简单数论 & 数论函数

任轩笛

EECS,PKU

2019 年 6 月 26 日

Contents

Contents

Contents

两种常见的素性测试

两种常见的素性测试

试除法，配合质数筛法可以做到 $O(\sqrt{n})$ 的复杂度。

两种常见的素性测试

试除法，配合质数筛法可以做到 $O(\sqrt{n})$ 的复杂度。

Miller-Rabin 素性测试，可以做到 $O(\log n)$ 的复杂度，属于 RP 算法。

两种常见的素性测试

试除法，配合质数筛法可以做到 $O(\sqrt{n})$ 的复杂度。

Miller-Rabin 素性测试，可以做到 $O(\log n)$ 的复杂度，属于 RP 算法。

其它还有印度人的 AKS 算法可以做到正确率 100% 的 $O(\log n)$ 复杂度。

两种常见的素性测试

试除法，配合质数筛法可以做到 $O(\sqrt{n})$ 的复杂度。

Miller-Rabin 素性测试，可以做到 $O(\log n)$ 的复杂度，属于 RP 算法。

其它还有印度人的 AKS 算法可以做到正确率 100% 的 $O(\log n)$ 复杂度。

绝大多数情况 Miller-Rabin 都够优秀了。

两种常见的素性测试

试除法，配合质数筛法可以做到 $O(\sqrt{n})$ 的复杂度。

Miller-Rabin 素性测试，可以做到 $O(\log n)$ 的复杂度，属于 RP 算法。

其它还有印度人的 AKS 算法可以做到正确率 100% 的 $O(\log n)$ 复杂度。

绝大多数情况 Miller-Rabin 都够优秀了。

可以查阅 wiki 以确定 Miller-Rabin 应该试到多少。

Miller-Rabin

Miller-Rabin

基本原理是费马小定理：若 p 是质数，a, p 互质，则 $a^{p-1} \equiv 1 \pmod{p}$。

Miller-Rabin

基本原理是费马小定理：若 p 是质数，a, p 互质，则 $a^{p-1} \equiv 1 \pmod{p}$。

于是对于某个 p，若能找到与它互质的 a 使得 $a^{p-1} \neq 1 \pmod{p}$，则 p 必不是质数。

Miller-Rabin

基本原理是费马小定理：若 p 是质数，a, p 互质，则 $a^{p-1} \equiv 1 \pmod{p}$。

于是对于某个 p，若能找到与它互质的 a 使得 $a^{p-1} \neq 1 \pmod{p}$，则 p 必不是质数。

然而有一些合数 p，满足所有与它互质的 a 都有 $a^{p-1} \equiv 1 \pmod{p}$，这种数称为 Carmichael 数（如 $561 = 3 * 11 * 17$），这样的数是用上面的方法检验不出来的。

Miller-Rabin

基本原理是费马小定理：若 p 是质数，a, p 互质，则 $a^{p-1} \equiv 1 \pmod{p}$。

于是对于某个 p，若能找到与它互质的 a 使得 $a^{p-1} \neq 1 \pmod{p}$，则 p 必不是质数。

然而有一些合数 p，满足所有与它互质的 a 都有 $a^{p-1} \equiv 1 \pmod{p}$，这种数称为 Carmichael 数（如 $561 = 3 * 11 * 17$），这样的数是用上面的方法检验不出来的。

所以还需要奇素数判定。对于奇素数 p，如果 $a^{p-1} \equiv 1 \pmod{p}$ 即 $(a^{\frac{p-1}{2}} + 1)(a^{\frac{p-1}{2}} - 1) \equiv 0 \pmod{p}$，由于 F_p 是整环，所以 $a^{\frac{p-1}{2}} \equiv 1$ 或 $p - 1$。

Miller-Rabin

基本原理是费马小定理：若 p 是质数，a, p 互质，则 $a^{p-1} \equiv 1 \pmod{p}$。

于是对于某个 p，若能找到与它互质的 a 使得 $a^{p-1} \neq 1 \pmod{p}$，则 p 必不是质数。

然而有一些合数 p，满足所有与它互质的 a 都有 $a^{p-1} \equiv 1 \pmod{p}$，这种数称为 Carmichael 数（如 $561 = 3 * 11 * 17$），这样的数是用上面的方法检验不出来的。

所以还需要奇素数判定。对于奇素数 p，如果 $a^{p-1} \equiv 1 \pmod{p}$ 即 $(a^{\frac{p-1}{2}} + 1)(a^{\frac{p-1}{2}} - 1) \equiv 0 \pmod{p}$，由于 F_p 是整环，所以 $a^{\frac{p-1}{2}} \equiv 1$ 或 $p-1$。如果 $\frac{p-1}{2}$ 还是偶数则可以继续往下检验 ……

Miller-Rabin

基本原理是费马小定理：若 p 是质数，a, p 互质，则 $a^{p-1} \equiv 1 \pmod{p}$。

于是对于某个 p，若能找到与它互质的 a 使得 $a^{p-1} \neq 1 \pmod{p}$，则 p 必不是质数。

然而有一些合数 p，满足所有与它互质的 a 都有 $a^{p-1} \equiv 1 \pmod{p}$，这种数称为 Carmichael 数（如 $561 = 3 * 11 * 17$），这样的数是用上面的方法检验不出来的。

所以还需要奇素数判定。对于奇素数 p，如果 $a^{p-1} \equiv 1 \pmod{p}$ 即 $(a^{\frac{p-1}{2}} + 1)(a^{\frac{p-1}{2}} - 1) \equiv 0 \pmod{p}$，由于 F_p 是整环，所以 $a^{\frac{p-1}{2}} \equiv 1$ 或 $p-1$。如果 $\frac{p-1}{2}$ 还是偶数则可以继续往下检验 ……

用原根的一些理论可以证明这样就能保证对于任意合数至少存在一个 a 可以判定它是合数。

Contents

两种常见的质因子分解方法

两种常见的质因子分解方法

试除法，复杂度 $O(\sqrt{n})$，太慢了。

两种常见的质因子分解方法

试除法，复杂度 $O(\sqrt{n})$，太慢了。
Pollard's Rho，期望复杂度 $O(\sqrt[4]{n}\log n)$，又名启发式分解。

为什么叫启发式分解?

为什么叫启发式分解?

n 是素数的时候用 Miller-Rabin 素性测试。

为什么叫启发式分解?

n 是素数的时候用 Miller-Rabin 素性测试。

n 不是素数的时候，复杂度只和最小的质因子有关。所以称为启发式。

Pollards' Rho

Pollards' Rho

假如要分解一个数 n:

Pollards' Rho

假如要分解一个数 n:
首先进行素性测试，是素数直接返回。

Pollards' Rho

假如要分解一个数 n:

首先进行素性测试，是素数直接返回。

否则就要生成一些随机的 x_i，去求 $gcd(|x_i - x_j|, n)$，如果这个 $\in (1, n)$ 则找到了 n 的一个因子，递归分解。

Pollards' Rho

假如要分解一个数 n:

首先进行素性测试，是素数直接返回。

否则就要生成一些随机的 x_i，去求 $gcd(|x_i - x_j|, n)$，如果这个 $\in (1, n)$ 则找到了 n 的一个因子，递归分解。

一个挺靠谱的随机方法就是 $x \leftarrow x^2 + c$，c 是个随机数。

Pollards' Rho

假如要分解一个数 n:

首先进行素性测试，是素数直接返回。

否则就要生成一些随机的 x_i，去求 $gcd(|x_i - x_j|, n)$，如果这个 $\in (1, n)$ 则找到了 n 的一个因子，递归分解。

一个挺靠谱的随机方法就是 $x \leftarrow x^2 + c$，c 是个随机数。

这样随机出来的 x 可能会进入循环，假如进入循环了我们还没找到因子，就重新随个 x 和 c，重新做。

如何判已经进入循环?

如何判已经进入循环?

可以证明 $x \leftarrow x^2 + c$，形成的一定是一个 ρ 形结构。

如何判已经进入循环?

可以证明 $x \leftarrow x^2 + c$，形成的一定是一个 ρ 形结构。

每次当 i 为 2 的幂次的时候就令 $y \leftarrow x_i$，如果某时刻 $x_i = y$ 了则说明已经在环上绕了一圈了。

如何判已经进入循环?

可以证明 $x \leftarrow x^2 + c$，形成的一定是一个 ρ 形结构。

每次当 i 为 2 的幂次的时候就令 $y \leftarrow x_i$，如果某时刻 $x_i = y$ 了则说明已经在环上绕了一圈了。

即：看 $x_{(2,4]}$ 是否 $= x_2$，看 $x_{(4,8]}$ 是否 $= x_4$，看 $x_{(8,16]}$ 是否 $= x_8 \cdots\cdots$

如何判已经进入循环?

可以证明 $x \leftarrow x^2 + c$，形成的一定是一个 ρ 形结构。

每次当 i 为 2 的幂次的时候就令 $y \leftarrow x_i$，如果某时刻 $x_i = y$ 了则说明已经在环上绕了一圈了。

即：看 $x_{(2,4]}$ 是否 $= x_2$，看 $x_{(4,8]}$ 是否 $= x_4$，看 $x_{(8,16]}$ 是否 $= x_8$……

这样“浪费”的步数仅仅是 $O($ 环长 $)$ 级别的。

Contents

Contents

欧几里德求 gcd

欧几里德求 gcd

若 $x|a, x|b$，则 $x|a+b$。

欧几里德求 gcd

若 $x|a, x|b$，则 $x|a+b$。
于是 $gcd(a,b) = gcd(b, a\%b)$。

扩展欧几里德算法

扩展欧几里德算法

已知 a, b，求出 x, y 满足 $ax + by = gcd(a, b)$。

扩展欧几里德算法

已知 a, b，求出 x, y 满足 $ax + by = gcd(a, b)$。

在欧几里德算法中递归地求：若已有 $b = 0$，则 $gcd = a$，令 $x = 1, y = 0$。

扩展欧几里德算法

已知 a, b，求出 x, y 满足 $ax + by = gcd(a, b)$。

在欧几里德算法中递归地求：若已有 $b = 0$，则 $gcd = a$，令 $x = 1, y = 0$。

否则求出 x', y' 满足

$bx' + (a - a/b * b) * y' = gcd(b, a\%b) = gcd(a, b)$。

扩展欧几里德算法

已知 a, b，求出 x, y 满足 $ax + by = gcd(a, b)$。

在欧几里德算法中递归地求：若已有 $b = 0$，则 $gcd = a$，令 $x = 1, y = 0$。

否则求出 x', y' 满足

$bx' + (a - a/b * b) * y' = gcd(b, a\%b) = gcd(a, b)$。

于是 $a * y' + b * (x' - a/b * y') = gcd(a, b)$。

辗转相减（除）的其它用处

辗转相减（除）的其它用处

只要一个环定义了带余除法，就可以在上面辗转相减（除）。

辗转相减（除）的其它用处

只要一个环定义了带余除法，就可以在上面辗转相减（除）。比如模合数的环、多项式环等等。

辗转相减（除）的其它用处

只要一个环定义了带余除法，就可以在上面辗转相减（除）。比如模合数的环、多项式环等等。

比如求行列式模一个合数，根据行列式的性质可以把一行的倍数加到另一行上。辗转相减把其中一个位置消成 0 即可。

辗转相减（除）的其它用处

只要一个环定义了带余除法，就可以在上面辗转相减（除）。比如模合数的环、多项式环等等。

比如求行列式模一个合数，根据行列式的性质可以把一行的倍数加到另一行上。辗转相减把其中一个位置消成 0 即可。

比如求两个多项式的最大公约式，或者说多项式取模，可以不停地把一个的倍数加到另一个上，把其中一个多项式变成 0。

类欧几里德算法

类欧几里德算法

最基本的模型：$solve(n, A, B, C) = \sum_{i=1}^{n} \lfloor \frac{Ai+B}{C} \rfloor$。

类欧几里德算法

最基本的模型：$solve(n, A, B, C) = \sum_{i=1}^{n} \lfloor \frac{Ai+B}{C} \rfloor$。

如果 $A \geq C$，$ans+ = \frac{n(n+1)}{2} * (A/C)$，然后 $A\% = C$。

类欧几里德算法

最基本的模型：$solve(n, A, B, C) = \sum_{i=1}^{n} \lfloor \frac{Ai+B}{C} \rfloor$。

如果 $A \geq C$，$ans+ = \frac{n(n+1)}{2} * (A/C)$，然后 $A\% = C$。

如果 $|B| \geq C$，讨论下正负号算下，然后把 B 搞到 $[0, C)$ 之间。

类欧几里德算法

最基本的模型：$solve(n, A, B, C) = \sum_{i=1}^{n} \lfloor \frac{Ai+B}{C} \rfloor$。

如果 $A \geq C$，$ans+ = \frac{n(n+1)}{2} * (A/C)$，然后 $A\% = C$。

如果 $|B| \geq C$，讨论下正负号算下，然后把 B 搞到 $[0, C)$ 之间。

设 $m = \lfloor \frac{An+B}{C} \rfloor$，则要算 $\sum_{i=1}^{n} \sum_{j=1}^{m} [Cj \leq Ai + B]$。

类欧几里德算法

最基本的模型：$solve(n, A, B, C) = \sum_{i=1}^{n} \lfloor \frac{Ai+B}{C} \rfloor$。

如果 $A \geq C$，$ans+ = \frac{n(n+1)}{2} * (A/C)$，然后 $A\% = C$。

如果 $|B| \geq C$，讨论下正负号算下，然后把 B 搞到 $[0, C)$ 之间。

设 $m = \lfloor \frac{An+B}{C} \rfloor$，则要算 $\sum_{i=1}^{n} \sum_{j=1}^{m} [Cj \leq Ai + B]$。

即 $nm - \sum_{j=1}^{m} \sum_{i=1}^{n} [Ai \leq Cj - B - 1]$。

类欧几里德算法

最基本的模型：$solve(n, A, B, C) = \sum_{i=1}^{n} \lfloor \frac{Ai+B}{C} \rfloor$。

如果 $A \geq C$，$ans+ = \frac{n(n+1)}{2} * (A/C)$，然后 $A\% = C$。

如果 $|B| \geq C$，讨论下正负号算下，然后把 B 搞到 $[0, C)$ 之间。

设 $m = \lfloor \frac{An+B}{C} \rfloor$，则要算 $\sum_{i=1}^{n} \sum_{j=1}^{m} [Cj \leq Ai + B]$。

即 $nm - \sum_{j=1}^{m} \sum_{i=1}^{n} [Ai \leq Cj - B - 1]$。

即 $nm - solve(m, C, -B - 1, A)$。

类欧几里德算法

最基本的模型：$solve(n, A, B, C) = \sum_{i=1}^{n} \lfloor \frac{Ai+B}{C} \rfloor$。

如果 $A \geq C$，$ans+ = \frac{n(n+1)}{2} * (A/C)$，然后 $A\% = C$。

如果 $|B| \geq C$，讨论下正负号算下，然后把 B 搞到 $[0, C)$ 之间。

设 $m = \lfloor \frac{An+B}{C} \rfloor$，则要算 $\sum_{i=1}^{n} \sum_{j=1}^{m} [Cj \leq Ai + B]$。

即 $nm - \sum_{j=1}^{m} \sum_{i=1}^{n} [Ai \leq Cj - B - 1]$。

即 $nm - solve(m, C, -B - 1, A)$。

每次 A 对 C 取模后互换位置，复杂度同欧几里德算法，为 $O(\log C)$。

Contents

中国剩余定理

中国剩余定理

有 n 个方程 $x \equiv a_i \pmod{p_i}$，p_i 两两互质，求 x。

中国剩余定理

有 n 个方程 $x \equiv a_i \pmod{p_i}$，p_i 两两互质，求 x。
设 $w_i = \prod_{j \neq i} p_j$，则答案 $\equiv \sum_{i=1}^{n} a_i * w_i * inv(w_i, p_i)$。

Ex 中国剩余定理

Ex 中国剩余定理

如果模数不互质，只要两两合并方程即可。

Ex 中国剩余定理

如果模数不互质，只要两两合并方程即可。

比如 $x \equiv a_1 \pmod{p_1}, x \equiv a_2 \pmod{p_2}$，设 $d = gcd(p_1, p_2)$，那么必须 $a_1 \equiv a_2 \pmod{d}$。

Ex 中国剩余定理

如果模数不互质，只要两两合并方程即可。

比如 $x \equiv a_1 \pmod{p_1}, x \equiv a_2 \pmod{p_2}$，设 $d = gcd(p_1, p_2)$，那么必须 $a_1 \equiv a_2 \pmod{d}$。然后答案一定可以表示成 $w * d + (a_1 \bmod d)$。

Ex 中国剩余定理

如果模数不互质，只要两两合并方程即可。

比如 $x \equiv a_1 \pmod{p_1}, x \equiv a_2 \pmod{p_2}$，设 $d = gcd(p_1, p_2)$，那么必须 $a_1 \equiv a_2 \pmod{d}$。然后答案一定可以表示成 $w * d + (a_1 \bmod d)$。

用普通中国剩余定理求出 w 即可。

Ex 中国剩余定理

如果模数不互质，只要两两合并方程即可。

比如 $x \equiv a_1 \pmod{p_1}, x \equiv a_2 \pmod{p_2}$，设 $d = gcd(p_1, p_2)$，那么必须 $a_1 \equiv a_2 \pmod{d}$。然后答案一定可以表示成 $w * d + (a_1 \bmod d)$。

用普通中国剩余定理求出 w 即可。

$w \equiv (a_1/d) \pmod{p_1/d}, w \equiv (a_2/d) \pmod{p_2/d}$。

Ex 中国剩余定理

如果模数不互质，只要两两合并方程即可。

比如 $x \equiv a_1 \pmod{p_1}, x \equiv a_2 \pmod{p_2}$，设 $d = gcd(p_1, p_2)$，那么必须 $a_1 \equiv a_2 \pmod{d}$。然后答案一定可以表示成 $w * d + (a_1 \bmod d)$。

用普通中国剩余定理求出 w 即可。

$w \equiv (a_1/d) \pmod{p_1/d}, w \equiv (a_2/d) \pmod{p_2/d}$。

每次把两个方程合并成一个模数是它们 lcm 的方程。

Contents

BSGS

BSGS

已知 A, B, C，求 x 使 $A^x \equiv B \pmod{C}$。

BSGS

已知 A, B, C，求 x 使 $A^x \equiv B \pmod{C}$。

如果 C 是合数可以拆成质数的幂，然后中国剩余定理合并起来。于是考虑 $C = p^c$。先假设 A, C 互质。

BSGS

已知 A, B, C，求 x 使 $A^x \equiv B \pmod{C}$。

如果 C 是合数可以拆成质数的幂，然后中国剩余定理合并起来。于是考虑 $C = p^c$。先假设 A, C 互质。

令 $S = \sqrt{C}$，如果有 $x = k_1 S - k_2$，有 $(A^S)^{k_1} \equiv BA^{k_2} \pmod{C}$。

BSGS

已知 A, B, C，求 x 使 $A^x \equiv B \pmod{C}$。

如果 C 是合数可以拆成质数的幂，然后中国剩余定理合并起来。于是考虑 $C = p^c$。先假设 A, C 互质。

令 $S = \sqrt{C}$，如果有 $x = k_1 S - k_2$，有 $(A^S)^{k_1} \equiv BA^{k_2} \pmod{C}$。

把一边放入 Hash 表，另一边查询即可。复杂度 $O(\sqrt{C})$。

ExBSGS

ExBSGS

如果 $d = gcd(A, C) > 1$，那么有 $(A/d) * A^{x-1} \equiv B/d \pmod{C/d}$。

ExBSGS

如果 $d = gcd(A, C) > 1$，那么有 $(A/d) * A^{x-1} \equiv B/d \pmod{C/d}$。

继续除以 gcd，直到 $gcd = 1$，有 $\frac{A^t}{\prod d_i} * A^{x-t} \equiv \frac{B}{\prod d_i} \pmod{\frac{C}{\prod d_i}}$。

ExBSGS

如果 $d = gcd(A, C) > 1$，那么有 $(A/d) * A^{x-1} \equiv B/d \pmod{C/d}$。

继续除以 gcd，直到 $gcd = 1$，有 $\frac{A^t}{\prod d_i} * A^{x-t} \equiv \frac{B}{\prod d_i} \pmod{\frac{C}{\prod d_i}}$。

做普通的 BSGS 即可。

ExBSGS

如果 $d = gcd(A, C) > 1$，那么有 $(A/d) * A^{x-1} \equiv B/d \pmod{C/d}$。

继续除以 gcd，直到 $gcd = 1$，有 $\frac{A^t}{\prod d_i} * A^{x-t} \equiv \frac{B}{\prod d_i} \pmod{\frac{C}{\prod d_i}}$。

做普通的 BSGS 即可。

还需要检查下 $x = [0, t)$ 是否是原问题的解。

Contents

群、环、域

群、环、域

群：非空集合 G 上定义了一种二元运算，满足封闭性、结合律、单位元、逆元。

群、环、域

群：非空集合 G 上定义了一种二元运算，满足封闭性、结合律、单位元、逆元。

环：非空集合 R 上定义了加法和乘法，在加法下构成交换群，满足乘法结合律、分配律。

群、环、域

群：非空集合 G 上定义了一种二元运算，满足封闭性、结合律、单位元、逆元。

环：非空集合 R 上定义了加法和乘法，在加法下构成交换群，满足乘法结合律、分配律。

域：非零元素都可逆的交换幺环。

缩系，原根

缩系，原根

循环群：指群可以由一个元素生成：$G = x, x^2, x^3 \ldots$。

缩系，原根

循环群：指群可以由一个元素生成：$G = x, x^2, x^3 \ldots$。

阶：满足 $x^d = 1$ 的最小正整数 d。记为 $ord(x)$。$x^m = 1$ 当且仅当 $ord(x) | m$。

缩系，原根

循环群：指群可以由一个元素生成：$G = x, x^2, x^3 \ldots$。

阶：满足 $x^d = 1$ 的最小正整数 d。记为 $ord(x)$。$x^m = 1$ 当且仅当 $ord(x) | m$。

模素数 p 的剩余类构成一个有限域。

缩系，原根

循环群：指群可以由一个元素生成：$G = x, x^2, x^3 \ldots$。

阶：满足 $x^d = 1$ 的最小正整数 d。记为 $ord(x)$。$x^m = 1$ 当且仅当 $ord(x) | m$。

模素数 p 的剩余类构成一个有限域。

模 m 意义下与 m 互质的元素组成缩系，大小为 $\phi(m)$。

缩系，原根

循环群：指群可以由一个元素生成：$G = x, x^2, x^3 \ldots$。

阶：满足 $x^d = 1$ 的最小正整数 d。记为 $ord(x)$。$x^m = 1$ 当且仅当 $ord(x) | m$。

模素数 p 的剩余类构成一个有限域。

模 m 意义下与 m 互质的元素组成缩系，大小为 $\phi(m)$。

原根：能生成缩系的元素，即 x^i 两两不同（$0 \leq i < \phi(m)$）的 x。原根不一定存在。事实上，当且仅当 $m = 2, 4, p^k, 2 * p^k$ 时模 m 缩系的原根存在。p 是任意奇质数。

模质数 p 域下原根的存在性

模质数 p 域下原根的存在性

Fact: 设 a 的阶是 m，$d|m$，则 a^d 的阶是 $\frac{m}{d}$。

模质数 p 域下原根的存在性

Fact: 设 a 的阶是 m，$d|m$，则 a^d 的阶是 $\frac{m}{d}$。

Fact: 设 a 的阶是 m，b 的阶是 n，则必存在一个数，阶是 $lcm(n, m)$。

模质数 p 域下原根的存在性

Fact: 设 a 的阶是 m，$d|m$，则 a^d 的阶是 $\frac{m}{d}$。

Fact: 设 a 的阶是 m，b 的阶是 n，则必存在一个数，阶是 $lcm(n, m)$。

先考虑 n, m 互质，设 ab 的阶是 e，则有 $1 = (ab)^{me} = b^{me}$，于是 $n|me$，于是 $n|e$。

模质数 p 域下原根的存在性

Fact: 设 a 的阶是 m，$d|m$，则 a^d 的阶是 $\frac{m}{d}$。

Fact: 设 a 的阶是 m，b 的阶是 n，则必存在一个数，阶是 $lcm(n, m)$。

先考虑 n, m 互质，设 ab 的阶是 e，则有 $1 = (ab)^{me} = b^{me}$，于是 $n|me$，于是 $n|e$。同理有 $m|e$，于是 $nm|e$。

模质数 p 域下原根的存在性

Fact: 设 a 的阶是 m，$d|m$，则 a^d 的阶是 $\frac{m}{d}$。

Fact: 设 a 的阶是 m，b 的阶是 n，则必存在一个数，阶是 $lcm(n, m)$。

先考虑 n, m 互质，设 ab 的阶是 e，则有 $1 = (ab)^{me} = b^{me}$，于是 $n|me$，于是 $n|e$。同理有 $m|e$，于是 $nm|e$。而 $(ab)^{nm} = 1$，于是 $e|nm$，于是 $e = nm$。

模质数 p 域下原根的存在性

Fact: 设 a 的阶是 m，$d|m$，则 a^d 的阶是 $\frac{m}{d}$。

Fact: 设 a 的阶是 m，b 的阶是 n，则必存在一个数，阶是 $lcm(n,m)$。

先考虑 n,m 互质，设 ab 的阶是 e，则有 $1=(ab)^{me}=b^{me}$，于是 $n|me$，于是 $n|e$。同理有 $m|e$，于是 $nm|e$。而 $(ab)^{nm}=1$，于是 $e|nm$，于是 $e=nm$。

如果 n,m 不互质，只要取 $n'|n, m'|m, n', m'$ 互质且 $n'm'=lcm(n,m)$（比如把每个质因子分到 n', m' 中）

模质数 p 域下原根的存在性

Fact: 设 a 的阶是 m，$d|m$，则 a^d 的阶是 $\frac{m}{d}$。

Fact: 设 a 的阶是 m，b 的阶是 n，则必存在一个数，阶是 $lcm(n,m)$。

先考虑 n,m 互质，设 ab 的阶是 e，则有 $1=(ab)^{me}=b^{me}$，于是 $n|me$，于是 $n|e$。同理有 $m|e$，于是 $nm|e$。而 $(ab)^{nm}=1$，于是 $e|nm$，于是 $e=nm$。

如果 n,m 不互质，只要取 $n'|n,m'|m,n',m'$ 互质且 $n'm'=lcm(n,m)$（比如把每个质因子分到 n',m' 中），此时 $a^{\frac{n}{n'}}$ 阶为 n'，$b^{\frac{m}{m'}}$ 阶为 m'。乘起来即可。

模质数 p 域下原根的存在性

Fact: 设 a 的阶是 m，$d|m$，则 a^d 的阶是 $\frac{m}{d}$。

Fact: 设 a 的阶是 m，b 的阶是 n，则必存在一个数，阶是 $lcm(n,m)$。

先考虑 n,m 互质，设 ab 的阶是 e，则有 $1=(ab)^{me}=b^{me}$，于是 $n|me$，于是 $n|e$。同理有 $m|e$，于是 $nm|e$。而 $(ab)^{nm}=1$，于是 $e|nm$，于是 $e=nm$。

如果 n,m 不互质，只要取 $n'|n,m'|m,n',m'$ 互质且 $n'm'=lcm(n,m)$（比如把每个质因子分到 n',m' 中），此时 $a^{\frac{n}{n'}}$ 阶为 n'，$b^{\frac{m}{m'}}$ 阶为 m'。乘起来即可。

这样就说明一定存在一个数，阶是所有元素的阶的倍数 d。

模质数 p 域下原根的存在性

Fact: 设 a 的阶是 m，$d|m$，则 a^d 的阶是 $\frac{m}{d}$。

Fact: 设 a 的阶是 m，b 的阶是 n，则必存在一个数，阶是 $lcm(n,m)$。

先考虑 n,m 互质，设 ab 的阶是 e，则有 $1=(ab)^{me}=b^{me}$，于是 $n|me$，于是 $n|e$。同理有 $m|e$，于是 $nm|e$。而 $(ab)^{nm}=1$，于是 $e|nm$，于是 $e=nm$。

如果 n,m 不互质，只要取 $n'|n,m'|m,n',m'$ 互质且 $n'm'=lcm(n,m)$（比如把每个质因子分到 n',m' 中），此时 $a^{\frac{n}{n'}}$ 阶为 n'，$b^{\frac{m}{m'}}$ 阶为 m'。乘起来即可。

这样就说明一定存在一个数，阶是所有元素的阶的倍数 d。由 $x^{p-1}=1$ 知 $d|p-1$，而 $x^d=1$ 有 $p-1$ 个不同的根。

模质数 p 域下原根的存在性

Fact: 设 a 的阶是 m，$d|m$，则 a^d 的阶是 $\frac{m}{d}$。

Fact: 设 a 的阶是 m，b 的阶是 n，则必存在一个数，阶是 $lcm(n,m)$。

先考虑 n,m 互质，设 ab 的阶是 e，则有 $1=(ab)^{me}=b^{me}$，于是 $n|me$，于是 $n|e$。同理有 $m|e$，于是 $nm|e$。而 $(ab)^{nm}=1$，于是 $e|nm$，于是 $e=nm$。

如果 n,m 不互质，只要取 $n'|n,m'|m,n',m'$ 互质且 $n'm'=lcm(n,m)$（比如把每个质因子分到 n',m' 中），此时 $a^{\frac{n}{n'}}$ 阶为 n'，$b^{\frac{m}{m'}}$ 阶为 m'。乘起来即可。

这样就说明一定存在一个数，阶是所有元素的阶的倍数 d。由 $x^{p-1}=1$ 知 $d|p-1$，而 $x^d=1$ 有 $p-1$ 个不同的根。由于是域，d 至少为 $p-1$，于是 $d=p-1$。即 F_p 的乘法群是个循环群，即原根存在。

原根的个数

原根的个数

Fact:n 个元素的循环群的生成元个数为 $\phi(n)$。

原根的个数

Fact:n 个元素的循环群的生成元个数为 $\phi(n)$。
取一个生成元 g，考虑 g^r 的阶 e。

原根的个数

Fact:n 个元素的循环群的生成元个数为 $\phi(n)$。

取一个生成元 g，考虑 g^r 的阶 e。设 $d = \gcd(n, r)$，有 $(g^r)^{\frac{n}{d}} = (g^n)^{\frac{r}{d}} = 1$，于是 $e | \frac{n}{d}$。

原根的个数

Fact:n 个元素的循环群的生成元个数为 $\phi(n)$。

取一个生成元 g，考虑 g^r 的阶 e。设 $d = \gcd(n, r)$，有 $(g^r)^{\frac{n}{d}} = (g^n)^{\frac{r}{d}} = 1$，于是 $e|\frac{n}{d}$。

而若有 $(g^r)^e = 1$ 则 $n|re$，于是 $\frac{n}{d}|e$，于是 $e = \frac{n}{d}$。

原根的个数

Fact:n 个元素的循环群的生成元个数为 $\phi(n)$。

取一个生成元 g，考虑 g^r 的阶 e。设 $d = \gcd(n, r)$，有 $(g^r)^{\frac{n}{d}} = (g^n)^{\frac{r}{d}} = 1$，于是 $e|\frac{n}{d}$。

而若有 $(g^r)^e = 1$ 则 $n|re$，于是 $\frac{n}{d}|e$，于是 $e = \frac{n}{d}$。

当且仅当 $e = n$ 即 $\gcd(n, r) = 1$ 时 g^r 也是生成元，于是恰有 $\phi(n)$ 个。一般地，阶为 $d|n$ 的元素恰有 $\phi(d)$ 个 (即求有多少 r 使 $\frac{n}{\gcd(r,n)} = d$)。

原根的个数

Fact:n 个元素的循环群的生成元个数为 $\phi(n)$。

取一个生成元 g，考虑 g^r 的阶 e。设 $d = \gcd(n, r)$，有 $(g^r)^{\frac{n}{d}} = (g^n)^{\frac{r}{d}} = 1$，于是 $e|\frac{n}{d}$。

而若有 $(g^r)^e = 1$ 则 $n|re$，于是 $\frac{n}{d}|e$，于是 $e = \frac{n}{d}$。

当且仅当 $e = n$ 即 $\gcd(n, r) = 1$ 时 g^r 也是生成元，于是恰有 $\phi(n)$ 个。一般地，阶为 $d|n$ 的元素恰有 $\phi(d)$ 个 (即求有多少 r 使 $\frac{n}{\gcd(r,n)} = d$)。

如果原根存在，即缩系是大小为 $\phi(m)$ 的循环群，那么原根个数为 $\phi(\phi(m))$。特别地，模质数域的原根个数为 $\phi(p-1)$。

原根的个数

Fact:n 个元素的循环群的生成元个数为 $\phi(n)$。

取一个生成元 g，考虑 g^r 的阶 e。设 $d=\gcd(n,r)$，有 $(g^r)^{\frac{n}{d}}=(g^n)^{\frac{r}{d}}=1$，于是 $e|\frac{n}{d}$。

而若有 $(g^r)^e=1$ 则 $n|re$，于是 $\frac{n}{d}|e$，于是 $e=\frac{n}{d}$。

当且仅当 $e=n$ 即 $\gcd(n,r)=1$ 时 g^r 也是生成元，于是恰有 $\phi(n)$ 个。一般地，阶为 $d|n$ 的元素恰有 $\phi(d)$ 个 (即求有多少 r 使 $\frac{n}{\gcd(r,n)}=d$)。

如果原根存在，即缩系是大小为 $\phi(m)$ 的循环群，那么原根个数为 $\phi(\phi(m))$。特别地，模质数域的原根个数为 $\phi(p-1)$。

最小的原根一般不大，可以暴力枚举判断。求出一个原根 g 后，全部原根的集合就是 $\{g^r|\gcd(n,r)=1\}$。

Contents

勒让德符号

勒让德符号

记 $\left(\frac{a}{p}\right)$ 为勒让德符号，p 是一个奇素数。

勒让德符号

记 $\left(\frac{a}{p}\right)$ 为勒让德符号，p 是一个奇素数。

$\left(\frac{a}{p}\right) \equiv 1$ 表示 a 是模 p 域下的二次剩余，-1 表示是二次非剩余。

勒让德符号

记 $\left(\frac{a}{p}\right)$ 为勒让德符号，p 是一个奇素数。

$\left(\frac{a}{p}\right) \equiv 1$ 表示 a 是模 p 域下的二次剩余，-1 表示是二次非剩余。

计算方法（欧拉准则）：$\left(\frac{a}{p}\right) \equiv a^{\frac{p-1}{2}} \pmod{p}$。

勒让德符号

记 $\left(\frac{a}{p}\right)$ 为勒让德符号，p 是一个奇素数。

$\left(\frac{a}{p}\right) \equiv 1$ 表示 a 是模 p 域下的二次剩余，-1 表示是二次非剩余。

计算方法（欧拉准则）：$\left(\frac{a}{p}\right) \equiv a^{\frac{p-1}{2}} \pmod{p}$。

二次互反律：对于奇素数 p, q，有 $\left(\frac{p}{q}\right)\left(\frac{q}{p}\right) = (-1)^{\frac{(p-1)(q-1)}{4}}$

计算二次剩余的方法

计算二次剩余的方法

1. 原根法：如果存在原根，可以求出原根并 BSGS 求出指数，除以 2 即可。复杂度 $O(\sqrt{p})$。

计算二次剩余的方法

1. 原根法：如果存在原根，可以求出原根并 BSGS 求出指数，除以 2 即可。复杂度 $O(\sqrt{p})$。
2. Tonelli–Shanks 算法：复杂度 $O(\log^2 p)$。这里不作介绍。

计算二次剩余的方法

1. 原根法：如果存在原根，可以求出原根并 BSGS 求出指数，除以 2 即可。复杂度 $O(\sqrt{p})$。
2. Tonelli–Shanks 算法：复杂度 $O(\log^2 p)$。这里不作介绍。
3. Cipolla 算法：复杂度 $O(\log p)$。

Cipolla 算法

Cipolla 算法

要求 $x^2 \equiv n \pmod{p}$ 的解（$\left(\frac{n}{p}\right) \equiv n^{\frac{p-1}{2}} \equiv 1$）。

Cipolla 算法

要求 $x^2 \equiv n \pmod{p}$ 的解（$\left(\frac{n}{p}\right) \equiv n^{\frac{p-1}{2}} \equiv 1$）。

随机一个 a 使得 $a^2 - n$ 是二次非剩余，即 $(a^2 - n)^{\frac{p-1}{2}} \equiv -1$。

Cipolla 算法

要求 $x^2 \equiv n \pmod{p}$ 的解（$\left(\frac{n}{p}\right) \equiv n^{\frac{p-1}{2}} \equiv 1$）。

随机一个 a 使得 $a^2 - n$ 是二次非剩余，即 $(a^2 - n)^{\frac{p-1}{2}} \equiv -1$。设 $\omega = \sqrt{a^2 - n}$。可证 $x + y\omega$ 构成一个域。

Cipolla 算法

要求 $x^2 \equiv n \pmod{p}$ 的解（$\left(\frac{n}{p}\right) \equiv n^{\frac{p-1}{2}} \equiv 1$）。

随机一个 a 使得 $a^2 - n$ 是二次非剩余，即 $(a^2 - n)^{\frac{p-1}{2}} \equiv -1$。设 $\omega = \sqrt{a^2 - n}$。可证 $x + y\omega$ 构成一个域。

构造得二次剩余的解为 $(a + \omega)^{\frac{p+1}{2}}$。

Cipolla 算法

要求 $x^2 \equiv n \pmod{p}$ 的解（$\left(\frac{n}{p}\right) \equiv n^{\frac{p-1}{2}} \equiv 1$）。

随机一个 a 使得 $a^2 - n$ 是二次非剩余，即 $(a^2 - n)^{\frac{p-1}{2}} \equiv -1$。设 $\omega = \sqrt{a^2 - n}$。可证 $x + y\omega$ 构成一个域。

构造得二次剩余的解为 $(a + \omega)^{\frac{p+1}{2}}$。

证明：$(a + \omega)^{p+1} \equiv (a^p + \omega^p)(a + \omega) \equiv (a - \omega)(a + \omega) \equiv n \pmod{p}$。

Cipolla 算法

要求 $x^2 \equiv n \pmod{p}$ 的解（$\left(\frac{n}{p}\right) \equiv n^{\frac{p-1}{2}} \equiv 1$）。

随机一个 a 使得 $a^2 - n$ 是二次非剩余，即 $(a^2 - n)^{\frac{p-1}{2}} \equiv -1$。设 $\omega = \sqrt{a^2 - n}$。可证 $x + y\omega$ 构成一个域。

构造得二次剩余的解为 $(a + \omega)^{\frac{p+1}{2}}$。

证明：$(a + \omega)^{p+1} \equiv (a^p + \omega^p)(a + \omega) \equiv (a - \omega)(a + \omega) \equiv n \pmod{p}$。

并且 $(a + \omega)^{\frac{p+1}{2}}$ 不存在 ω 项：

Cipolla 算法

要求 $x^2 \equiv n \pmod{p}$ 的解（$\left(\frac{n}{p}\right) \equiv n^{\frac{p-1}{2}} \equiv 1$）。

随机一个 a 使得 $a^2 - n$ 是二次非剩余，即 $(a^2 - n)^{\frac{p-1}{2}} \equiv -1$。设 $\omega = \sqrt{a^2 - n}$。可证 $x + y\omega$ 构成一个域。

构造得二次剩余的解为 $(a + \omega)^{\frac{p+1}{2}}$。

证明：$(a + \omega)^{p+1} \equiv (a^p + \omega^p)(a + \omega) \equiv (a - \omega)(a + \omega) \equiv n \pmod{p}$。

并且 $(a + \omega)^{\frac{p+1}{2}}$ 不存在 ω 项：设 $(x + y\omega)^2 \equiv x^2 + (a^2 - n)y^2 + (2xy)\omega \equiv n$，则有 $xy \equiv 0$。

Cipolla 算法

要求 $x^2 \equiv n \pmod p$ 的解（$\left(\frac{n}{p}\right) \equiv n^{\frac{p-1}{2}} \equiv 1$）。

随机一个 a 使得 $a^2 - n$ 是二次非剩余，即 $(a^2-n)^{\frac{p-1}{2}} \equiv -1$。设 $\omega = \sqrt{a^2-n}$。可证 $x + y\omega$ 构成一个域。

构造得二次剩余的解为 $(a+\omega)^{\frac{p+1}{2}}$。

证明：$(a+\omega)^{p+1} \equiv (a^p + \omega^p)(a+\omega) \equiv (a-\omega)(a+\omega) \equiv n \pmod p$。

并且 $(a+\omega)^{\frac{p+1}{2}}$ 不存在 ω 项：设 $(x+y\omega)^2 \equiv x^2 + (a^2-n)y^2 + (2xy)\omega \equiv n$，则有 $xy \equiv 0$。若 $y \neq 0$ 则有 $(a^2-n) \equiv \frac{n}{y^2}$，左边没有二次剩余而右边有。矛盾。

Contents

Chinese leftovers II

PE 552

题意

考虑一系列同余方程 $x \equiv i \pmod{p_i}$，其中 p_i 是第 i 个质数。设满足前 i 个方程的最小非负解是 A_i。给出 m，求对于所有不超过 m 的质数，满足至少整除一个 A_i 的质数的和。

范围

$m = 300000$。

Chinese leftovers II

PE 552

Chinese leftovers II

PE 552

从前往后合并同余方程，维护 A_i 模所有质数的值，顺便算答案。

Chinese leftovers II

PE 552

从前往后合并同余方程，维护 A_i 模所有质数的值，顺便算答案。

考虑两个同余方程：

$$x \equiv a_1 \pmod{p_1}$$

$$x \equiv a_2 \pmod{p_2}$$

Chinese leftovers II

PE 552

从前往后合并同余方程，维护 A_i 模所有质数的值，顺便算答案。

考虑两个同余方程：

$$x \equiv a_1 \pmod{p_1}$$

$$x \equiv a_2 \pmod{p_2}$$

传统中国剩余定理：

$$x \equiv a_1 * p_2 * inv(p_2, p_1) + a_2 * p_1 * inv(p_1, p_2) \pmod{p_1 p_2}$$

Chinese leftovers II

PE 552

从前往后合并同余方程，维护 A_i 模所有质数的值，顺便算答案。

考虑两个同余方程：

$$x \equiv a_1 \pmod{p_1}$$

$$x \equiv a_2 \pmod{p_2}$$

传统中国剩余定理：

$$x \equiv a_1 * p_2 * inv(p_2, p_1) + a_2 * p_1 * inv(p_1, p_2) \pmod{p_1 p_2}$$

在这题中，p_1 是前 $i-1$ 个质数之积，p_2 是第 i 个质数。$inv(p_2, p_1)$ 不好求。

Chinese leftovers II

PE 552

Chinese leftovers II

PE 552

换一个中国剩余定理的姿势:

Chinese leftovers II

PE 552

换一个中国剩余定理的姿势:

$$a_1 + (a_2 - a_1) * p_1 * inv(p_1, p_2) \pmod{p_1 p_2}$$

Chinese leftovers II

PE 552

换一个中国剩余定理的姿势:

$$a_1 + (a_2 - a_1) * p_1 * inv(p_1, p_2) \pmod{p_1 p_2}$$

如果把 $(a_2 - a_1) * inv(p_1, p_2)$ 模去 p_2，上面这个式子甚至不用取模。

Chinese leftovers II

PE 552

换一个中国剩余定理的姿势:

$$a_1 + (a_2 - a_1) * p_1 * inv(p_1, p_2) \pmod{p_1 p_2}$$

如果把 $(a_2 - a_1) * inv(p_1, p_2)$ 模去 p_2，上面这个式子甚至不用取模。

那么相当于每次答案加上 p_1 乘以一个模过 p_2 的数 q。

Chinese leftovers II

PE 552

换一个中国剩余定理的姿势:

$$a_1 + (a_2 - a_1) * p_1 * inv(p_1, p_2) \pmod{p_1 p_2}$$

如果把 $(a_2 - a_1) * inv(p_1, p_2)$ 模去 p_2，上面这个式子甚至不用取模。

那么相当于每次答案加上 p_1 乘以一个模过 p_2 的数 q。要算 q 只要知道 $a_1 \bmod p_2$ 和 $p_1 \bmod p_2$ 就行了。

Chinese leftovers II

PE 552

换一个中国剩余定理的姿势：

$$a_1 + (a_2 - a_1) * p_1 * inv(p_1, p_2) \pmod{p_1 p_2}$$

如果把 $(a_2 - a_1) * inv(p_1, p_2)$ 模去 p_2，上面这个式子甚至不用取模。

那么相当于每次答案加上 p_1 乘以一个模过 p_2 的数 q。要算 q 只要知道 $a_1 \bmod p_2$ 和 $p_1 \bmod p_2$ 就行了。$O(n^2)$ 暴力算，然后把模所有质数的值都维护下就行了。

Chinese leftovers II

PE 552

换一个中国剩余定理的姿势：

$$a_1 + (a_2 - a_1) * p_1 * inv(p_1, p_2) \pmod{p_1 p_2}$$

如果把 $(a_2 - a_1) * inv(p_1, p_2)$ 模去 p_2，上面这个式子甚至不用取模。

那么相当于每次答案加上 p_1 乘以一个模过 p_2 的数 q。要算 q 只要知道 $a_1 \bmod p_2$ 和 $p_1 \bmod p_2$ 就行了。$O(n^2)$ 暴力算，然后把模所有质数的值都维护下就行了。

复杂度 $O(\pi(m)^2)$，$\pi(m)$ 表示 m 以内质数个数。本地跑了 10s 左右。

Lucky Days

Codechef LUCKYDAY

题意

定义序列 S:

$$S_0 = a, S_1 = b$$

$$S_i = (xS_{i-1} + yS_{i-2} + z) \pmod{p}$$

有 Q 组询问 (l, r, c)，求对于 $i \in [l, r]$，有多少 $S_i \equiv c \pmod{p}$。

范围

$q \leq 20000, p \leq 10007, l, r \leq 10^{18}$，$p$ 是质数。

Lucky Days

Codechef LUCKYDAY

Lucky Days

Codechef LUCKYDAY

这是一个 3 阶线性递推，写成矩阵形式：A^nV，其中 A 是 $3*3$ 转移矩阵，V 是 $3*1$ 列向量。

Lucky Days

Codechef LUCKYDAY

这是一个 3 阶线性递推，写成矩阵形式：A^nV，其中 A 是 $3*3$ 转移矩阵，V 是 $3*1$ 列向量。

观察到 A 的行列式等于 $-y$。那么特判掉 $y=0$ 的情况，$y\neq 0$ 时行列式就非 0，转移矩阵可逆。

Lucky Days

Codechef LUCKYDAY

这是一个 3 阶线性递推，写成矩阵形式：A^nV，其中 A 是 $3*3$ 转移矩阵，V 是 $3*1$ 列向量。

观察到 A 的行列式等于 $-y$。那么特判掉 $y=0$ 的情况，$y\neq 0$ 时行列式就非 0，转移矩阵可逆。于是结构必然是一个环，而不是一个 ρ 形。且环长是确定的（$A^nV=V$ 的最小 n）。

Lucky Days

Codechef LUCKYDAY

这是一个 3 阶线性递推，写成矩阵形式：A^nV，其中 A 是 $3*3$ 转移矩阵，V 是 $3*1$ 列向量。

观察到 A 的行列式等于 $-y$。那么特判掉 $y=0$ 的情况，$y\neq 0$ 时行列式就非 0，转移矩阵可逆。于是结构必然是一个环，而不是一个 ρ 形。且环长是确定的（$A^nV=V$ 的最小 n）。

模数 p 非常小，可以暴力枚举最终列向量 F 中除了 1、c 以外的那个值。只要求第一次出现这个向量是什么时候。之后用循环节算下即可。

Lucky Days

Codechef LUCKYDAY

这是一个 3 阶线性递推，写成矩阵形式：A^nV，其中 A 是 $3*3$ 转移矩阵，V 是 $3*1$ 列向量。

观察到 A 的行列式等于 $-y$。那么特判掉 $y=0$ 的情况，$y\neq 0$ 时行列式就非 0，转移矩阵可逆。于是结构必然是一个环，而不是一个 ρ 形。且环长是确定的（$A^nV=V$ 的最小 n）。

模数 p 非常小，可以暴力枚举最终列向量 F 中除了 1、c 以外的那个值。只要求第一次出现这个向量是什么时候。之后用循环节算下即可。相当于要求 $A^nV\equiv F$ 的最小 n，BSGS 即可。

Lucky Days

Codechef LUCKYDAY

这是一个 3 阶线性递推，写成矩阵形式：A^nV，其中 A 是 $3*3$ 转移矩阵，V 是 $3*1$ 列向量。

观察到 A 的行列式等于 $-y$。那么特判掉 $y=0$ 的情况，$y\neq 0$ 时行列式就非 0，转移矩阵可逆。于是结构必然是一个环，而不是一个 ρ 形。且环长是确定的（$A^nV=V$ 的最小 n）。

模数 p 非常小，可以暴力枚举最终列向量 F 中除了 1、c 以外的那个值。只要求第一次出现这个向量是什么时候。之后用循环节算下即可。相当于要求 $A^nV\equiv F$ 的最小 n，BSGS 即可。

n 的上界是 p^2，BSGS 要预处理一次，做 p 次，复杂度是 $O(S+p*\frac{p^2}{S})$，$S=p^{\frac{3}{2}}$ 时取到最优复杂度 $O(p^{\frac{3}{2}})$。

Fibonacci Number

Codechef FN

题意

求最小的 n 使得 $Fib_n \equiv C \pmod{P}$。

范围

$11 \leq P \leq 2 * 10^9$，P 是质数，$P \bmod 10$ 是完全平方数（1 或 9）。

Fibonacci Number

Codechef FN

Fibonacci Number

Codechef FN

$P \bmod 10$ 是 1 或 9 有啥用?

Fibonacci Number

Codechef FN

$P \bmod 10$ 是 1 或 9 有啥用?

$\left(\frac{P}{5}\right) \equiv P^{\frac{5-1}{2}} \equiv 1 \pmod 5$。

Fibonacci Number

Codechef FN

$P \bmod 10$ 是 1 或 9 有啥用?

$\left(\frac{P}{5}\right) \equiv P^{\frac{5-1}{2}} \equiv 1 \pmod 5$。

二次互反律，$\left(\frac{P}{5}\right)\left(\frac{5}{P}\right) \equiv (-1)^{\frac{(5-1)(P-1)}{4}} \equiv 1$。

Fibonacci Number

Codechef FN

$P \bmod 10$ 是 1 或 9 有啥用?

$\left(\frac{P}{5}\right) \equiv P^{\frac{5-1}{2}} \equiv 1 \pmod 5$。

二次互反律，$\left(\frac{P}{5}\right)\left(\frac{5}{P}\right) \equiv (-1)^{\frac{(5-1)(P-1)}{4}} \equiv 1$。

于是 5 在模 P 域下也是二次剩余。

Fibonacci Number

Codechef FN

Fibonacci Number

Codechef FN

考虑 Fibonacci 序列的通项：$\frac{\alpha^n-\beta^n}{\sqrt{5}}$，其中 $\alpha=\frac{1+\sqrt{5}}{2},\beta=\frac{1-\sqrt{5}}{2},\alpha+\beta=1,\alpha\beta=-1$。

Fibonacci Number

Codechef FN

考虑 Fibonacci 序列的通项：$\frac{\alpha^n-\beta^n}{\sqrt{5}}$，其中 $\alpha=\frac{1+\sqrt{5}}{2},\beta=\frac{1-\sqrt{5}}{2},\alpha+\beta=1,\alpha\beta=-1$。

如果 n 是奇数，那么有 $\frac{\alpha^n+\alpha^{-n}}{\sqrt{5}}\equiv C$。

Fibonacci Number

Codechef FN

考虑 Fibonacci 序列的通项：$\frac{\alpha^n-\beta^n}{\sqrt{5}}$，其中 $\alpha=\frac{1+\sqrt{5}}{2},\beta=\frac{1-\sqrt{5}}{2},\alpha+\beta=1,\alpha\beta=-1$。

如果 n 是奇数，那么有 $\frac{\alpha^n+\alpha^{-n}}{\sqrt{5}}\equiv C$。解这个关于 α^n 的二次方程得 $\alpha^n\equiv\frac{\sqrt{5}C\pm\sqrt{5C^2+4}}{2}$。

Fibonacci Number

Codechef FN

考虑 Fibonacci 序列的通项：$\frac{\alpha^n-\beta^n}{\sqrt{5}}$，其中 $\alpha=\frac{1+\sqrt{5}}{2},\beta=\frac{1-\sqrt{5}}{2},\alpha+\beta=1,\alpha\beta=-1$。

如果 n 是奇数，那么有 $\frac{\alpha^n+\alpha^{-n}}{\sqrt{5}}\equiv C$。解这个关于 α^n 的二次方程得 $\alpha^n\equiv\frac{\sqrt{5}C\pm\sqrt{5C^2+4}}{2}$。

n 是偶数类似。

Fibonacci Number

Codechef FN

考虑 Fibonacci 序列的通项：$\frac{\alpha^n-\beta^n}{\sqrt{5}}$，其中 $\alpha=\frac{1+\sqrt{5}}{2},\beta=\frac{1-\sqrt{5}}{2},\alpha+\beta=1,\alpha\beta=-1$。

如果 n 是奇数，那么有 $\frac{\alpha^n+\alpha^{-n}}{\sqrt{5}}\equiv C$。解这个关于 α^n 的二次方程得 $\alpha^n\equiv\frac{\sqrt{5}C\pm\sqrt{5C^2+4}}{2}$。

n 是偶数类似。

开根号就用随便什么求二次剩余的做法就行，之后 BSGS 求出最小奇数/偶数解 n。

Fibonacci Number

Codechef FN

考虑 Fibonacci 序列的通项：$\frac{\alpha^n-\beta^n}{\sqrt{5}}$，其中 $\alpha=\frac{1+\sqrt{5}}{2},\beta=\frac{1-\sqrt{5}}{2},\alpha+\beta=1,\alpha\beta=-1$。

如果 n 是奇数，那么有 $\frac{\alpha^n+\alpha^{-n}}{\sqrt{5}}\equiv C$。解这个关于 α^n 的二次方程得 $\alpha^n\equiv\frac{\sqrt{5}C\pm\sqrt{5C^2+4}}{2}$。

n 是偶数类似。

开根号就用随便什么求二次剩余的做法就行，之后 BSGS 求出最小奇数/偶数解 n。

复杂度 $O(\sqrt{P})$。

Chef attic window

Codechef WINDOW

题意

T 组数据，给出 n, A, B, K, L，求

$$\sum_{x=1}^{n} \binom{\lfloor \frac{Ax}{B} \rfloor}{K+1} \binom{x}{L}$$

范围

$T \leq 50, n, A, B \leq 10^{18}, K, L \leq 10$

Chef attic window

Codechef WINDOW

$$\binom{n}{m} = \frac{n^{\underline{m}}}{m!}$$

组合数是一个关于 n 的 m 次多项式。

Chef attic window

Codechef WINDOW

$$\binom{n}{m} = \frac{n^{\underline{m}}}{m!}$$

组合数是一个关于 n 的 m 次多项式。相当于要求

$$\sum_{x=1}^{n} \left\lfloor \frac{Ax}{B} \right\rfloor^{a} x^{b}$$

Chef attic window

Codechef WINDOW

$$\binom{n}{m} = \frac{n^{\underline{m}}}{m!}$$

组合数是一个关于 n 的 m 次多项式。相当于要求

$$\sum_{x=1}^{n} \left\lfloor \frac{Ax}{B} \right\rfloor^{a} x^{b}$$

类欧几里德算法，考虑求

$$\sum_{x=1}^{n} \left\lfloor \frac{Ax+B}{C} \right\rfloor^{a} x^{b}$$

Chef attic window

Codechef WINDOW

$$\sum_{x=1}^{n} \left\lfloor \frac{Ax+B}{C} \right\rfloor^{a} x^{b}$$

Chef attic window

Codechef WINDOW

$$\sum_{x=1}^{n} \left\lfloor \frac{Ax+B}{C} \right\rfloor^{a} x^{b}$$

$$\left\lfloor \frac{Ax+B}{C} \right\rfloor = \left\lfloor \frac{A}{C} \right\rfloor + \left\lfloor \frac{(A \bmod C)x+B}{C} \right\rfloor$$

Chef attic window

Codechef WINDOW

$$\sum_{x=1}^{n} \left\lfloor \frac{Ax+B}{C} \right\rfloor^{a} x^{b}$$

$$\left\lfloor \frac{Ax+B}{C} \right\rfloor = \left\lfloor \frac{A}{C} \right\rfloor + \left\lfloor \frac{(A \bmod C)x+B}{C} \right\rfloor$$

A 可以先模掉 C。然后注意到

$$\lfloor x \rfloor^{a} = \sum_{y=1}^{N} [y \le x] \left(y^{a} - (y-1)^{a}\right)$$

Chef attic window

Codechef WINDOW

设 $S_a(n) = \sum_{i=1}^{n} i^a$，原式

Chef attic window

Codechef WINDOW

设 $S_a(n) = \sum_{i=1}^{n} i^a$，原式

$$= \sum_{x=1}^{n} x^b \sum_{y=1}^{m} [Cy \le Ax + B](y^a - (y-1)^a)$$

Chef attic window

Codechef WINDOW

设 $S_a(n) = \sum_{i=1}^{n} i^a$，原式

$$= \sum_{x=1}^{n} x^b \sum_{y=1}^{m} [Cy \leq Ax + B](y^a - (y-1)^a)$$

$$= S_b(n) * m^a - \sum_{x=1}^{n} \sum_{y=1}^{m} [Cy - B - 1 \geq Ax] x^b (y^a - (y-1)^a)$$

Chef attic window

Codechef WINDOW

设 $S_a(n) = \sum_{i=1}^{n} i^a$，原式

$$= \sum_{x=1}^{n} x^b \sum_{y=1}^{m} [Cy \le Ax + B](y^a - (y-1)^a)$$

$$= S_b(n) * m^a - \sum_{x=1}^{n} \sum_{y=1}^{m} [Cy - B - 1 \ge Ax] x^b (y^a - (y-1)^a)$$

$$= S_b(n) * m^a - \sum_{y=1}^{m} (y^a - (y-1)^a) S_b\left(\left\lfloor \frac{Cy - B - 1}{A} \right\rfloor\right)$$

Chef attic window

Codechef WINDOW

设 $S_a(n) = \sum_{i=1}^{n} i^a$，原式

$$= \sum_{x=1}^{n} x^b \sum_{y=1}^{m} [Cy \le Ax + B](y^a - (y-1)^a)$$

$$= S_b(n) * m^a - \sum_{x=1}^{n} \sum_{y=1}^{m} [Cy - B - 1 \ge Ax] x^b (y^a - (y-1)^a)$$

$$= S_b(n) * m^a - \sum_{y=1}^{m} (y^a - (y-1)^a) S_b\left(\left\lfloor \frac{Cy - B - 1}{A} \right\rfloor\right)$$

具体实现的时候在同一层把所有 (a, b) 的答案都算出来，可以通过预处理部分和降低复杂度。复杂度 $O(K^3 \log n)$。

Contents

Contents

数论函数

数论函数：定义域为正整数集，陪域为复数域的函数。

数论函数

数论函数：定义域为正整数集，陪域为复数域的函数。
积性函数：$\forall a, b \in N^+, \gcd(a, b) = 1$ 有 $f(ab) = f(a)f(b)$。

数论函数

数论函数：定义域为正整数集，陪域为复数域的函数。

积性函数：$\forall a, b \in N^+, \gcd(a, b) = 1$ 有 $f(ab) = f(a)f(b)$。如欧拉函数 $\phi(n)$，莫比乌斯函数 $\mu(n)$，除数函数 $\sigma_k(n)$（所有正因子的 k 次方之和）等。

数论函数

数论函数：定义域为正整数集，陪域为复数域的函数。

积性函数：$\forall a, b \in N^+, \gcd(a, b) = 1$ 有 $f(ab) = f(a)f(b)$。如欧拉函数 $\phi(n)$，莫比乌斯函数 $\mu(n)$，除数函数 $\sigma_k(n)$（所有正因子的 k 次方之和）等。

完全积性函数：$\forall a, b \in N^+$ 有 $f(ab) = f(a)f(b)$。

数论函数

数论函数：定义域为正整数集，陪域为复数域的函数。

积性函数：$\forall a, b \in N^+, \gcd(a, b) = 1$ 有 $f(ab) = f(a)f(b)$。如欧拉函数 $\phi(n)$，莫比乌斯函数 $\mu(n)$，除数函数 $\sigma_k(n)$（所有正因子的 k 次方之和）等。

完全积性函数：$\forall a, b \in N^+$ 有 $f(ab) = f(a)f(b)$。如常数函数 1，幂函数 $Id_k(n) = n^k$，单位函数 $\epsilon(n) = [n = 1]$ 等。

积性函数的性质

积性函数的性质

性质 1：设 $n = \prod p_i^{q_i}$，那么 $f(n) = \prod f(p_i^{q_i})$。

积性函数的性质

性质 1：设 $n = \prod p_i^{q_i}$，那么 $f(n) = \prod f(p_i^{q_i})$。

于是可以用线性筛求出 $g(n) = p_1^{q_1}$，然后 $f(n) = f(g(n))f(\frac{n}{g(n)})$，在 $O(n)$ 时间内预处理积性函数的值。

积性函数的性质

性质 1：设 $n = \prod p_i^{q_i}$，那么 $f(n) = \prod f(p_i^{q_i})$。

于是可以用线性筛求出 $g(n) = p_1^{q_1}$，然后 $f(n) = f(g(n))f(\frac{n}{g(n)})$，在 $O(n)$ 时间内预处理积性函数的值。

性质 2：若函数 $f(n), g(n)$ 都是积性函数，那么下列函数都是积性函数：$(fg)(n)$，$(f/g)(n)$。

狄利克雷卷积

狄利克雷卷积

两个数论函数的狄利克雷卷积

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

满足:

狄利克雷卷积

两个数论函数的狄利克雷卷积

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

满足:

交换律: $f * g = g * f$

狄利克雷卷积

两个数论函数的狄利克雷卷积

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

满足:

交换律: $f*g=g*f$

结合律: $(f*g)*h=f*(g*h)$

狄利克雷卷积

两个数论函数的狄利克雷卷积

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

满足:

交换律: $f*g=g*f$

结合律: $(f*g)*h=f*(g*h)$

分配律: $f*(g+h)=f*g+f*h$

狄利克雷卷积

两个数论函数的狄利克雷卷积

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

满足:

交换律: $f*g=g*f$

结合律: $(f*g)*h=f*(g*h)$

分配律: $f*(g+h)=f*g+f*h$

单位元: $f*\epsilon=f$

狄利克雷卷积

两个数论函数的狄利克雷卷积

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

满足：

交换律：$f*g=g*f$

结合律：$(f*g)*h=f*(g*h)$

分配律：$f*(g+h)=f*g+f*h$

单位元：$f*\epsilon=f$

若 f,g 是积性函数则 $f*g$ 也是积性函数。

莫比乌斯反演

莫比乌斯反演

$$\mu * 1 = \epsilon_{\circ}$$

莫比乌斯反演

$\mu * 1 = \epsilon$。

设 $n = \prod_{i=1}^{k} p_i^{q_i}$，$n' = \prod_{i=1}^{k} p_i$，那么

莫比乌斯反演

$\mu * 1 = \epsilon$。

设 $n = \prod_{i=1}^{k} p_i^{q_i}$，$n' = \prod_{i=1}^{k} p_i$，那么

$$\sum_{d|n} \mu(d) = \sum_{d|n'} \mu(d)$$

莫比乌斯反演

$\mu * 1 = \epsilon$。
设 $n = \prod_{i=1}^{k} p_i^{q_i}$，$n' = \prod_{i=1}^{k} p_i$，那么

$$\sum_{d|n} \mu(d) = \sum_{d|n'} \mu(d)$$

$$= \sum_{i=0}^{k} \binom{k}{i} (-1)^i = (1-1)^k$$

莫比乌斯反演

莫比乌斯反演

若 $g = f * 1$，则 $f = g * \mu$。

莫比乌斯反演

若 $g = f * 1$，则 $f = g * \mu$。
两边同时卷 μ 即得。

莫比乌斯反演

若 $g = f * 1$，则 $f = g * \mu$。
两边同时卷 μ 即得。
应用：

莫比乌斯反演

若 $g = f * 1$，则 $f = g * \mu$。
两边同时卷 μ 即得。
应用：
$\sigma_k = Id_k * 1$，$Id_k = \sigma_k * \mu$。

莫比乌斯反演

若 $g = f * 1$，则 $f = g * \mu$。
两边同时卷 μ 即得。
应用：
$\sigma_k = Id_k * 1$，$Id_k = \sigma_k * \mu$。
$Id_1 = \phi * 1$，$\phi = Id_1 * \mu$。

Contents

底和顶

底和顶

底和顶的一些性质：

底和顶

底和顶的一些性质：

$x \geq n \Leftrightarrow \lfloor x \rfloor \geq n$

底和顶

底和顶的一些性质：

$x \geq n \Leftrightarrow \lfloor x \rfloor \geq n$

$x > n \Leftrightarrow \lceil x \rceil > n$

底和顶

底和顶的一些性质：

$x \geq n \Leftrightarrow \lfloor x \rfloor \geq n$

$x > n \Leftrightarrow \lceil x \rceil > n$

$x \leq n \Leftrightarrow \lceil x \rceil \leq n$

底和顶

底和顶的一些性质：

$x \geq n \Leftrightarrow \lfloor x \rfloor \geq n$

$x > n \Leftrightarrow \lceil x \rceil > n$

$x \leq n \Leftrightarrow \lceil x \rceil \leq n$

$x < n \Leftrightarrow \lfloor x \rfloor < n$

底和顶

底和顶的一些性质：

$x \geq n \Leftrightarrow \lfloor x \rfloor \geq n$

$x > n \Leftrightarrow \lceil x \rceil > n$

$x \leq n \Leftrightarrow \lceil x \rceil \leq n$

$x < n \Leftrightarrow \lfloor x \rfloor < n$

对于 $i \in [1, n]$，不同的 $\lfloor \frac{n}{i} \rfloor, \lceil \frac{n}{i} \rceil$ 都只有 $O(\sqrt{n})$ 种。

杜教筛

杜教筛

我们要算 f 的前缀和，如果有一个函数 g 使得 $f * g$ 和 g 的前缀和都能快速算出，就可以使用杜教筛。

杜教筛

我们要算 f 的前缀和，如果有一个函数 g 使得 $f*g$ 和 g 的前缀和都能快速算出，就可以使用杜教筛。

$$\sum_{i=1}^{n} f_i = \sum_{i=1}^{n} (f*g)_i - \sum_{i=2}^{n} g_i \sum_{j=1}^{\lfloor \frac{n}{i} \rfloor} f_j$$

杜教筛

我们要算 f 的前缀和，如果有一个函数 g 使得 $f*g$ 和 g 的前缀和都能快速算出，就可以使用杜教筛。

$$\sum_{i=1}^{n} f_i = \sum_{i=1}^{n} (f*g)_i - \sum_{i=2}^{n} g_i \sum_{j=1}^{\lfloor \frac{n}{i} \rfloor} f_j$$

预处理 $n^{\frac{2}{3}}$，复杂度 $n^{\frac{2}{3}} + \sum_{i=1}^{n^{\frac{1}{3}}} \sqrt{\frac{n}{i}} = O(n^{\frac{2}{3}})$。

杜教筛

杜教筛

难点在于看出狄利克雷卷积的形式并构造函数。常见的构造都是通过莫比乌斯反演构造的。

杜教筛

难点在于看出狄利克雷卷积的形式并构造函数。常见的构造都是通过莫比乌斯反演构造的。

如果有 $f = h * 1$ 我们就构造 $g = \mu$。

杜教筛

难点在于看出狄利克雷卷积的形式并构造函数。常见的构造都是通过莫比乌斯反演构造的。

如果有 $f = h * 1$ 我们就构造 $g = \mu$。

如果有 $f = h * \mu$ 我们就构造 $g = 1$。

杜教筛

难点在于看出狄利克雷卷积的形式并构造函数。常见的构造都是通过莫比乌斯反演构造的。

如果有 $f = h * 1$ 我们就构造 $g = \mu$。

如果有 $f = h * \mu$ 我们就构造 $g = 1$。

杜教筛分层复杂度不会上升。

杜教筛

难点在于看出狄利克雷卷积的形式并构造函数。常见的构造都是通过莫比乌斯反演构造的。

如果有 $f = h * 1$ 我们就构造 $g = \mu$。

如果有 $f = h * \mu$ 我们就构造 $g = 1$。

杜教筛分层复杂度不会上升。

一些特殊的构造:

杜教筛

难点在于看出狄利克雷卷积的形式并构造函数。常见的构造都是通过莫比乌斯反演构造的。

如果有 $f = h * 1$ 我们就构造 $g = \mu$。

如果有 $f = h * \mu$ 我们就构造 $g = 1$。

杜教筛分层复杂度不会上升。

一些特殊的构造:

因为有 $(\mu \cdot Id_k) * Id_k = \epsilon$，所以如果有 $f = h * (\mu \cdot Id_k)$ 就构造 $g = Id_k$，$f = h * Id_k$ 就构造 $g = (\mu \cdot Id_k)$。

杜教筛

难点在于看出狄利克雷卷积的形式并构造函数。常见的构造都是通过莫比乌斯反演构造的。

如果有 $f = h * 1$ 我们就构造 $g = \mu$。

如果有 $f = h * \mu$ 我们就构造 $g = 1$。

杜教筛分层复杂度不会上升。

一些特殊的构造:

因为有 $(\mu \cdot Id_k) * Id_k = \epsilon$，所以如果有 $f = h * (\mu \cdot Id_k)$ 就构造 $g = Id_k$，$f = h * Id_k$ 就构造 $g = (\mu \cdot Id_k)$。

$\sigma_0^2 = (\mu^2 * 1 * 1 * 1)$

杜教筛

难点在于看出狄利克雷卷积的形式并构造函数。常见的构造都是通过莫比乌斯反演构造的。

如果有 $f = h * 1$ 我们就构造 $g = \mu$。

如果有 $f = h * \mu$ 我们就构造 $g = 1$。

杜教筛分层复杂度不会上升。

一些特殊的构造:

因为有 $(\mu \cdot Id_k) * Id_k = \epsilon$，所以如果有 $f = h * (\mu \cdot Id_k)$ 就构造 $g = Id_k$，$f = h * Id_k$ 就构造 $g = (\mu \cdot Id_k)$。

$\sigma_0^2 = (\mu^2 * 1 * 1 * 1)$

……

Min25 筛

Min25 筛

对于一般的积性函数可以考虑 Min25 筛。

Min25 筛

对于一般的积性函数可以考虑 Min25 筛。
复杂度在 $n \leq 10^{12}$ 时可以近似估成 $O(\frac{n^{0.75}}{\log n})$。

Min25 筛

对于一般的积性函数可以考虑 Min25 筛。
复杂度在 $n \leq 10^{12}$ 时可以近似估成 $O(\frac{n^{0.75}}{\log n})$。
下面介绍传统的 Min25 筛，分为两部分。

Min25 筛-Part1

Min25 筛-Part1

需要对于所有 $\lfloor \frac{n}{i} \rfloor$，求 $\leq$ 它的所有质数的答案和。

Min25 筛-Part1

需要对于所有 $\lfloor \frac{n}{i} \rfloor$，求 $\leq$ 它的所有质数的答案和。

设 $F(i, n)$ 表示 $x = 1 \sim n$，x 是质数或与前 i 个质数互质（最小质因子 $> p_i$）的数的答案。

Min25 筛-Part1

需要对于所有 $\lfloor \frac{n}{i} \rfloor$，求 $\leq$ 它的所有质数的答案和。

设 $F(i, n)$ 表示 $x = 1 \sim n$，x 是质数或与前 i 个质数互质（最小质因子 $> p_i$）的数的答案。当 $p_i^2 > n$ 时，留下的就都是 $1 \sim n$ 所有质数的答案和了。

Min25 筛-Part1

需要对于所有 $\lfloor \frac{n}{i} \rfloor$，求 $\leq$ 它的所有质数的答案和。

设 $F(i,n)$ 表示 $x=1\sim n$，x 是质数或与前 i 个质数互质（最小质因子 $>p_i$）的数的答案。当 $p_i^2>n$ 时，留下的就都是 $1\sim n$ 所有质数的答案和了。

转移：减掉最小质因子恰是 p_i 的：

$F(i,n)=F(i-1,n)-T(p_i)(F(i-1,\lfloor \frac{n}{p_i} \rfloor)-F(i-1,p_{i-1}))$。

Min25 筛-Part1

需要对于所有 $\lfloor \frac{n}{i} \rfloor$，求 $\leq$ 它的所有质数的答案和。

设 $F(i,n)$ 表示 $x=1\sim n$，x 是质数或与前 i 个质数互质（最小质因子 $>p_i$）的数的答案。当 $p_i^2>n$ 时，留下的就都是 $1\sim n$ 所有质数的答案和了。

转移：减掉最小质因子恰是 p_i 的：

$F(i,n)=F(i-1,n)-T(p_i)(F(i-1,\lfloor \frac{n}{p_i} \rfloor)-F(i-1,p_{i-1}))$。

可以发现这部分转移只要转移 $n\geq p_i^2$ 的。

Min25 筛-Part2

Min25 筛-Part2

设 $S(i,n)$ 表示与前 $i-1$ 个质数互质的 $2\sim n$ 的数（最小质因子 $\geq p_i$）的答案。答案就是 $S(1,n)+1$。

Min25 筛-Part2

设 $S(i,n)$ 表示与前 $i-1$ 个质数互质的 $2\sim n$ 的数（最小质因子 $\geq p_i$）的答案。答案就是 $S(1,n)+1$。

显然 $p_i > n$ 时答案是 0。

Min25 筛-Part2

设 $S(i,n)$ 表示与前 $i-1$ 个质数互质的 $2\sim n$ 的数（最小质因子 $\geq p_i$）的答案。答案就是 $S(1,n)+1$。

显然 $p_i > n$ 时答案是 0。

首先把质数部分算进答案：$F(n)-F(p_{i-1})$。

Min25 筛-Part2

设 $S(i,n)$ 表示与前 $i-1$ 个质数互质的 $2\sim n$ 的数（最小质因子 $\geq p_i$）的答案。答案就是 $S(1,n)+1$。

显然 $p_i > n$ 时答案是 0。

首先把质数部分算进答案：$F(n)-F(p_{i-1})$。

然后考虑合数部分，枚举最小质因子以及最小质因子的幂次 p_j^e，加上 $T(p_j^e)*S(j+1,\lfloor\frac{n}{p_j^e}\rfloor)+T(p_j^{e+1})$。

Min25 筛-Part2

设 $S(i,n)$ 表示与前 $i-1$ 个质数互质的 $2\sim n$ 的数（最小质因子 $\geq p_i$）的答案。答案就是 $S(1,n)+1$。

显然 $p_i > n$ 时答案是 0。

首先把质数部分算进答案：$F(n)-F(p_{i-1})$。

然后考虑合数部分，枚举最小质因子以及最小质因子的幂次 p_j^e，加上 $T(p_j^e)*S(j+1,\lfloor\frac{n}{p_j^e}\rfloor)+T(p_j^{e+1})$。两部分都只要算 $p_j^{e+1}\leq n$ 的答案。由于 $e\geq 1$，所以仍然只枚举了 $n\geq p_j^2$ 的部分。

Contents

一种简单的问题形式

一种简单的问题形式

求

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

一种简单的问题形式

求

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

$$=\sum_{d}\sum_{i=1}^{\lfloor \frac{n}{d} \rfloor}\sum_{j=1}^{\lfloor \frac{m}{d} \rfloor} f(d)\epsilon(\gcd(i,j))$$

一种简单的问题形式

求

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

$$=\sum_{d}\sum_{i=1}^{\lfloor \frac{n}{d} \rfloor}\sum_{j=1}^{\lfloor \frac{m}{d} \rfloor} f(d)\epsilon(\gcd(i,j))$$

$$=\sum_{d} f(d)\sum_{d'}\mu(d')\lfloor \frac{n}{dd'} \rfloor \lfloor \frac{m}{dd'} \rfloor$$

一种简单的问题形式

求

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

$$=\sum_{d}^{\lfloor\frac{n}{d}\rfloor}\sum_{i=1}^{\lfloor\frac{m}{d}\rfloor}\sum_{j=1} f(d)\epsilon(\gcd(i,j))$$

$$=\sum_{d} f(d)\sum_{d'}\mu(d')\lfloor\frac{n}{dd'}\rfloor\lfloor\frac{m}{dd'}\rfloor$$

分段计算即可。

DZY Loves Math IV

BZOJ 3512

题意

求

$$\sum_{i=1}^{n}\sum_{j=1}^{m}\phi(ij)$$

答案模 10^9+7。

范围

$n \le 10^5, m \le 10^9$。

DZY Loves Math IV

BZOJ 3512

DZY Loves Math IV

BZOJ 3512

设 $S(n,m)=\sum_{k=1}^{m}\phi(nk)$。

DZY Loves Math IV

BZOJ 3512

设 $S(n,m)=\sum_{k=1}^{m}\phi(nk)$。

若 $\mu(n)\neq 0$，可求一个最大的 $d|n$ 使得 $|\mu(d)|=1$，然后 $S(n,m)=\frac{n}{d}S(d,m)$。

DZY Loves Math IV

BZOJ 3512

设 $S(n,m)=\sum_{k=1}^{m}\phi(nk)$。

若 $\mu(n)\neq 0$，可求一个最大的 $d|n$ 使得 $|\mu(d)|=1$，然后 $S(n,m)=\frac{n}{d}S(d,m)$。

若 $|\mu(n)|=1$，设 $\gcd(n,k)=d$，有

DZY Loves Math IV

BZOJ 3512

设 $S(n,m)=\sum_{k=1}^{m}\phi(nk)$。

若 $\mu(n)\neq 0$，可求一个最大的 $d|n$ 使得 $|\mu(d)|=1$，然后 $S(n,m)=\frac{n}{d}S(d,m)$。

若 $|\mu(n)|=1$，设 $\gcd(n,k)=d$，有

$$\phi(nk)=\phi(k)\phi(\frac{n}{d})d$$

DZY Loves Math IV

BZOJ 3512

设 $S(n,m)=\sum_{k=1}^{m}\phi(nk)$。

若 $\mu(n)\neq 0$，可求一个最大的 $d|n$ 使得 $|\mu(d)|=1$，然后 $S(n,m)=\frac{n}{d}S(d,m)$。

若 $|\mu(n)|=1$，设 $\gcd(n,k)=d$，有

$$\phi(nk)=\phi(k)\phi(\frac{n}{d})d$$

$$=\phi(k)\phi(\frac{n}{d})\sum_{i|d}\phi(\frac{d}{i})$$

DZY Loves Math IV

BZOJ 3512

设 $S(n,m)=\sum_{k=1}^{m}\phi(nk)$。

若 $\mu(n)\neq 0$，可求一个最大的 $d|n$ 使得 $|\mu(d)|=1$，然后 $S(n,m)=\frac{n}{d}S(d,m)$。

若 $|\mu(n)|=1$，设 $\gcd(n,k)=d$，有

$$\phi(nk)=\phi(k)\phi(\frac{n}{d})d$$

$$=\phi(k)\phi(\frac{n}{d})\sum_{i|d}\phi(\frac{d}{i})$$

$$=\phi(k)\sum_{i|d}\phi(\frac{n}{i})$$

DZY Loves Math IV

BZOJ 3512

DZY Loves Math IV

BZOJ 3512

代入原式

$$S(n,m)=\sum_{k=1}^{m}\phi(nk)$$

DZY Loves Math IV

BZOJ 3512

代入原式

$$S(n,m)=\sum_{k=1}^{m}\phi(nk)$$

$$=\sum_{k=1}^{m}\phi(k)\sum_{d|(n,k)}\phi(\frac{n}{d})$$

DZY Loves Math IV

BZOJ 3512

代入原式

$$S(n,m)=\sum_{k=1}^{m}\phi(nk)$$

$$=\sum_{k=1}^{m}\phi(k)\sum_{d|(n,k)}\phi(\frac{n}{d})$$

$$=\sum_{d|n}\phi(\frac{n}{d})\sum_{k=1}^{\lfloor\frac{m}{d}\rfloor}\phi(dk)$$

DZY Loves Math IV

BZOJ 3512

代入原式

$$S(n,m)=\sum_{k=1}^{m}\phi(nk)$$

$$=\sum_{k=1}^{m}\phi(k)\sum_{d|(n,k)}\phi(\frac{n}{d})$$

$$=\sum_{d|n}\phi(\frac{n}{d})\sum_{k=1}^{\lfloor\frac{m}{d}\rfloor}\phi(dk)$$

$$=\sum_{d|n}\phi(\frac{n}{d})S(d,\lfloor\frac{m}{d}\rfloor)$$

DZY Loves Math IV

BZOJ 3512

DZY Loves Math IV

BZOJ 3512

$S(1, m)$ 就是个 ϕ 的前缀和。

DZY Loves Math IV

BZOJ 3512

$S(1, m)$ 就是个 ϕ 的前缀和。
记忆化搜下去，复杂度近似为杜教筛复杂度。

循环之美

NOI 2016 D1T3

题意

给出 n, m, k，问有多少本质不同的分数，分子是 $1 \sim n$，分母是 $1 \sim m$，分数值在 k 进制下是纯循环小数。

范围

$n, m \leq 10^9, k \leq 2000$。

循环之美

NOI 2016 D1T3

循环之美

NOI 2016 D1T3

一个分数是纯循环小数的充要条件是分母和 k 互质。

循环之美

NOI 2016 D1T3

一个分数是纯循环小数的充要条件是分母和 k 互质。
于是就是算

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

循环之美

NOI 2016 D1T3

一个分数是纯循环小数的充要条件是分母和 k 互质。
于是就是算

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

设

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

循环之美

NOI 2016 D1T3

一个分数是纯循环小数的充要条件是分母和 k 互质。
于是就是算

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

设

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

f 可以 $O(\sqrt{k})$ 求出，现在要求的就是

$$\sum_{d}\lfloor\frac{n}{d}\rfloor f(\lfloor\frac{m}{d}\rfloor)[\gcd(d,k)=1]\mu(d)$$

循环之美

NOI 2016 D1T3

循环之美

NOI 2016 D1T3

问题变成了求

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

循环之美

NOI 2016 D1T3

问题变成了求

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

展开来是

$$\sum_{d|k}\mu(d)\sum_{i=1}^{\lfloor\frac{n}{d}\rfloor}\mu(id)$$

循环之美

NOI 2016 D1T3

循环之美

NOI 2016 D1T3

设 $S(n,k)=\sum_{i=1}^{n}\mu(ik)$。

循环之美

NOI 2016 D1T3

设 $S(n,k)=\sum_{i=1}^{n}\mu(ik)$。若 k 有平方因子，显然答案是 0，否则

$$\mu(ik)=\mu(k)\mu(i)[\gcd(i,k)=1]$$

循环之美

NOI 2016 D1T3

设 $S(n,k)=\sum_{i=1}^{n}\mu(ik)$。若 k 有平方因子，显然答案是 0，否则

$$\mu(ik)=\mu(k)\mu(i)[\gcd(i,k)=1]$$

推一推可以得到 $S(n,d)=\mu(d)\sum_{k|d}\mu(k)S(\lfloor\frac{n}{k}\rfloor,k)$。

循环之美

NOI 2016 D1T3

设 $S(n,k)=\sum_{i=1}^{n}\mu(ik)$。若 k 有平方因子，显然答案是 0，否则

$$\mu(ik)=\mu(k)\mu(i)[\gcd(i,k)=1]$$

推一推可以得到 $S(n,d)=\mu(d)\sum_{k|d}\mu(k)S(\lfloor\frac{n}{k}\rfloor,k)$。记忆化搜下去就行了。

老夫的魔法

from whzzt

题意

定义魔法阵为 $3*3$ 的没有相同元素的且每行每列和主对角线元素乘积都相等的正整数矩阵，求中心元素在 $[l,r]$ 之间的本质不同（不能通过旋转或翻转变得相同）魔法阵数目。

范围

$r \leq 2*10^{11}$。

老夫的魔法

from whzzt

老夫的魔法

from whzzt

本质不同只要最后除以 8 就行了。

老夫的魔法

from whzzt

本质不同只要最后除以 8 就行了。

对于每个质因子去考虑，每行、每列、两条主对角线上的指数的和都要相等，还要求最终的魔法阵没有相同元素。

老夫的魔法

from whzzt

本质不同只要最后除以 8 就行了。

对于每个质因子去考虑，每行、每列、两条主对角线上的指数的和都要相等，还要求最终的魔法阵没有相同元素。两个元素相同当且仅当在所有质因子处的指数都相同，由于限制很强，不同的连通性状态是很少的。

老夫的魔法

from whzzt

本质不同只要最后除以 8 就行了。

对于每个质因子去考虑，每行、每列、两条主对角线上的指数的和都要相等，还要求最终的魔法阵没有相同元素。两个元素相同当且仅当在所有质因子处的指数都相同，由于限制很强，不同的连通性状态是很少的。暴搜可得只有 10 种。

老夫的魔法

from whzzt

本质不同只要最后除以 8 就行了。

对于每个质因子去考虑，每行、每列、两条主对角线上的指数的和都要相等，还要求最终的魔法阵没有相同元素。两个元素相同当且仅当在所有质因子处的指数都相同，由于限制很强，不同的连通性状态是很少的。暴搜可得只有 10 种。

我们暴搜预处理出 $f(x, S)$ 表示对于任意质数，中间元素的指数为 x，连通性状态为 S 的方案数。

老夫的魔法

from whzzt

本质不同只要最后除以 8 就行了。

对于每个质因子去考虑，每行、每列、两条主对角线上的指数的和都要相等，还要求最终的魔法阵没有相同元素。两个元素相同当且仅当在所有质因子处的指数都相同，由于限制很强，不同的连通性状态是很少的。暴搜可得只有 10 种。

我们暴搜预处理出 $f(x, S)$ 表示对于任意质数，中间元素的指数为 x，连通性状态为 S 的方案数。现在相当于对每个质数去确定它的指数，使得中间元素落在 $[l, r]$ 内且到最后 S 变为 0（没有相同元素）。

老夫的魔法

from whzzt

老夫的魔法

from whzzt

用 Min25 筛的过程把所有的质数背包起来。复杂度 $O(Min25 * 10)$。

老夫的魔法

from whzzt

用 Min25 筛的过程把所有的质数背包起来。复杂度 $O(Min25 * 10)$。

又注意到答案只和指数的（有序）序列有关，跟具体的质因子序列无关。

老夫的魔法

from whzzt

用 Min25 筛的过程把所有的质数背包起来。复杂度 $O(Min25 * 10)$。

又注意到答案只和指数的（有序）序列有关，跟具体的质因子序列无关。本质不同的指数有序序列也不多，可以全部搜出来把答案算出来存在字母树上。

老夫的魔法

from whzzt

用 Min25 筛的过程把所有的质数背包起来。复杂度 $O(Min25 * 10)$。

又注意到答案只和指数的（有序）序列有关，跟具体的质因子序列无关。本质不同的指数有序序列也不多，可以全部搜出来把答案算出来存在字母树上。Min25 筛的时候只要考虑有多少种把质因子分配给这个指数序列的方法能使数落在 $[l, r]$ 内就可以了。

老夫的魔法

from whzzt

用 Min25 筛的过程把所有的质数背包起来。复杂度 $O(Min25 * 10)$。

又注意到答案只和指数的（有序）序列有关，跟具体的质因子序列无关。本质不同的指数有序序列也不多，可以全部搜出来把答案算出来存在字母树上。Min25 筛的时候只要考虑有多少种把质因子分配给这个指数序列的方法能使数落在 $[l, r]$ 内就可以了。

复杂度 $O(Min25) \simeq O(\frac{n^{0.75}}{\log n})$。