

# 数论 3

## 杜教筛

### 算法简介

一种  $O\left(n^{\frac{2}{3}}\right)$  计算积性函数前缀和的算法。

### 算法思路

设  $f, g$  为积性函数  $S(n) = \sum_{i=1}^n f(i)$  考虑  $f, g$  的狄利克雷卷积的前缀和

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

所以有

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

变形得

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

观察式子，发现如果能快速求出  $(f * g)(n)$  和  $g(n)$  的前缀和，就可以通过整数分块和记忆化搜索快速求出  $S(n)$

### 复杂度证明

下面假设  $(f * g)(n)$  和  $g(n)$  的前缀和可以  $O(1)$  求出。

若要求出  $S(n)$  需要先求出  $S\left(\frac{n}{d}\right) (d=2 \sim n)$

事实上，有  $\{x \mid \exists d \left( (2 \leq d \leq n) \wedge \left( \frac{n}{d} = x \right) \right)\} \subseteq \{1, 2, 3, \dots, \lfloor \sqrt{n} \rfloor\} \cup \{\lfloor \frac{n}{2} \rfloor, \lfloor \frac{n}{3} \rfloor, \lfloor \frac{n}{4} \rfloor, \dots, \lfloor \frac{n}{\lfloor \sqrt{n} \rfloor} \rfloor\}$

对  $m \in \{x \mid \exists d \left( (2 \leq d \leq n) \wedge \left( \frac{n}{d} = x \right) \right)\}$  有

$$\{1, 2, 3, \dots, \lfloor \sqrt{m} \rfloor\} \cup \{\lfloor \frac{m}{2} \rfloor, \lfloor \frac{m}{3} \rfloor, \lfloor \frac{m}{4} \rfloor, \dots, \lfloor \frac{m}{\lfloor \sqrt{m} \rfloor} \rfloor\} \subseteq \{1, 2, 3, \dots, \lfloor \sqrt{n} \rfloor\} \cup \{\lfloor \frac{n}{2} \rfloor, \lfloor \frac{n}{3} \rfloor, \lfloor \frac{n}{4} \rfloor, \dots, \lfloor \frac{n}{\lfloor \sqrt{n} \rfloor} \rfloor\}$$

因为首先  $m < n$  于是

$$\{1, 2, 3, \dots, \lfloor \sqrt{m} \rfloor\} \subseteq \{1, 2, 3, \dots, \lfloor \sqrt{n} \rfloor\}$$

$$\lfloor \frac{n}{d} \rfloor$$

设  $m = \lfloor \frac{n}{d} \rfloor$  有

$$\lfloor \frac{n}{2d} \rfloor, \lfloor \frac{n}{3d} \rfloor, \lfloor \frac{n}{4d} \rfloor, \dots, \lfloor \frac{n}{\lfloor \sqrt{m} \rfloor d} \rfloor \subset \lfloor \frac{n}{2} \rfloor, \lfloor \frac{n}{3} \rfloor, \lfloor \frac{n}{4} \rfloor, \dots, \lfloor \frac{n}{\lfloor \sqrt{n} \rfloor} \rfloor$$

所以记忆化搜索只需要求出最开始的状态，即  $\{1, 2, 3, \dots, \lfloor \sqrt{n} \rfloor\} \cup \{\lfloor \frac{n}{2} \rfloor, \lfloor \frac{n}{3} \rfloor, \lfloor \frac{n}{4} \rfloor, \dots, \lfloor \frac{n}{\lfloor \sqrt{n} \rfloor} \rfloor\}$

根据整数分块，每个状态统计答案的时间复杂度为  $O(\sqrt{n})$  总时间复杂度为

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

考虑线性筛预处理前  $k$  个前缀和  $(k \geq \sqrt{n})$

总时间复杂度变为

$$O(k) + \sum_{i=1}^{\lfloor \sqrt{\frac{n}{k}} \rfloor} O(\sqrt{\frac{n}{i}}) = O(k) + O(\int_{x=1}^{\sqrt{\frac{n}{k}}} \sqrt{\frac{n}{x}} dx) = O(k) + O(\frac{n}{\sqrt{k}})$$

发现取  $k \sim O(n^{\frac{2}{3}})$  可以达到最佳时间复杂度  $O(n^{\frac{2}{3}})$

## 算法练习

### 习题一

[洛谷p4213](#)

题意

题解

From: <https://wiki.cvbbacm.com/> - CVBB ACM Team

Permanent link: [https://wiki.cvbbacm.com/doku.php?id=2020-2021:teams:legal\\_string:jxm2001:%E6%95%B0%E8%AE%BA\\_3&rev=1594306339](https://wiki.cvbbacm.com/doku.php?id=2020-2021:teams:legal_string:jxm2001:%E6%95%B0%E8%AE%BA_3&rev=1594306339)

Last update: 2020/07/09 22:52