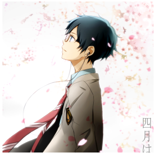
数学 preview

长郡中学 周书予

2019 年 9 月 19 日



欧几里得算法及其扩展

求 $\gcd(a, b)$?

欧几里得算法及其扩展

求 $\gcd(a, b)$？

求 $ax + by = \gcd(a, b)$ 的一组整数解？

欧几里得算法及其扩展

求 $\gcd(a, b)$？

求 $ax + by = \gcd(a, b)$ 的一组整数解？

由 $\gcd(a, b) = \gcd(b, a \bmod b)$ 可得：

$$ax_1 + by_1 = bx_2 + (a - \lfloor \frac{a}{b} \rfloor b)y_2$$

该不定方程的一组特解为：

$$\begin{cases} x_1 = y_2 \\ y_1 = x_2 - \lfloor \frac{a}{b} \rfloor y_2 \end{cases}$$

递归到 $b = 0$ 时存在一组特解 $\begin{cases} x = 1 \\ y = 0 \end{cases}$ 。

费马小定理

若 p 为质数，则 $a^{p-1} \equiv 1 \mod p$。

常使用 $a^{p-2} \equiv a^{-1} \mod p$ 来求逆元。

欧拉定理及其扩展

若 $\gcd(a,n)=1$，则 $a^{\varphi(n)}\equiv 1 \mod n$。

可以发现费马小定理是欧拉定理在 n 为质数时的特殊情况。

若 $\gcd(a,n)\neq 1$，则

$$a^b \equiv \begin{cases} a^b & b<\varphi(n) \\ a^{b \bmod \varphi(n)+\varphi(n)} & b\geq\varphi(n) \end{cases} \mod n$$

线性求逆元

在 $O(n)$ 时间内求出 $1 \ldots n$ 在模 p 意义下的逆元。

$$
\begin{aligned}
i \times \lfloor \frac{p}{i} \rfloor + (p \bmod i) &\equiv 0 && \mod p \\
i \times \lfloor \frac{p}{i} \rfloor &\equiv -(p \bmod i) && \mod p \\
i^{-1} &\equiv -\lfloor \frac{p}{i} \rfloor \times (p \bmod i)^{-1} && \mod p
\end{aligned}
$$

这种求法同样适用于 p 不为质数的情况。

当然也可以使用 $O(n + \log p)$ 的方法，但要注意去除与 p 不互质（不存在逆元）的数。

中国剩余定理

假设 $p_1, p_2, ..., p_k$ 两两互质，并记 $P = \prod_{i=1}^{k} p_i$，则同余方程组

$$\begin{cases} x \equiv a_1 \mod p_1 \\ x \equiv a_2 \mod p_2 \\ \dots \\ x \equiv a_k \mod p_k \end{cases}$$

的最小整数解为 $\sum_{i=1}^{k} e_i w_i a_i \mod P$，其中 $w_i = \frac{P}{p_i}, e_i w_i \equiv 1 \mod p_i$。

唯一分解定理

任意正整数都可以被唯一分解成若干质数的乘积。

$$n = \prod_{i=1}^{k} p_i^{\alpha_i}$$

接下来会默认存在上述定义。

干货

本页给出一张 n 以内最多不同质因子个数与最多约数个数的表格，在计算复杂度时，大可不必用 $O(\log n)$ 和 $O(\sqrt{n})$ 去估计此二者。

$n \le$	10^1	10^2	10^3	10^4	10^5	10^6	10^7	10^8	10^9
$\max\{\omega(n)\}$	2	3	4	5	6	7	8	8	9
$\max\{d(n)\}$	4	12	32	64	128	240	448	768	1344
$n \le$	10^{10}	10^{11}	10^{12}	10^{13}	10^{14}	10^{15}	10^{16}	10^{17}	10^{18}
$\max\{\omega(n)\}$	10	10	11	12	12	13	13	14	15
$\max\{d(n)\}$	2304	4032	6720	10752	17280	26880	41472	64512	103680

质数分解算法

$O(\sqrt{n})$ 分解

由于 n 只包含至多 1 个大于 $\sqrt{n}$ 的质因子，所以可以枚举所有小于等于 $\sqrt{n}$ 的质因子试除，剩下的数若大于 1 则说明也是 n 的一个质因子。

$O(n) - O(\log n)$ 分解

考虑到质因子个数是 $O(\log n)$ 级别的，因此先 $O(n)$ 预处理 $1 \dots n$ 所有数的最小质因子 d_i，每次分解时通过不断地 $x \to \frac{x}{d_x}$ 即可实现单次 $O(\log n)$ 分解。

Pollard-Rho 质因数分解

复杂度 $O(n^{\frac{1}{4}})$，有兴趣的同学可以去自行了解。~~说白了就是懒得讲~~

积性函数

若 $f(n)$ 的定义域为正整数集，值域为复数集，则称 $f(n)$ 为数论函数。

若 $f(n)$ 为数论函数，且 $f(1)=1$，对于任意互质的正整数 p, q 均满足 $f(p\times q)=f(p)\times f(q)$，则称 $f(n)$ 为积性函数。

若 $f(n)$ 为积性函数，且对于任意正整数 p, q 均满足 $f(p\times q)=f(p)\times f(q)$，则称 $f(n)$ 为完全积性函数。

常见积性函数

莫比乌斯函数 $\mu(n)=\prod_{i=1}^{k}-[\alpha_i=1]$。

欧拉函数 $\varphi(n)=\sum_{i=1}^{n}[\gcd(i,n)=1]=\prod_{i=1}^{k}(p_i-1)p_i^{\alpha_i-1}$。

除数函数 $\sigma_x(n)=\sum_{d|n}d^x=\prod_{i=1}^{k}\sum_{j=0}^{\alpha_i}p_i^{xj}$。

单位元函数 $e(n)=[n=1]$。

恒等函数 $I(n)=1$。

幂函数 $id_x(n)=n^x$。

Dirichlet 卷积

数论函数 $f(n)$ 与 $g(n)$ 的 Dirichlet 卷积为 $(f*g)(n)=\sum_{d|n}f(d)g(\frac{n}{d})$，当 f 和 g 均为积性函数时，$f*g$ 也是积性函数。

Dirichlet 卷积满足交换律、结合律，对加法满足分配率，存在单位元函数 $e(n)=[n=1]$ 使得 $f*e=e*f=f$，且任意 $f(1)\neq 0$ 的数论函数 $f(n)$ 存在唯一的逆元 $f^{-1}(n)$ 使得 $f*f^{-1}=e$。

常见 Dirichlet 卷积

常见 Dirichlet 卷积

大家都知道

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

常见 Dirichlet 卷积

大家都知道

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

写成 Dirichlet 卷积的形式就是

$$\mu*I=e,\varphi*I=id_1$$

常见 Dirichlet 卷积

大家都知道

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

写成 Dirichlet 卷积的形式就是

$$\mu*I=e,\varphi*I=id_1$$

那么就很显然可以看出

$$\mu*id_1=\varphi$$

常见 Dirichlet 卷积

大家都知道

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

写成 Dirichlet 卷积的形式就是

$$\mu*I=e,\varphi*I=id_1$$

那么就很显然可以看出

$$\mu*id_1=\varphi$$

也即

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

筛法

埃氏筛法

枚举每个质数并筛去其倍数，时间复杂度 $O(n\log\log n)$。

欧拉筛法（线性筛）

枚举每个数 n，从小到大枚举质数 $p \le p_1$ 并把 $n \times p$ 筛掉，这样可以保证所有数只会被其最小质因子筛掉一次。

原根

假设 g 是质数 p 的一个原根，则 $g^0, g^1, ..., g^{p-2}$ 在模 p 意义下两两不同。

也即，对于 $\forall x \in [1, p-1]$，均存在 $k \in [0, p-2]$ 使 $x \equiv g^k \mod p$。

由这种方式可以把模意义下的乘法转化成原根指数上的加法，也就是实现了模意义下的离散对数。

大步小步算法

给出 a, b, p，求 $a^x \equiv b \mod p$。

大步小步算法

给出 a, b, p，求 $a^x \equiv b \mod p$。

分块，令 $k = \lceil \sqrt{p} \rceil$，设 $x = ky - z$，则有 $a^{ky} = b \times a^z$。
显然 $y, z \in [0, k]$，因此预处理出所有 a^{ky} 并哈希存储，再枚举 z 判断是否存在相等的即可。

大步小步算法

给出 a, b, p，求 $a^x \equiv b \mod p$。

分块，令 $k = \lceil \sqrt{p} \rceil$，设 $x = ky - z$，则有 $a^{ky} = b \times a^z$。
显然 $y, z \in [0, k]$，因此预处理出所有 a^{ky} 并哈希存储，再枚举 z 判断是否存在相等的即可。

这个做法要求 $\gcd(a, p) = 1$，因为推导过程中用到了 a 在模 p 意义下的逆元。

组合数

$\binom{n}{m}$ 表示从 n 个元素中选出 m 个的方案数。

通项：$\binom{n}{m} = \binom{n}{n-m} = \frac{n!}{m!(n-m)!}$

递推式：$\binom{n}{m} = \binom{n-1}{m} + \binom{n-1}{m-1}$

同行递推：$\binom{n}{m} = \binom{n}{m-1} \times \frac{n-m+1}{m}$

二项式定理：$(x+y)^n = \sum_{i=0}^{n} \binom{n}{i} x^i y^{n-i}$

格路问题

网格图上每步可以向右或向上走一步，从 $(0,0)$ 走到 (n,m) 的方案数？

格路问题

网格图上每步可以向右或向上走一步，从 $(0,0)$ 走到 (n,m) 的方案数？

$\binom{n+m}{n}$，即从共计 $n+m$ 步中选出 n 步向右走。

格路问题

网格图上每步可以向右或向上走一步，从 $(0,0)$ 走到 (n,m) 的方案数？

$\binom{n+m}{n}$，即从共计 $n+m$ 步中选出 n 步向右走。

从 $(0,0)$ 走到 (n,m)，要求不碰到直线 $y=x+b(n+b<m)$ 的方案数？

格路问题

网格图上每步可以向右或向上走一步，从 $(0,0)$ 走到 (n,m) 的方案数？

$\binom{n+m}{n}$，即从共计 $n+m$ 步中选出 n 步向右走。

从 $(0,0)$ 走到 (n,m)，要求不碰到直线 $y=x+b(n+b<m)$ 的方案数？

$\binom{n+m}{n}-\binom{n+m}{n+b}$，因为所有碰到直线 $y=x+b$ 的方案均可唯一对应到一种从 $(-b,b)$ 出发走到 (n,m) 的方案。

Lucas 定理

已知 p 为质数，则有 $\binom{n}{m} \bmod p = \binom{\lfloor n/p \rfloor}{\lfloor m/p \rfloor} \times \binom{n \bmod p}{m \bmod p} \bmod p$。

Lucas 定理

已知 p 为质数，则有 $\binom{n}{m} \bmod p = \binom{\lfloor n/p \rfloor}{\lfloor m/p \rfloor} \times \binom{n \bmod p}{m \bmod p} \bmod p$。

另一种表述是，令 $n = \overline{n_1 n_2 \ldots n_k}_{(p)}, m = \overline{m_1 m_2 \ldots m_k}_{(p)}$（即 n, m 的 p 进制表示），则有

$$\binom{n}{m} \bmod p = \prod_{i=1}^{k} \binom{n_i}{m_i} \bmod p$$

定义 $n < m$ 时 $\binom{n}{m} = 0$。

Lucas 定理

已知 p 为质数，则有 $\binom{n}{m} \bmod p = \binom{\lfloor n/p \rfloor}{\lfloor m/p \rfloor} \times \binom{n \bmod p}{m \bmod p} \bmod p$。

另一种表述是，令 $n = \overline{n_1 n_2 \dots n_k}_{(p)}, m = \overline{m_1 m_2 \dots m_k}_{(p)}$（即 n, m 的 p 进制表示），则有

$$\binom{n}{m} \bmod p = \prod_{i=1}^{k} \binom{n_i}{m_i} \bmod p$$

定义 $n < m$ 时 $\binom{n}{m} = 0$。

证明可以考虑 $\binom{n}{m}$ 的生成函数解释是 $[x^m](1+x)^n$，把此处的 n, m 按 p 进制拆分后再运用一些组合小技巧（$(1+x)^p \equiv 1 + x^p \mod p$）即可完成证明。

容斥原理

$$|\overline{\cup_{i=1}^{n} A_i}| = \sum_{k=1}^{n}(-1)^k \sum_{1\le i_1<i_2<\ldots<i_k\le n} |A_{i_1}\cap A_{i_2}\cap \ldots \cap A_{i_k}|$$

说人话就是，n 件坏事都不发生的概率，可以通过 2^n 个子集中的坏事同时发生的概率通过加加减减得到。

错排问题

有 n 个人编号为 $1, ..., n$，问这 n 个人站成一排全都站错位置的方案数。

错排问题

有 n 个人编号为 $1, ..., n$，问这 n 个人站成一排全都站错位置的方案数。

设 $f(n)$ 表示 n 个人随便站的方案数，显然 $f(n) = n!$，设 $g(n)$ 表示 n 个人都站错的方案数，枚举有多少个人站错了位置，我们有

$$f(n) = \sum_{k=0}^{n} \binom{n}{k} g(k)$$

然而我们是不知道 g 而知道 f 吧！

错排问题

有 n 个人编号为 $1, ..., n$，问这 n 个人站成一排全都站错位置的方案数。

设 $f(n)$ 表示 n 个人随便站的方案数，显然 $f(n) = n!$，设 $g(n)$ 表示 n 个人都站错的方案数，枚举有多少个人站错了位置，我们有

$$f(n) = \sum_{k=0}^{n} \binom{n}{k} g(k)$$

然而我们是不知道 g 而知道 f 吧！

这里给出一个组合恒等式，接下来我们将会使用它来证明二项式反演。

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

请开始你的表演

请开始你的表演

首先说一句废话

$$g(n)=\sum_{m=0}^{n}[n-m=0]\binom{n}{m}g(m)$$

请开始你的表演

首先说一句废话

$$g(n)=\sum_{m=0}^{n}[n-m=0]\binom{n}{m}g(m)$$

然后将我们之前的那个式子代入

$$g(n)=\sum_{m=0}^{n}\sum_{k=0}^{n-m}(-1)^k\binom{n-m}{k}\binom{n}{m}g(m)$$

请开始你的表演

首先说一句废话

$$g(n)=\sum_{m=0}^{n}[n-m=0]\binom{n}{m}g(m)$$

然后将我们之前的那个式子代入

$$g(n)=\sum_{m=0}^{n}\sum_{k=0}^{n-m}(-1)^k\binom{n-m}{k}\binom{n}{m}g(m)$$

利用组合恒等式进行变换（提示：考虑组合意义）

$$g(n)=\sum_{m=0}^{n}\sum_{k=0}^{n-m}(-1)^k\binom{n}{k}\binom{n-k}{m}g(m)$$

请开始你的表演

首先说一句废话

$$g(n)=\sum_{m=0}^{n}[n-m=0]\binom{n}{m}g(m)$$

然后将我们之前的那个式子代入

$$g(n)=\sum_{m=0}^{n}\sum_{k=0}^{n-m}(-1)^k\binom{n-m}{k}\binom{n}{m}g(m)$$

利用组合恒等式进行变换（提示：考虑组合意义）

$$g(n)=\sum_{m=0}^{n}\sum_{k=0}^{n-m}(-1)^k\binom{n}{k}\binom{n-k}{m}g(m)$$

交换求和号

$$g(n)=\sum_{k=0}^{n}(-1)^k\binom{n}{k}\sum_{m=0}^{n-k}\binom{n-k}{m}g(m)$$

证完了

$$g(n)=\sum_{k=0}^{n}(-1)^k\binom{n}{k}\sum_{m=0}^{n-k}\binom{n-k}{m}g(m)$$

证完了

$$g(n)=\sum_{k=0}^{n}(-1)^k\binom{n}{k}\sum_{m=0}^{n-k}\binom{n-k}{m}g(m)$$

后面的那坨就是 $f(n-k)$，也就是说

$$\begin{cases} f(n)=\sum_{k=0}^{n}\binom{n}{k}g(k) \\ g(n)=\sum_{k=0}^{n}(-1)^{n-k}\binom{n}{k}f(k) \end{cases}$$

此即为二项式反演。

莫比乌斯反演

莫比乌斯反演的一般形式为

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

莫比乌斯函数满足 $\sum_{d|n} \mu(d) = [n = 1]$。

莫比乌斯反演

莫比乌斯反演的一般形式为

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

莫比乌斯函数满足 $\sum_{d|n} \mu(d) = [n = 1]$。

~~证明留作课后练习。~~

莫比乌斯反演

莫比乌斯反演的一般形式为

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

莫比乌斯函数满足 $\sum_{d|n} \mu(d) = [n = 1]$。

~~证明留作课后练习。~~

（考虑 $\mu * I = e$，由 $f = g * I$ 得 $g = f * \mu$ 就直接证完了）

感性理解莫比乌斯反演

对于一个正整数 $n=\prod_{i=1}^{k} p_i^{\alpha_i}$，我们将其视作一个 k 维空间上的点 $(\alpha_1,\alpha_2,...,\alpha_k)$。

感性理解莫比乌斯反演

对于一个正整数 $n=\prod_{i=1}^{k} p_i^{\alpha_i}$，我们将其视作一个 k 维空间上的点 $(\alpha_1, \alpha_2, ..., \alpha_k)$。

之前提到的 $g(n)$ 相当于是 $(\alpha_1, \alpha_2, ..., \alpha_k)$ 这个点的点权，而 $f(n)$ 呢？

感性理解莫比乌斯反演

对于一个正整数 $n=\prod_{i=1}^{k}p_i^{\alpha_i}$，我们将其视作一个 k 维空间上的点 $(\alpha_1,\alpha_2,...,\alpha_k)$。

之前提到的 $g(n)$ 相当于是 $(\alpha_1,\alpha_2,...,\alpha_k)$ 这个点的点权，而 $f(n)$ 呢？

是 $(\alpha_1,\alpha_2,...,\alpha_k)$ 这个点的 k 维前缀和，即所有满足 $\beta_i\le\alpha_i$ 的 $(\beta_1,\beta_2,...,\beta_k)$ 点权之和。

感性理解莫比乌斯反演

对于一个正整数 $n = \prod_{i=1}^{k} p_i^{\alpha_i}$，我们将其视作一个 k 维空间上的点 $(\alpha_1, \alpha_2, ..., \alpha_k)$。

之前提到的 $g(n)$ 相当于是 $(\alpha_1, \alpha_2, ..., \alpha_k)$ 这个点的点权，而 $f(n)$ 呢？

是 $(\alpha_1, \alpha_2, ..., \alpha_k)$ 这个点的 k 维前缀和，即所有满足 $\beta_i \le \alpha_i$ 的 $(\beta_1, \beta_2, ..., \beta_k)$ 点权之和。

回顾二维前缀和、二维差分的做法，或者脑补一下三维的情况，处理 k 维前缀和时应该需要找到 2^k 个前缀和来加加减减。

感性理解莫比乌斯反演

对于一个正整数 $n=\prod_{i=1}^{k}p_i^{\alpha_i}$，我们将其视作一个 k 维空间上的点 $(\alpha_1,\alpha_2,...,\alpha_k)$。

之前提到的 $g(n)$ 相当于是 $(\alpha_1,\alpha_2,...,\alpha_k)$ 这个点的点权，而 $f(n)$ 呢？

是 $(\alpha_1,\alpha_2,...,\alpha_k)$ 这个点的 k 维前缀和，即所有满足 $\beta_i\le\alpha_i$ 的 $(\beta_1,\beta_2,...,\beta_k)$ 点权之和。

回顾二维前缀和、二维差分的做法，或者脑补一下三维的情况，处理 k 维前缀和时应该需要找到 2^k 个前缀和来加加减减。

然而 $g(n)=\sum_{d|n}\mu(d)f(\frac{n}{d})$ 哪里来的 2^k？

感性理解莫比乌斯反演

对于一个正整数 $n=\prod_{i=1}^{k} p_i^{\alpha_i}$，我们将其视作一个 k 维空间上的点 $(\alpha_1, \alpha_2, ..., \alpha_k)$。

之前提到的 $g(n)$ 相当于是 $(\alpha_1, \alpha_2, ..., \alpha_k)$ 这个点的点权，而 $f(n)$ 呢？

是 $(\alpha_1, \alpha_2, ..., \alpha_k)$ 这个点的 k 维前缀和，即所有满足 $\beta_i \leq \alpha_i$ 的 $(\beta_1, \beta_2, ..., \beta_k)$ 点权之和。

回顾二维前缀和、二维差分的做法，或者脑补一下三维的情况，处理 k 维前缀和时应该需要找到 2^k 个前缀和来加加减减。

然而 $g(n)=\sum_{d|n}\mu(d)f(\frac{n}{d})$ 哪里来的 2^k？

冷静分析，上式枚举的所有 $\mu(d)$ 中恰有 2^k 个非零，这说明了莫比乌斯反演的本质是以质因子为维度的高维前缀和/差分变换。

高维前缀和

用于解决高维空间中的部分和问题。由于高维空间目前还没什么实际应用，因此这种高维问题往往是由某些其他问题抽象而来，比如约数（把每个质因子的指数看做一维的坐标），子集（每一维大小为 2 的高维空间）等问题。

经典问题

description

给出 $\{a_0, a_1, ..., a_{2^n-1}\}$，求 $\{b_0, b_1, ..., b_{2^n-1}\}$ 满足

$$b_i = \sum_{j \subseteq i} a_j$$

constriction

$1 \leq n \leq 20.$

经典问题

description

给出 $\{a_0, a_1, ..., a_{2^n-1}\}$，求 $\{b_0, b_1, ..., b_{2^n-1}\}$ 满足

$$b_i = \sum_{j \subseteq i} a_j$$

constriction

$1 \leq n \leq 20.$

solution

矩阵

一个 $n \times m$ 的矩阵大概长这样：

$$A = \begin{bmatrix} a_{1,1} & a_{1,2} & \cdots & a_{1,m} \\ a_{2,1} & a_{2,2} & \cdots & a_{2,m} \\ \vdots & \vdots & \ddots & \vdots \\ a_{n,1} & a_{n,2} & \cdots & a_{n,m} \end{bmatrix}$$

特别地，$n \times 1$ 的矩阵常被称作列向量，$1 \times m$ 的矩阵常被称作行向量，1×1 的矩阵有时会与矩阵元素不作区分。

矩阵乘法

一个 $n \times m$ 的矩阵 A 与一个 $m \times r$ 的矩阵 B 的乘积定义为一个 $n \times r$ 的矩阵 C，其中

$$c_{i,j} = \sum_{k=1}^{m} a_{i,k} \times b_{k,j}$$

即矩阵 A 的第 i 行与矩阵 B 的第 j 列点乘的结果。

矩阵乘法满足结合律不满足交换率。

$n \times n$ 的矩阵乘法存在单位矩阵 E 满足 $e_{i,j} = [i = j]$，不是所有的矩阵都存在逆矩阵。

经典问题

description

给一张 n 个点的有向图 G，求点 1 到点 n 经过 k 条边的最短路。

constriction

$1 \leq n \leq 100, 1 \leq k \leq 10^9.$

经典问题

description

给一张 n 个点的有向图 G，求点 1 到点 n 经过 k 条边的最短路。

constriction

$1 \leq n \leq 100, 1 \leq k \leq 10^9$.

solution

经典问题

description

给一张 n 个点的有向图 G，求点 1 到点 n 长度至少为 k 的路径经过的最少边数。

constriction

$1 \le n \le 100, 1 \le k \le 10^9$.

经典问题

description

给一张 n 个点的有向图 G，求点 1 到点 n 长度至少为 k 的路径经过的最少边数。

constriction

$1 \le n \le 100, 1 \le k \le 10^9$.

solution

线性递推

矩阵乘法可以用于表示线性递推，比如最经典的斐波那契递推：

$$f_k = f_{k-1} + f_{k-2} (k \geq 2)$$

$$\begin{bmatrix} f_0 & f_1 \end{bmatrix} \begin{bmatrix} 0 & 1 \\ 1 & 1 \end{bmatrix} = \begin{bmatrix} f_1 & f_2 \end{bmatrix}$$

$$\begin{bmatrix} f_0 & f_1 \end{bmatrix} \begin{bmatrix} 0 & 1 \\ 1 & 1 \end{bmatrix}^n = \begin{bmatrix} f_n & f_{n+1} \end{bmatrix}$$

这样可以把求一个 m 阶递推数列的第 n 项这个问题做到 $O(m^3 \log n)$ 的时间复杂度。

hdu4471 Homework

description

已知数列 f 的前 m 项值和另外 q 个值 $f_{x_i}=y_i$，其余值满足递推式 $f_k=\sum_{i=1}^{m} c_i\times f_{k-i}$。求 f_n 的值。

constriction

$1\le n\le 10^9, 1\le m,q\le 100.$

hdu4471 Homework

description

已知数列 f 的前 m 项值和另外 q 个值 $f_{x_i}=y_i$，其余值满足递推式 $f_k=\sum_{i=1}^{m} c_i\times f_{k-i}$。求 f_n 的值。

constriction

$1\le n\le 10^9, 1\le m, q\le 100.$

solution

高斯消元

解线性方程组

$$\begin{cases} a_{1,1}x_1 + a_{1,2}x_2 + \ldots + a_{1,n}x_n = b_1 \\ a_{2,1}x_1 + a_{2,2}x_2 + \ldots + a_{2,n}x_n = b_2 \\ \ldots \\ a_{m,1}x_1 + a_{m,2}x_2 + \ldots + a_{m,n}x_n = b_m \end{cases}$$

解方程的过程可以概括为以下三步：

- 选取一个未被选过的未知数作为主元，选取一个未被选过的主元系数不为 0 的方程；
- 将这个方程中主元系数化为 1；
- 通过加减消元，使其他方程中主元的系数都变成 0。

该算法的时间复杂度为 $O(n^3)$。需要注意可能存在无解或多解的情况。

感性理解线性基

狭义的线性基指异或空间中的一组线性无关的基底，该空间下的任意一个向量均可以用这些基底线性表示。

感性理解线性基

狭义的线性基指异或空间中的一组线性无关的基底，该空间下的任意一个向量均可以用这些基底线性表示。

实际上将数域从 $\{0,1\}$ 扩展到 $\mathbb{C}$ 时，线性无关的那套理论仍然成立。

因而在解线性方程组时，可以把每个方程看做一个 n 维带权向量。

感性理解线性基

狭义的线性基指异或空间中的一组线性无关的基底，该空间下的任意一个向量均可以用这些基底线性表示。

实际上将数域从 $\{0,1\}$ 扩展到 $\mathbb{C}$ 时，线性无关的那套理论仍然成立。

因而在解线性方程组时，可以把每个方程看做一个 n 维带权向量。

方程组无解，当且仅当存在一组线性相关的向量，经过某种线性组合后得到零向量，而该零向量的权值非零。

方程组有多解，当且仅当方程组有解，且最大的线性无关组大小小于 n。

否则，方程组有唯一解。

概率与期望

求概率往往可以转化为求方案数。（所以归根结底还是考数数）

期望的线性性是指和的期望等于期望的和，即使拆成的两部分是相关的。

概率与期望

求概率往往可以转化为求方案数。（所以归根结底还是考数数）

期望的线性性是指和的期望等于期望的和，即使拆成的两部分是相关的。

举个例子。假设和同时做一道题，有 p_1 的概率自己做出来，有 p_2 的概率自己做出来，做出来后有 q_1 的概率告诉怎么做，做出来后有 q_2 的概率告诉怎么做。求和做出的总题数的期望。

概率与期望

求概率往往可以转化为求方案数。（所以归根结底还是考数数）

期望的线性性是指和的期望等于期望的和，即使拆成的两部分是相关的。

举个例子。假设和同时做一道题，有 p_1 的概率自己做出来，有 p_2 的概率自己做出来，做出来后有 q_1 的概率告诉怎么做，做出来后有 q_2 的概率告诉怎么做。求和做出的总题数的期望。

apio2019 奇怪装置

description

有一个奇怪的装置，在 t 时刻会显示数对 $((t+\lfloor\frac{t}{B}\rfloor) \bmod A, t \bmod B)$。给出 n 个不交的连续时间段 $[l_i, r_i]$，问这个装置在这些时间段内会显示出多少种不同的数对。

constriction

$1 \le n \le 10^6, 1 \le A, B \le 10^{18}, 0 \le l_i \le r_i \le 10^{18}$.

apio2019 奇怪装置

description

有一个奇怪的装置，在 t 时刻会显示数对 $((t+\lfloor\frac{t}{B}\rfloor) \bmod A, t \bmod B)$。给出 n 个不交的连续时间段 $[l_i, r_i]$，问这个装置在这些时间段内会显示出多少种不同的数对。

constriction

$1 \leq n \leq 10^6, 1 \leq A, B \leq 10^{18}, 0 \leq l_i \leq r_i \leq 10^{18}$.

solution

snoi2019 数论

description

给出正整数 P, Q, T，大小为 n 的整数集 A 和大小为 m 的整数集 B，求：

$$\sum_{i=0}^{T-1}[i \bmod P \in A][i \bmod Q \in B]$$

换言之，就是求有多少小于 T 的非负整数 x 满足 x 除以 P 的余数属于 A 且 x 除以 Q 的余数属于 B。

constriction

$1 \leq n, m, P, Q \leq 10^6, 1 \leq T \leq 10^{18}$.

snoi2019 数论

description

给出正整数 P, Q, T，大小为 n 的整数集 A 和大小为 m 的整数集 B，求：

$$\sum_{i=0}^{T-1} [i \bmod P \in A][i \bmod Q \in B]$$

换言之，就是求有多少小于 T 的非负整数 x 满足 x 除以 P 的余数属于 A 且 x 除以 Q 的余数属于 B。

constriction

$1 \leq n, m, P, Q \leq 10^6, 1 \leq T \leq 10^{18}$.

solution

jxoi2018 游戏

description

初始时有 $r-l+1$ 个数 $l, l+1, ..., r$，每次操作为随机选取一个数并删除其所有倍数，求删完所有数的期望操作次数模 10^9+7。

constriction

$1 \leq l \leq r \leq 10^7$.

jxoi2018 游戏

description

初始时有 $r-l+1$ 个数 $l,l+1,...,r$，每次操作为随机选取一个数并删除其所有倍数，求删完所有数的期望操作次数模 10^9+7。

constriction

$1 \leq l \leq r \leq 10^7$.

solution

51nod1769 Clarke and math 2

description

已知数论函数 $f(n), g(n)$ 满足

$$g(i) = \sum_{i_1|i} \sum_{i_2|i_1} \cdots \sum_{i_k|i_{k-1}} f(i_k)$$

给出 $f(1)...f(n)$，求 $g(1)...g(n)$ 对 10^9+7 取模后的结果。

constriction

$1 \le n \le 10^5, 1 \le k \le 10^{10^5}, 0 \le f(i) < 10^9+7.$

51nod1769 Clarke and math 2

description

已知数论函数 $f(n), g(n)$ 满足

$$g(i) = \sum_{i_1|i} \sum_{i_2|i_1} \cdots \sum_{i_k|i_{k-1}} f(i_k)$$

给出 $f(1)...f(n)$，求 $g(1)...g(n)$ 对 10^9+7 取模后的结果。

constriction

$1 \le n \le 10^5, 1 \le k \le 10^{10^5}, 0 \le f(i) < 10^9+7.$

solution

snoi2017 遗失的答案

description

求从 $\{1, 2, ..., n\}$ 中选出一个非空子集，其最大公约数恰好为 G，最小公倍数恰好为 L 的方案数。
此外有 q 次询问，每次询问给出一个正整数 a_i，询问选出的非空子集必须包含 a_i 时的方案数。
输出答案对 $10^9 + 7$ 取模后的结果。

constriction

$1 \leq n, G, L \leq 10^8, q \leq 10^5, 1 \leq a_i \leq n.$

snoi2017 遗失的答案

description

求从 $\{1,2,...,n\}$ 中选出一个非空子集，其最大公约数恰好为 G，最小公倍数恰好为 L 的方案数。
此外有 q 次询问，每次询问给出一个正整数 a_i，询问选出的非空子集必须包含 a_i 时的方案数。
输出答案对 10^9+7 取模后的结果。

constriction

$1 \le n, G, L \le 10^8, q \le 10^5, 1 \le a_i \le n.$

solution

bzoj4767 两双手

description

有一张无限大的棋盘以及两种移动方式 (x_1, y_1) 和 (x_2, y_2)，同时棋盘上存在 k 个障碍点不能经过。求从 $(0, 0)$ 走到 (n, m) 不经过障碍点的方案数模 $10^9 + 7$。

constriction

$0 \le |x_1|, |y_1|, |x_2|, |y_2|, k, n, m \le 1000, (x_1, y_1) \times (x_2, y_2) \ne 0.$

bzoj4767 两双手

description

有一张无限大的棋盘以及两种移动方式 (x_1, y_1) 和 (x_2, y_2)，同时棋盘上存在 k 个障碍点不能经过。求从 $(0,0)$ 走到 (n,m) 不经过障碍点的方案数模 10^9+7。

constriction

$0 \le |x_1|, |y_1|, |x_2|, |y_2|, k, n, m \le 1000, (x_1, y_1) \times (x_2, y_2) \neq 0.$

solution

hncpc2019H 有向图

description

一张 $n+m$ 个点的图，初始在 1 号点，每次在 i 号点时有 $P_{i,j}$ 的概率走到 j 号点，保证 $\forall i \in [n+1, n+m], P_{i,j} = [i=j]$。可以发现经过无限次行走后停留在前 n 号点上的概率是 0，求停留在后 m 号点每个点的概率模 10^9+7。

constriction

$1 \le n+m \le 500, \forall i \in [1,i], 0 < P_{i,j} < 1$ 且 $\sum_{j=1}^{n+m} P_{i,j} = 1.$

hncpc2019H 有向图

description

一张 $n+m$ 个点的图，初始在 1 号点，每次在 i 号点时有 $P_{i,j}$ 的概率走到 j 号点，保证 $\forall i \in [n+1, n+m], P_{i,j} = [i=j]$。可以发现经过无限次行走后停留在前 n 号点上的概率是 0，求停留在后 m 号点每个点的概率模 10^9+7。

constriction

$1 \leq n+m \leq 500, \forall i \in [1, i], 0 < P_{i,j} < 1$ 且 $\sum_{j=1}^{n+m} P_{i,j} = 1.$

solution

tjoi2019 唱、跳、rap 和

description

最喜欢唱、跳、rap 和的同学数量分别为 a, b, c, d。和想选出 n 个同学排成一队，但是他们不希望队伍中有连续四个人依次最喜欢唱、跳、rap 和，因为这样他们就会聚在一起讨论。求排队的方案数模 998244353，两种方案不同当且仅当存在某个位置上的同学的喜好不同。

constriction

$1 \le n, a, b, c, d \le 5000, a + b + c + d \ge n.$

tjoi2019 唱、跳、rap 和

description

最喜欢唱、跳、rap 和的同学数量分别为 a, b, c, d。和想选出 n 个同学排成一队，但是他们不希望队伍中有连续四个人依次最喜欢唱、跳、rap 和，因为这样他们就会聚在一起讨论。求排队的方案数模 998244353，两种方案不同当且仅当存在某个位置上的同学的喜好不同。

constriction

$1 \leq n, a, b, c, d \leq 5000, a + b + c + d \geq n.$

solution

codeforces1097D Makoto and a Blackboard

description

有一个数 n，每次操作为随机选择 n 的一个约数 m 并将 n 替换成 m，求 k 次操作后 n 的期望值模 10^9+7。

constriction

$1 \leq n \leq 10^{15}, 1 \leq k \leq 10^4.$

codeforces1097D Makoto and a Blackboard

description

有一个数 n，每次操作为随机选择 n 的一个约数 m 并将 n 替换成 m，求 k 次操作后 n 的期望值模 10^9+7。

constriction

$1 \le n \le 10^{15}, 1 \le k \le 10^4.$

solution

codeforces1097F Alex and a TV Show

description

你需要编写一种数据结构来维护 n 个可重集合，支持四种操作：

- 将 S_x 设为 $\{v\}$
- 将 S_x 设为 $S_y + S_z$，操作后 $|S_x| = |S_y| + |S_z|$
- 将 S_x 设为 $S_y \times S_z$，这里的 $A \times B$ 定义为 $\{\gcd(a, b) | a \in A, b \in B\}$，操作后 $|S_x| = |S_y| \times |S_z|$
- 询问 S_x 中 v 元素的数量

由于一些原因，你只需要输出每个询问的答案对 2 取模后的结果。

constriction

$1 \leq n, m \leq 10^5, 1 \leq v \leq 7000.$

codeforces1097F Alex and a TV Show

solution

谢谢大家！

感谢，、以及的友情出镜。