\right)$ 后 $\left(\frac{a}{b}, \frac{a+p}{b+q}, \frac{p}{q}\right)$ 为其左儿子，$\left(\frac{p}{q}, \frac{p+c}{q+d}, \frac{c}{d}\right)$ 为其右儿子

根据构造，Stern-Brocot 树显然是分数的**二叉搜索树**，即树上节点的中序遍历得到的分数序列是递增的。除此之外，他还满足如下两个重要性质：

- 最简性：Stern-Brocot 树构造出来的所有分数都是最简分数
- 完全性：Stern-Brocot 树能构造出来任意一个正的最简分数

完全性是显然的，因为往子树搜索一定会让分子或者分母变大，这样对于任意一个最简分数 $\frac{p}{q}$，搜索次数是有限的。

下面证明最简性。

最简性的证明

三元组 $\left(\frac{a}{b}, \frac{a+c}{b+d}, \frac{c}{d}\right)$ 的构造意味着每个 Stern-Brocot 树上的节点都对应着一个矩阵

$$S = \begin{pmatrix} b & d \\ a & c \end{pmatrix}.$$

由数学归纳法，$\frac{a}{b}$ 和 $\frac{c}{d}$ 其中一个是上一层的分数，不妨设 $\frac{a}{b}$ 是最简分数，即 $\gcd(a, b) = 1$，我们证明 $\gcd(c, d) = 1$。

最简性的证明

三元组 $\left(\frac{a}{b}, \frac{a+c}{b+d}, \frac{c}{d}\right)$ 的构造意味着每个 Stern-Brocot 树上的节点都对应着一个矩阵

$$S = \begin{pmatrix} b & d \\ a & c \end{pmatrix}.$$

由数学归纳法，$\frac{a}{b}$ 和 $\frac{c}{d}$ 其中一个是上一层的分数，不妨设 $\frac{a}{b}$ 是最简分数，即 $\gcd(a, b) = 1$，我们证明 $\gcd(c, d) = 1$。
Stern–Brocot 树的根节点是单位矩阵 I，而向左子节点移动和向右子节点移动则分别对应将当前节点矩阵右乘以矩阵

$$L = \begin{pmatrix} 1 & 1 \\ 0 & 1 \end{pmatrix}, \ R = \begin{pmatrix} 1 & 0 \\ 1 & 1 \end{pmatrix}.$$

而我们有 $\det I = \det L = \det R = 1$，于是 S 是 I 和一堆 L 和 R 之积，则 $\det S = 1$

于是

$$bc - ad = \begin{vmatrix} b & d \\ a & c \end{vmatrix} = 1$$

于是有

$$\gcd(c, d) \mid 1 \implies \gcd(c, d) = 1$$

Stern-Brocot 树上查找

- 在 Stern-Brocot 树上查找一个确定的分数 $\frac{p}{q}$

每次暴力往左/右跳的复杂度是 $O(p+q)$ 的，无法接受。
这个复杂度高的原因是我们经常会连续往左/往右跳很多步，而一次往左跳再接一次往右跳（或者一次往右跳接一次往左跳）一定会让分子分母都至少变为原来的两倍。所以“拐弯”的次数是 $O(\log(p+q))$ 的。
只要我们能快速确定每次拐弯的位置就行了。

Stern-Brocot 树上查找

- 在 Stern-Brocot 树上查找一个确定的分数 $\frac{p}{q}$

每次暴力往左/右跳的复杂度是 $O(p+q)$ 的，无法接受。
这个复杂度高的原因是我们经常会连续往左/往右跳很多步，而一次往左跳再接一次往右跳（或者一次往右跳接一次往左跳）一定会让分子分母都至少变为原来的两倍。所以“拐弯”的次数是 $O(\log(p+q))$ 的。
只要我们能快速确定每次拐弯的位置就行了。
如果要查找的分数 $\dfrac{p}{q}$ 落入 $\dfrac{a}{b}$ 和 $\dfrac{c}{d}$ 之间，那么连续 t 次向右移动时，右侧边界保持不动，而左侧节点移动到 $\dfrac{a+tc}{b+td}$ 处；反过来，连续 t 次向左移动时，左侧边界保持不动，而右侧节点移动到 $\dfrac{ta+c}{tb+d}$ 处。因此，可以直接通过 $\dfrac{a+tc}{b+td}<\dfrac{p}{q}$ 或 $\dfrac{p}{q}<\dfrac{ta+c}{tb+d}$ 解出向右和向左移动的次数。
复杂度 $O(\log(p+q))$。

- 大部分时候，我们在树上查找的分数其实是不确定的，比如我们需要根据当前分数的值跑一个二分查找的检验算法来决定具体是往左走还是往右走。

这个时候，把刚才通过解方程来确定次数的部分改为二分连续向左/向右的次数 t 即可。复杂度 $O(\log^2(p+q))$。

例题

洛谷 P5179 Fraction

给你四个正整数 a, b, c, d，求一个最简分数 $\frac{p}{q}$ 满足 $\frac{a}{b} < \frac{p}{q} < \frac{c}{d}$。
若有多组解，输出 q 最小的一组，若仍有多组解，输出 p 最小的一组。
$T \leqslant 500, a, b, c, d \leqslant 10^9$

例题

洛谷 P5179 Fraction

给你四个正整数 a, b, c, d ，求一个最简分数 $\frac{p}{q}$ 满足 $\frac{a}{b} < \frac{p}{q} < \frac{c}{d}$。
若有多组解，输出 q 最小的一组，若仍有多组解，输出 p 最小的一组。
$T \leqslant 500, a, b, c, d \leqslant 10^9$

在 Stern-Brocot 树上二分，连续移动次数可以二分算出来，也可以列出不等式解出来，复杂度 $O(T\log^2 a)$ 或者 $O(T\log a)$。

例题

「BalkanOI 2003」Farey 序列

求分子和分母都 $\leq n$ 的**最简真分数**组成的序列中第 k 小的数。
$2 \leq n \leq 4 \times 10^4$（可加强到 2×10^8），$1 \leq k \leq$ 符合条件的分数的个数。

例题

「BalkanOI 2003」Farey 序列

求分子和分母都 $\leq n$ 的**最简真分数**组成的序列中第 k 小的数。
$2 \leq n \leq 4 \times 10^4$（可加强到 2×10^8），$1 \leq k \leq$ 符合条件的分数的个数。

在 Stern-Brocot 树上二分，剩下的就是类似二分答案地，对于一个分数 $\frac{x}{y}$，我们需要计算 $\leq \frac{x}{y}$ 的分子和分母都 $\leq n$ 的最简真分数数量。

这个数量为

$$\sum_{i=1}^{n}\sum_{j=1}^{i}[\gcd(i,j)=1]\left[\frac{j}{i}\le\frac{x}{y}\right]$$

$$=\sum_{i=1}^{n}\sum_{j=1}^{u}\sum_{d|gcd(i,j)}\mu(d)\left[\frac{j}{i}\le\frac{x}{y}\right]$$

$$=\sum_{d=1}^{n}\mu(d)\sum_{i=1}^{n/d}\sum_{j=1}^{i}\left[\frac{j}{i}\le\frac{x}{y}\right]$$

$\left[\frac{j}{i}\le\frac{x}{y}\right]$ 等价于 $\left[j\le\frac{ix}{y}\right]$，于是

$$=\sum_{d=1}^{n}\mu(d)\sum_{i=1}^{\frac{n}{d}}\left\lfloor\frac{ix}{y}\right\rfloor$$

$$\sum_{d=1}^{n}\mu(d)\sum_{i=1}^{\frac{n}{d}}\left\lfloor\frac{ix}{y}\right\rfloor$$

整除分块，μ 的前缀和用杜教筛求，预处理 $O(n^{\frac{2}{3}})$，单次询问 $O(1)$。
后面的部分是一个类欧，整除分块，对于每个 $\frac{x}{y}$，需要做 $O(\sqrt{n})$ 次类欧，每次类欧 $O(\log n)$。
由于在 Stern-Brocot 树上二分，每一个连续段需要 $O(\log n)$ 二分确定连续段长度，所以总复杂度为 $O(n^{\frac{2}{3}}+\sqrt{n}\log^3 n)$。

完 结 撒 花