浅谈算法竞赛中的数论函数

Hong

SUSTech珂学研究院

一些定义和性质

- $\lceil \frac{n}{ab} \rceil = \lceil \frac{\lceil \frac{n}{a} \rceil}{b} \rceil, \lfloor \frac{n}{ab} \rfloor = \lfloor \frac{\lfloor \frac{n}{a} \rfloor}{b} \rfloor$
- $\lceil \frac{n}{i} \rceil$只有$O(\sqrt{N})$种取值
- *数论函数*：定义域为正整数，陪域为复数的函数。今天我们主要研究定义域为正整数，值域为整数的函数。
- *积性函数*：满足 若a, b互质，则$f(ab) = f(a)f(b)$ 的数论函数称为积性函数。
- *完全积性函数*：满足 $f(ab) = f(a)f(b)$ 的数论函数称为完全积性函数。
- *狄利克雷卷积*：设f, g为两个数论函数，则满足 $h(n) = \sum_{d|n} f(d)g(\frac{n}{d})$的函数称为$f$和$g$的狄利克雷卷积。也可写作$h(n) = \sum_{ij=n} f(i)g(j)$

一些定义和性质

- 两个积性函数的狄利克雷卷积仍为积性函数。
- 狄利克雷卷积满足交换律和结合律。
- 考虑使用$h(n)=\sum_{ij=n}f(i)g(j)$这个定义式，那么多个函数的狄利克雷卷积为$h(n)=\sum_{i_1i_2\ldots i_k=n}f_1(i_1)f_2(i_2)\ldots f_k(i_k)$由乘法的交换律和结合律可以直接得到狄利克雷卷积的交换律和结合律。

一些常见的积性函数

- 单位函数$e(x) = \begin{cases} 1, x = 1 \\ 0, x > 1 \end{cases}$
- 常函数$I(x) = 1$
- 幂函数$id(x) = x^k$
- 欧拉函数$\phi(x) = [1, x]$中与x互质的数的个数 $= x\Pi_{p|x, p\ is\ a\ prime}(1 - \frac{1}{p})$
- 莫比乌斯函数$\mu(x) = \begin{cases} 1, x = 1 \\ (-1)^k, x = p_1p_2...p_k \\ 0, x = others \end{cases}$

Count $a \times b$

定义$f(n)$ =选两个$[0,n)$的整数a,b，且$a \times b$不是n的倍数的方案数。

求$g(n) = \sum_{d|n} f(d)$

数据组数$1 \le T \le 20000$，$1 \le n \le 10^9$

$$f(n) = \sum_{a=0}^{n-1}\sum_{b=0}^{n-1}[n \nmid ab] = \sum_{a=1}^{n}\sum_{b=1}^{n}[n \nmid ab]$$

$$= n^2 - \sum_{a=1}^{n}\sum_{b=1}^{n}[n|ab] = n^2 - \sum_{a=1}^{n}\sum_{b=1}^{n}[\frac{n}{gcd(a,n)} | \frac{a}{gcd(a,n)} \cdot b]$$

$$= n^2 - \sum_{a=1}^{n}\sum_{b=1}^{n}[\frac{n}{gcd(a,n)} | b] = n^2 - \sum_{a=1}^{n} gcd(a,x)$$

$$= n^2 - \sum_{d|n} d \cdot \phi(\frac{n}{d})$$

令$h(n)=\sum_{d|n} d\cdot\phi(\frac{n}{d})$，显然$h(n)$为幂函数与欧拉函数的狄利克雷卷积，仍为积性函数。

则$g(n)=\sum_{d|n} f(d)=\sum_{d|n} d^2-\sum_{d|n} h(d)$。

设$P(n)=\sum_{d|n} d^2$，$Q(n)=\sum_{d|n} h(d)$。则P,Q均为积性函数。

只需要计算所有的$P(p^k)$和$Q(p^k)$，乘起来就是$P(n)$和$Q(n)$。

或是继续化简

$Q(n)=\sum_{d|n} h(d)=\sum_{d|n}\sum_{w|d}\phi(w)\cdot\frac{d}{w}=\sum_{d|n}\sum_{i|\frac{n}{d}} d\cdot\phi(i)$

$=\sum_{d|n} d\sum_{i|\frac{n}{d}}\phi(i)=\sum_{d|n} d\cdot\frac{n}{d}=\sum_{d|n} n=n\cdot n$的约数个数。

(2015ICPC ChangChun regional B)

数论函数前缀和

给出一个积性函数f，求$\sum_{i=1}^{n} f(i)$。

暴力求前缀和的复杂度显然不小于$O(n)$。

如果$f(p)$可以在$O(logn)$的时间复杂度内求出，则求出质数项的总时间为$O(n)$；通常，$f(p^k)$可以比较容易的由$f(p^{k-1})$等数递推出来。可以用类似欧拉筛的线性筛过程q求出前n项的值，时间复杂度为$O(n)$。

GRISAIA

求$\sum_{i=1}^{n}\sum_{j=1}^{i} n\%(i \times j), 1 \le n \le 10^{11}$

解法一

$\sum_{i=1}^{n}\sum_{j=1}^{i} n\%(i\times j)=\sum_{i=1}^{n}\sum_{j=1}^{i} n-\lfloor\frac{n}{ij}\rfloor ij$。现在只需要求$\sum_{i=1}^{n}\sum_{j=1}^{i}\lfloor\frac{n}{ij}\rfloor ij$。

$\sum_{i=1}^{n}\sum_{j=1}^{i}\lfloor\frac{n}{ij}\rfloor ij=\frac{\sum_{i=1}^{n}\sum_{j=1}^{n}\lfloor\frac{n}{ij}\rfloor ij+\sum_{i=1}^{n}\lfloor\frac{n}{i^2}\rfloor i^2}{2}$。重点是如何计算$\sum_{i=1}^{n}\sum_{j=1}^{n}\lfloor\frac{n}{ij}\rfloor ij$。

定义$f(n)=\sum_{i=1}^{n}\lfloor\frac{n}{i}\rfloor i$，则

$\sum_{i=1}^{n}\sum_{j=1}^{n}\lfloor\frac{n}{ij}\rfloor\times i\times j=\sum_{i=1}^{n}\sum_{j=1}^{n} i\times\lfloor\frac{\lfloor\frac{n}{i}\rfloor}{j}\rfloor\times j$

$=\sum_{i=1}^{n} i\times(\sum_{j=1}^{n}\lfloor\frac{\lfloor\frac{n}{i}\rfloor}{j}\rfloor\times j)=\sum_{i=1}^{n} i\times f(\lfloor\frac{n}{i}\rfloor)$

$O(\sqrt{n})$地枚举$\lfloor\frac{n}{i}\rfloor$的值，$O(\sqrt{\lfloor\frac{n}{i}\rfloor})$地计算$f(\lfloor\frac{n}{i}\rfloor)$，时间复杂度为$O(\sum_{i=1}^{\sqrt{n}}\sqrt{\lfloor\frac{n}{i}\rfloor})=O(n^{\frac{3}{4}})$

解法二

考虑优化解法一。

定义$g(n)=f(n)-f(n-1)$。

$g(n)=\sum_{i=1}^{n} i\times(\lfloor\frac{n}{i}\rfloor-\lfloor\frac{n-1}{i}\rfloor)=\sum_{i|n} i$

使用欧拉筛求出g的前$n^{\frac{2}{3}}$项，再求一下前缀和，就可以得到f的前$n^{\frac{2}{3}}$项，复杂度为$O(n^{\frac{2}{3}})$。

对于$\lfloor\frac{n}{i}\rfloor>n^{\frac{2}{3}}$，暴力计算$f$，时间复杂度为$O(\sum_{i=1}^{n^{\frac{1}{3}}}\sqrt{\lfloor\frac{n}{i}\rfloor})=O(n^{\frac{2}{3}})$

(2018 Sichuan Province Contest)

莫比乌斯反演

定义：

$\forall n, f(n) = \sum_{d|n} g(d) \Leftrightarrow \forall n, g(n) = \sum_{d|n} \mu(d) f(\frac{n}{d})$。

即：$f = g * I \Leftrightarrow g = \mu * f$。

证明：

先证明$\mu * I = e$：

$\forall n > 1, \sum_{d|n} \mu(d) = \sum_{i=0}^{k} C_k^i \times (-1)^i = (1-1)^k = 0$

利用狄利克雷卷积的交换律和结合律，

$f = g * I \Rightarrow \mu * f = g * (\mu * I) = g * e = g$

$g = \mu * f \Rightarrow g * I = f * (\mu * I) = f * e = f$

QED

一些常用莫比乌斯恒等式

$\phi(d) * I = n$

即$\sum_{d|n} \phi(d) = n, \phi(d) = \sum_{d|n} \mu(d) \frac{n}{d}$

$\omega(n)$表示n的质因子个数

$2^{\omega(n)} = \sum_{d|n} \mu^2(d)$

$\mu^2(d) = \sum_{k^2|d} \mu(k)$

约数个数和

设$d(x)$为x的约数个数，给定N, M，求$\sum_{i=1}^{N}\sum_{j=1}^{M} d(ij)$

数据组数$T \leq 50000, 1 \leq N, M \leq 50000$。

设ij质因数分解为$p_1^{k_1}p_2^{k_2}...p_n^{k_n}$，则

$d(ij)=(k_1+1)(k_2+1)...(k_n+1)$

考虑一个质数p对$d(ij)$的贡献。假设i质因数分解中p的次数为a，j质因数分解中p的次数为b，则p对$d(ij)$产生的贡献为$a+b+1$。

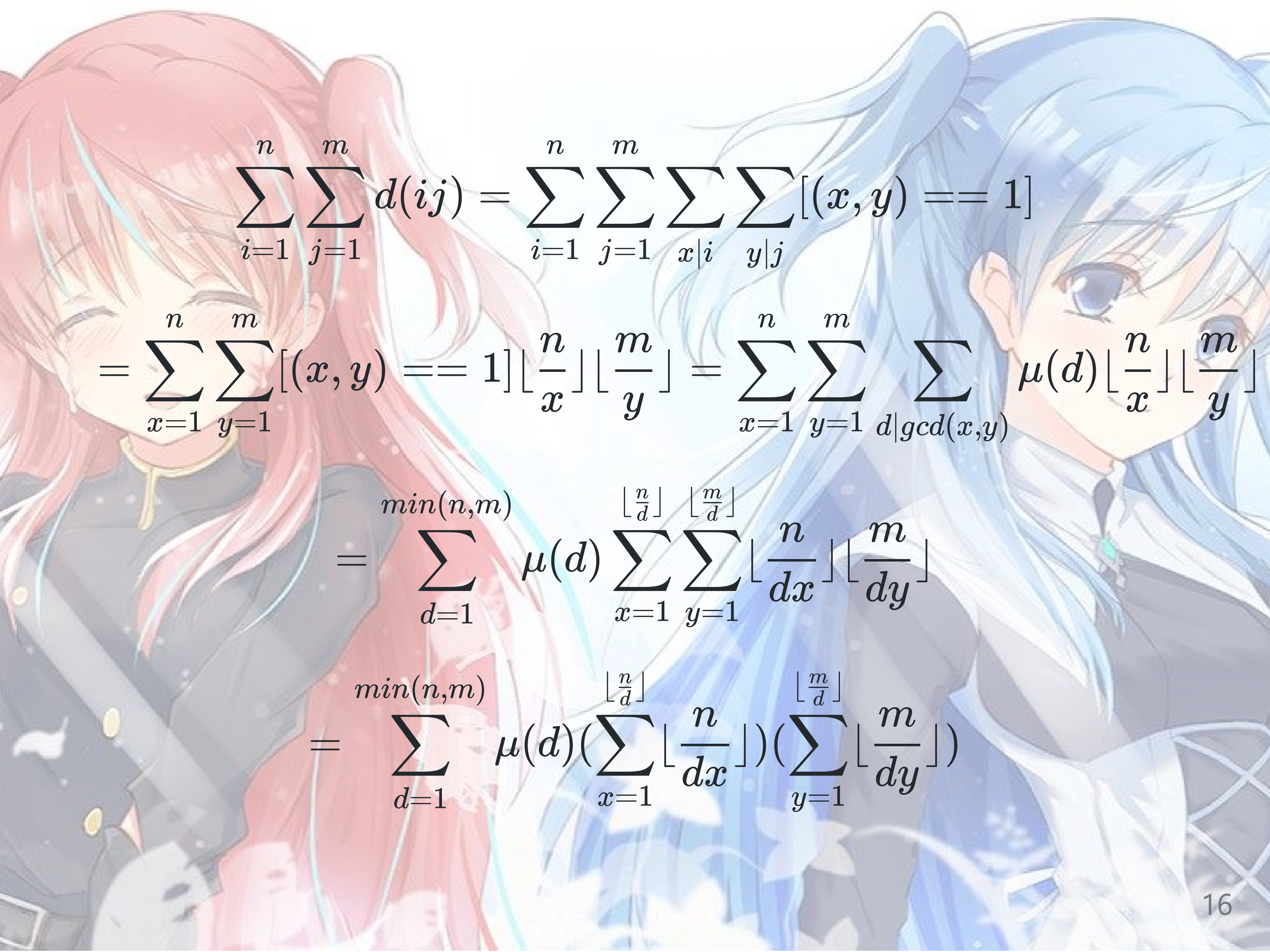
$a+b+1=\sum_{x=0}^{a}\sum_{y=0}^{b}[gcd(p^x,p^y)==1]$

考虑每个质数p_i的贡献，由乘法原理可得

$$d(ij)=\sum_{x_1=0}^{a_1}\sum_{y_1=0}^{b_1}[({p_1}^{x_1},{p_1}^{y_1})==1]...\sum_{x_n=0}^{a_n}\sum_{y_n=0}^{b_n}[({p_n}^{x_n},{p_n}^{y_n})==1]$$

令$x=x_1x_2...x_n,y=y_1y_2...y_n$，则

$$d(ij)=\sum_{x|i}\sum_{y|j}[(x,y)==1]$$



$$\sum_{i=1}^{n}\sum_{j=1}^{m} d(ij) = \sum_{i=1}^{n}\sum_{j=1}^{m}\sum_{x|i}\sum_{y|j}[(x,y)==1]$$

$$=\sum_{x=1}^{n}\sum_{y=1}^{m}[(x,y)==1]\lfloor\frac{n}{x}\rfloor\lfloor\frac{m}{y}\rfloor = \sum_{x=1}^{n}\sum_{y=1}^{m}\sum_{d|gcd(x,y)}\mu(d)\lfloor\frac{n}{x}\rfloor\lfloor\frac{m}{y}\rfloor$$

$$=\sum_{d=1}^{min(n,m)}\mu(d)\sum_{x=1}^{\lfloor\frac{n}{d}\rfloor}\sum_{y=1}^{\lfloor\frac{m}{d}\rfloor}\lfloor\frac{n}{dx}\rfloor\lfloor\frac{m}{dy}\rfloor$$

$$=\sum_{d=1}^{min(n,m)}\mu(d)(\sum_{x=1}^{\lfloor\frac{n}{d}\rfloor}\lfloor\frac{n}{dx}\rfloor)(\sum_{y=1}^{\lfloor\frac{m}{d}\rfloor}\lfloor\frac{m}{dy}\rfloor)$$

$$\sum_{d=1}^{min(n,m)} \mu(d)(\sum_{x=1}^{\lfloor\frac{n}{d}\rfloor}\lfloor\frac{n}{dx}\rfloor)(\sum_{y=1}^{\lfloor\frac{m}{d}\rfloor}\lfloor\frac{m}{dy}\rfloor)$$

预处理μ的前缀和，对后半部分整数分块即可。时间复杂度为$O(\sqrt{N})$。

(SDOI2015)

rng_58-clj等式

$$d(abc)=\sum_{gcd(i,j)==gcd(j,k)==gcd(k,i)==1}\lfloor\frac{a}{i}\rfloor\lfloor\frac{b}{j}\rfloor\lfloor\frac{c}{k}\rfloor$$

可推广至任意维。

杜教筛

设$S(n)=\sum_{i=1}^{n}f(i)$，杜教筛将求$S(n)$的过程一般化。

$\forall$数论函数g，设$h=f*g$，有$\sum_{i=1}^{n}h(i)=\sum_{i=1}^{n}g(i)S(\lfloor\frac{n}{i}\rfloor)$

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

如果我们可以$O(\sqrt{n})$计算$\sum_{i=1}^{n}h(i)$，$O(1)$计算g的前缀和，就可以将原问题递归为同类子问题，时间复杂度为

$O(\sum_{i=1}^{\sqrt{n}}\sqrt{\lfloor\frac{n}{i}\rfloor})=O(n^{\frac{3}{4}})$

如果f有一些比较好的性质，比如是积性函数，我们可以用欧拉筛求出前$n^{\frac{2}{3}}$项，更后面的项再递归，时间复杂度为$O(n^{\frac{2}{3}})$。

欧拉函数前缀和

求$\sum_{i=1}^{n}\phi(i), 1 \le n \le 10^{11}$，答案取模。

设$S(n)=\sum_{i=1}^{n}\phi(i), g(n)=1, h=\phi*g=n$

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

代入g,h的值，有$S(n)=\sum_{i=1}^{n}i-\sum_{i=2}^{n}S(\lfloor\frac{n}{i}\rfloor)$

用欧拉筛求出欧拉函数的前$n^{\frac{2}{3}}$项，求前缀和得到S的前$n^{\frac{2}{3}}$项。之后的递归计算，总复杂度为$O(n^{\frac{2}{3}})$。

Thinks for Listening.