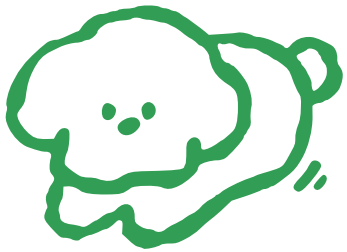
线代和数论

数论和线代

菜汪酱

。

2024 年 2 月 19 日



前言

- 线代好好讲要一个学期
- 而且我线代很差
- 线性代数在 OI 中的应用郭雨豪.pdf

高斯消元

- m 个方程 n 个未知数。
- 第 i 个方程形如 $\sum_{j=1}^{n} a_{i,j}x_j = b_i$
- 非齐次线性方程组。齐次就是后面是 0。
- 可以证明经过一下的操作得到的方程组和原来的方程组同解：
  - 某一个方程乘上一个非零数
  - 某一个方程加到另一个方程
- 高斯消元就是用这两条去得到解。

线性基

- 没有回代的高斯消元。
- 得到一组线性无关向量。
- 加入一个新的向量时看是否能用之前的向量线性表出。
- 比较多的是数域为 $\mathbb{Z}[2]$

线性基题目

- 异或最大值。

线性基题目

- 最大 XOR 和路径

行列式

- 一个矩阵 A 的行列式定义为：

$$\sum_{p}(-1)^{\sigma(p)}\prod_{i=1}^{n}A_{i,p_i}$$

  其中 $\sigma(p)$ 定义为排列 p 的逆序对的个数。
- 满足线性性、自反性。
- 用这两条搞高斯消元就能得到行列式了。

lgv 引理

- 有一个 DAG，每条边上有权，$\omega(P)$ 表示 P 这条路径上所有边权的乘积。
- $e(u,v)$ 表示所有 $u \to v$ 的路径权值的和，即 $e(u,v)=\sum_{P:u\to v}\omega(P)$
- 有两个大小都为 n 的数组 A,B 表示起点和终点。一组不相交路径 S 满足 S_i 是 $A_i \to B_{\sigma(S)_i}$ 的路径，$\forall i\neq j, S_i\cap S_j=\varnothing$。
- $\mathrm{sgn}(\sigma)$ 表示排列 σ 的逆序对个数。

lgv 引理

- $$M = \begin{bmatrix} e(A_1, B_1) & e(A_1, B_2) & \cdots & e(A_1, B_n) \\ e(A_2, B_1) & e(A_2, B_2) & \cdots & e(A_2, B_n) \\ \vdots & \vdots & \ddots & \vdots \\ e(A_n, B_1) & e(A_n, B_2) & \cdots & e(A_n, B_n) \end{bmatrix}$$

$$\det(M) = \sum_{S:A\to B} (-1)^{\mathrm{sgn}(\sigma(S))} \prod_{i=1}^{n} \omega(S_i)$$

lgv 题目

- 给定一个 $n \times m$ 的矩阵 A，问有多少填数的方案满足
  - $1 \le A_{i,j} \le v$
  - $\forall 2 \le i \le n, 1 \le j \le m, A_{i-1,j} \le A_{i,j}$
  - $\forall 1 \le i \le n, 2 \le j \le m, A_{i,j-1} \le A_{i,j}$
- 答案对质数取模。
- $n, m \le 10^6, v \le 100$。
- 5min。

一些简单的记号

- $\gcd(a,b)$ 表示 a,b 的最大公约数
- $\operatorname{lcm}(a,b)$ 表示 a,b 的最大公倍数
- $a \perp b$ 表示 $\gcd(a,b)=1$
- $a|b$ 表示 $b \bmod a = 0$
- $[p]$ 表示 p 成立为 1 否则为 0

一些基础知识

- 裴蜀定理：a, b 是不全为零的整数，存在整数 x, y 使得 $ax + by = \gcd(a, b)$。
- 费马小定理：若 p 为素数，$\gcd(a, p) = 1$，则有 $a^{p-1} \equiv 1 \pmod{p}$。
- 欧拉定理：若 $\gcd(a, m) = 1$，则 $a^{\varphi(m)} \equiv 1 \pmod{m}$。
- 扩展欧拉定理：

$$a^b \equiv \begin{cases} a^{b \bmod \varphi(m)} & , a \perp b \\ a^b & , a \not\perp b, b < \varphi(m) \\ a^{(b \bmod \varphi(m)) + \varphi(m)} & , a \not\perp b, b \geq \varphi(m) \end{cases}$$

P4139 上帝与集合的正确用法

- 计算：

$$2^{2^{2^{2^{\cdots}}}} \bmod p$$

- $T \leq 10^3, p \leq 10^7$。
- $1\text{s}, 125\text{MB}$。
- 1min。

P4139 上帝与集合的正确用法

- 考虑

$$2^{2^{2^{2^{\cdots}}}} \bmod p = 2^{2^{2^{2^{\cdots}}} \bmod \varphi(p)+\varphi(p)} \bmod p$$

- 递归求解即可。
- 次数是 $\mathcal{O}(\log p)$ 的。

Miller Rabin

- 二次探测：如果 p 是一个素数，$0 < x < p$，$x^2 \equiv 1 \pmod{p}$ 的解为 $x = 1$ 或 $x = p - 1$。
- 如果 $a^{(p-1)/2^t} \equiv 1 \pmod{p}$ 但是 $a^{(p-1)/2^{t+1}}$ 不是 1 或 $p - 1$ 那么就一定不是质数。
- 对于 $[1, 2^{32})$ 范围内的数，选取 $a = \{2, 7, 61\}$
- 对于 $[1, 2^{64})$ 范围内的数，选取前 12 个质数。
- ~~如果你坚信不会被卡，那么少选几个也没关系。~~

数论函数

- 你有一个函数 $f(i)$
- 定义域是 $\mathbb{N}^+$
- 也可以看作是一个序列
- 积性函数：$\forall a \perp b, f(a)f(b) = f(ab)$
- 完全积性函数：$\forall a, b, f(a)f(b) = f(ab)$
- 一些常见的积性函数：
  - 单位函数：$\epsilon(n) = [n = 1]$。
  - 恒等函数：$\mathrm{id}_k(n) = n^k$。$\mathrm{id}_1(n)$ 也记作 $\mathrm{id}(n)$。
  - 常数函数：$I(n) = 1$。
  - 除数函数：$\sigma_k(n) = \sum_{d|n} d^k$。$\sigma_0(n)$ 也记作 $d(n)$，$\sigma_1(n)$ 也记作 $\sigma(n)$。
  - 欧拉函数：$\varphi(n) = \sum_{i=1}^{n} [i \perp n]$。
  - 莫比乌斯函数：$\mu(n) = \begin{cases} 0 & \exists d > 1, d^2 | n \\ (-1)^{n\text{的质因子个数}} & \text{otherwise} \end{cases}$。

狄利克雷卷积

- $h = f * g \iff h(i) = \sum_{j|i} f(j) g\left(\frac{i}{j}\right)$
- 暴力求前 n 项时间复杂度是 $\mathcal{O}(n \log n)$。
- 交换律：$f * g = g * f$
- 结合律：$(f * g) * h = f * (g * h)$
- 分配律：$(f + g) * h = f * h + g * h$
- 等式的性质：$f = g \iff f * h = g * h$（$h(1) \neq 0$）
- 单位元：$f * \epsilon = f$
- 逆元：对于 $f(1) \neq 0$，存在逆元满足 $f * g = \epsilon$。逆元唯一。

$$g(x) = \left(\epsilon(x) - \sum_{d|x, d \neq 1} f(d) g\left(\frac{x}{d}\right)\right) \Big/ f(1)$$

几个例子

- $d = I * I$
- $\sigma = \mathrm{id} * I$
- $\epsilon = \mu * I$
- $\mathrm{id} = \varphi * I$
- $\varphi = \mu * \mathrm{id}$

积性函数的性质

- 两个积性函数的狄利克雷卷积也是积性函数。（5min）
- 对于 $h = f * g$:

$$\begin{aligned} h(a)h(b) &= \left(\sum_{d_1|a} f(d_1) g\left(\frac{a}{d_1}\right)\right)\left(\sum_{d_2|b} f(d_2) g\left(\frac{b}{d_2}\right)\right) \\ &= \sum_{d|ab} f(d) g\left(\frac{ab}{d}\right) \\ &= h(ab) \end{aligned}$$

积性函数的性质

- 积性函数的逆也是积性函数（5min）
- 对于 $f*g=\epsilon$，不妨假设 $f(1)=1$，显然 $g(1)=1$，因此如果 a 或 b 中有一个 1 一定成立。接下来假设 $a,b>1$:

$$
\begin{aligned}
g(ab) &= -\sum_{d_1|a,d_2|b,d_1\,d_2\neq 1} f(d_1\,d_2)g\left(\frac{ab}{d_1\,d_2}\right)\\
&= -\sum_{d_1|a,d_2|b,d_1\,d_2\neq 1} f(d_1)f(d_2)g\left(\frac{a}{d_1}\right)g\left(\frac{b}{d_2}\right)\\
&= f(1)f(1)g(a)g(b)-\left(\sum_{d_1|a} f(d_1)g\left(\frac{n}{d_1}\right)\right)\left(\sum_{d_2|b} f(d_2)g\left(\frac{b}{d_2}\right)\right)\\
&= g(a)g(b)
\end{aligned}
$$

狄利克雷前缀和

- 就是求 $g = f * I$
- 也就是 $g_j = \sum_{i|j} a_i$
- $i = \prod p_k^{\alpha_k}, j = \prod p_k^{\beta_k}$，$i$ 能贡献到 j 当且仅当 $\forall k, \alpha_k \le \beta_k$。

Algorithm: 狄利克雷前缀和

```
for p ∈ ℙ, p ≤ n do
    for i ← 1 to ⌊n/p⌋ do
        f_{p×i} ← f_{p×i} + f_i
    end
end
```

- 时间复杂度 $\mathcal{O}(n \log \log n)$。
- 倒着做就是狄利克雷差分。也就是乘 μ。

莫比乌斯反演

- 简单证明一下 $\mu * I = \epsilon$

$$
\begin{aligned}
(\mu * I)(n) = (\mu * I)(p_1^{\alpha_1} \dots p_k^{\alpha_k}) &= \sum_{d|n} \mu(d) \\
&= \sum_{d|p_1 \dots p_k} \mu(d) = \sum_{i=0}^{k} \binom{k}{i} (-1)^i \\
&= \epsilon(n)
\end{aligned}
$$

- 实际使用中用得更多的是 $[\gcd(i,j) = 1] = \sum_{d|i,j} \mu(d)$
- 拿刚刚的 $\varphi = \mu * \mathrm{id}$ 举个例子

莫比乌斯反演

- $$\begin{aligned}\varphi(n) &= \sum_{i=1}^{n}[\gcd(i,n)=1] = \sum_{i=1}^{n}\sum_{d|i,d|n}\mu(d)\\ &= \sum_{d|n}\mu(d)\sum_{i=1}^{n}[d|i] = \sum_{d|n}\mu(d)\,\mathrm{id}\left(\frac{n}{d}\right)\\ &= (\mu * \mathrm{id})(n)\end{aligned}$$
- 两边同乘 I 可以得到 $\varphi * I = \mathrm{id}$。

CF585E Present for Vitalik the Philatelist

- 有一个长度为 n 的数组 a
- 求有多少二元组 (x, S) 满足：
  - $1 \le x \le n, S \subseteq \{1, 2, \ldots, n\}, S \neq \varnothing$
  - $\gcd_{i \in S} a_i \neq 1$
  - $\gcd_{i \in S \cup \{x\}} a_i = 1$
- 对 998244353 取模。
- $n \le 5 \times 10^5, 2 \le a_i \le 10^7$
- $5\text{s}, 256\text{MB}$。
- 5min

CF585E Present for Vitalik the Philatelist

- 一个想法是枚举 $\gcd_{i\in S} a_i = x$，设这个的方案数是 g_x，然后 f_x 表示与 x 互质的数的个数，那么就是要求 $\sum_{i>1} f_i \times g_i$。
- 然后考虑 f_x 怎么求，记 c_i 表示值为 i 的个数：

$$f_x = \sum [\gcd(i,x) = 1] c_i = \sum_{d|x} \mu(d) \left(\sum_{d|i} c_i \right)$$

- 然后后面一项就相当于对 c_i 求狄利克雷后缀和，不妨记为 s_i，然后再求狄利克雷前缀和。
- 然后再考虑 g_x 怎么求，设 g'_x 表示 gcd 是 x 的倍数的集合个数，那么显然 $g'_x = \sum_{d|x} g_d = 2^{s_x} - 1$，那么这个就反着来搞一下狄利克雷差分即可。
- 复杂度 $\mathcal{O}(n + W \log\log W)$。

数论分块

- $\left\lfloor \frac{n}{i} \right\rfloor$ 只有 $\leq 2\sqrt{n}$ 种取值
- $i \leq \sqrt{n}$ 只有 $\sqrt{n}$ 个，$i \geq \sqrt{n}$ 值 $\leq \sqrt{n}$。
- 计算 $\sum_{i=1}^{n} f(i) \left\lfloor \frac{n}{i} \right\rfloor$，如果能 $\mathcal{O}(1)$ 计算 $f(i)$ 前缀和 $s(i)$：

Algorithm: 简单数论分块

```
1 l ← 1;
2 r ← 1;
3 result ← 0;
4 while l ≤ n do
5 |   r ← ⌊n/⌊n/i⌋⌋;
6 |   result ← result + [s(r) − s(l − 1)] × ⌊n/l⌋;
7 |   l ← r + 1;
8 end
```

- 时间复杂度 $\mathcal{O}(\sqrt{n})$。

P2522 [HAOI2011] Problem b

- T 组询问。每次给出 l_1, r_1, l_2, r_2, k，计算：

$$\sum_{i=l_1}^{r_1}\sum_{j=l_2}^{r_2}[\gcd(i,j)=k]$$

- $1 \le T, l_1, r_1, l_2, r_2, k \le 5 \times 10^4$。
- $2.5\text{s}, 250\text{MB}$。
- 5min。

P2522 [HAOI2011] Problem b

- 首先变成 4 次

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

- 然后就相当于

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

- 然后直接上莫反：

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

P2522 [HAOI2011] Problem b

- 交换求和顺序

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

- 后面两个 $\sum$ 其实是没有必要的：

$$\sum_{d=1}^{\min\left(\left\lfloor\frac{n}{k}\right\rfloor,\left\lfloor\frac{m}{k}\right\rfloor\right)} \mu(d) \left\lfloor\frac{n}{dk}\right\rfloor \left\lfloor\frac{m}{dk}\right\rfloor$$

- 使用整除分块，单次复杂度 $\mathcal{O}(\sqrt{n})$。

埃氏筛法

- 代码很好写。

Algorithm: 埃氏筛

```
1 npr ← [1 ... n];
2 for i ← 2 to n do
3 |   if not npr[i] then
4 |   |   for j ← 2 to ⌊n/i⌋ do
5 |   |   |   npr[i × j] ← 1;
6 |   |   end
7 |   end
8 end
```

- 时间复杂度 $\mathcal{O}(n \log \log n)$。
- 实际表现还是不错的。

埃氏筛法

- 切比雪夫素数定理：$\pi(n)$ 表示 $\leq n$ 的质数数量，那么

$$\exists 0 < A < B, \text{s.t.} \frac{Ax}{\log x} < \pi(x) < \frac{Bx}{\log x}$$

- 也就是说 $\pi(n) = \mathcal{O}\left(\frac{n}{\log n}\right)$。
- 换句话说就是 $p_i = \mathcal{O}(i \log i)$。
- 然后就能说明复杂度了：

$$\sum_{i=1}^{\pi(n)} \frac{1}{i \log i} = \mathcal{O}(\int_1^{\pi(n)} \frac{1}{x \log x} \mathrm{d}x) = \mathcal{O}(\log \log n)$$

线性筛

Algorithm: 线性筛

```
1  npr ← [1...n]; pr ← [1...n]; tot ← 0;
2  for i ← 2 to n do
3      if not npr[i] then
4          tot ← tot + 1; pr[tot] ← i;
5      end
6      j ← 1;
7      while j ≤ tot and i × pr[j] ≤ n do
8          npr[i × pr[j]] ← 1;
9          if i mod pr[j] = 0 then
10             break;
11         end
12         j ← j + 1;
13     end
```

线性筛

- 如果 $i \bmod pr_j = 0$，那么对于 $k > j$，$i \times pr_k$ 会被 $(i \times pr_k / pr_j) \times pr_j$ 筛掉，也就没有必要继续跑了。
- 时间复杂度 $\mathcal{O}(n)$。
- 实际上也求出了每个数的最小质因子。

常见积性函数的筛法

- 对于 id_k，用快速幂算 $\mathrm{id}_k(p)$，其他数的值都能 $\mathcal{O}(1)$ 得到。
- 对于 φ，如果 $i \bmod p \neq 0$，那么 $\varphi(i \times p) = \varphi(i) \times (p-1)$，否则 $\varphi(i \times p) = \varphi(i) \times p$
- 对于 μ，$i \bmod p \neq 0$，那么 $\mu(i \times p) = -\mu(i)$，否则 $\mu(i \times p) = 0$

一般积性函数的筛法

- 如果能 $\mathcal{O}(n)$ 求出所有 $f(p^c)$ 那么就能做。
- 刚才已经求出最小质因子 mn。
- 记录一个 pw_i 表示 $\max_{mn_i^\alpha | i} \alpha$，这个也可以 $\mathcal{O}(n)$ 求出。
- 然后就 $f(i) = f(i/pw_i)f(pw_i)$ 就行了。
- 以刚才的 $g = f * \mathrm{id}_2 * I$ 为例（$f(x) = x\mu(x)$）。
- 那么 $g(p^c) = \sum_{i=0}^{c} p^{2i} - p\sum_{i=0}^{c-1} p^{2i} = ((p^{2(c+1)} - 1) - p(p^{2c} - 1))/(p^2 - 1)$。

杜教筛

- 记 $S(f,n)=\sum_{i=1}^{n}f(i)$，也就是前缀和。
- 以下讨论的数论函数满足 $f(1)=1$。
- 我们想要求 $S(f,n)$，考虑另一个数论函数满足 $f*g=h$：

$$\begin{aligned}S(h,n)&=\sum_{i=1}^{n}\sum_{d|i}g(d)f\left(\frac{i}{d}\right)\\&=\sum_{d=1}^{n}g(d)\sum_{i=1}^{\lfloor\frac{n}{d}\rfloor}f(i)\\&=\sum_{d=1}^{n}g(d)S\left(f,\left\lfloor\frac{n}{d}\right\rfloor\right)\end{aligned}$$

杜教筛

- 把 $S(f,n)$ 单独提出来：

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

- 我们把计算所有 $S\left(f,\left\lfloor\frac{n}{d}\right\rfloor\right)$ 的值叫做**块筛**。
- 做的时候需要求出 f 的块筛，显然需要有 h 的块筛，然后跑整除分块的时候发现还要 g 的块筛。
- 如果已经求出了 g,h 的块筛，就能 $\mathcal{O}\left(n^{\frac{3}{4}}\right)$ 计算 f 的块筛。
- 这个 $n^{\frac{3}{4}}$ 是怎么来的呢？

$$O\left(\sum_{i=1}^{\sqrt{n}}\sqrt{i}+\sqrt{\frac{n}{i}}\right)=O\left(\int_{1}^{\sqrt{n}}\left(\sqrt{i}+\sqrt{\frac{n}{i}}\right)\mathrm{d}i\right)=\mathcal{O}\left(n^{\frac{3}{4}}\right)$$

杜教筛

- 假设我们用线性筛求出了 $S(i)$ 的前 k 项，那么还能进一步优化：

$$\mathcal{O}\left(k+\sum_{i=1}^{\lfloor n/k\rfloor}\sqrt{\frac{n}{i}}\right)=\mathcal{O}\left(k+\frac{n}{\sqrt{k}}\right)$$

- 当 $k=\mathcal{O}(n^{\frac{2}{3}})$ 时取最小值 $\mathcal{O}(n^{\frac{2}{3}})$。
- 数论函数点乘：$f\cdot g \iff (f\cdot g)(n)=f(n)g(n)$。
- 拆点乘：如果 c 是完全积性函数，那么 $(a\cdot c)*(b\cdot c)=(a*b)\cdot c$。

杜教筛例题

- 计算 $f=\mu$ 的块筛。
- 选取 $g=I$，那么 $f*g=\epsilon$。
- 计算 $f=\varphi$ 的块筛。
- 选取 $g=I$，那么 $f*g=\mathrm{id}$。
- 计算 $f=\mathrm{id}_k\cdot\varphi$ 的块筛。
- 选取 $g=\mathrm{id}_k$，那么 $f*g=(\varphi*I)\cdot\mathrm{id}_k=\mathrm{id}\cdot\mathrm{id}_k=\mathrm{id}_{k+1}$。
- 计算 $f=\mathrm{id}_k\cdot\mu$ 的块筛。
- 选取 $g=\mathrm{id}_k$，那么 $f*g=\epsilon$。

Thanks!