数论 4

Min_25筛

算法简介

一种 $O\left(\frac {n^{\frac 34}}{\log n}\right)$ 计算积性函数 $F(x)$ 前缀和的算法。

算法思路

首先给定 Min_25 筛的适用条件：$F(p)$ 的值可以拆分为若干个完全积性函数，且 $F(p^k)$ 可以快速计算。

设 $mdiv(x)$ 表示 x 的最小素因子。将所有素数从小到大排列，记为 $p_1,p_2,p_3\cdots$

考虑构造完全积性函数 $f(x)$ 满足 $f(p)=F(p)$，设 $g(n,k)=\sum_{i=1}^n[mdiv(i)\gt p_k\text{ or } i\in \text{prime}]f(i)$。

考虑从 $g(n,k-1)$ 转移到 $g(n,k)$，等价于减去 $mdiv(i)=p_k$ 且 $i\not\in \text{prime}$ 的 $f(i)$ 的贡献。如果 $p_k^2\ge n$，则这样的数不存在。

否则将所有满足该条件的 i 提取出一个 p_k，记 $i'=\frac i{p_k}$，于是 $mdiv(i')\ge p_k,f(i')=\frac {f(i)}{f(p_k)}$。

考虑减去 $f(p_k)g(\lfloor \frac n{p_k}\rfloor,k-1)$，发现 $g(\lfloor \frac n{p_k}\rfloor,k-1)$ 多包含了 $\sum_{i=1}^{k-1}f(p_i)$，于是再补上。

于是有状态转移方程

$$ g(n,k) \begin{cases} g(n,k-1), &p_k^2\ge n\\ g(n,k-1)-f(p_k)(g(\lfloor \frac n{p_k}\rfloor,k-1)-\sum_{i=1}^{k-1}f(p_i)), &p_k^2\lt n \end{cases} $$

由于 $\lfloor \frac{\lfloor \frac na\rfloor}{b}\rfloor=\lfloor \frac n{ab}\rfloor$，于是 $g(a,b)$ 中的 a 只有 $O(\sqrt n)$ 个。

使用刷表法暴力转移上述方程直到 $p_{k+1}^2\gt n$，此时得到 $g(a,k)=\sum_{i=1}^a[i\in \text{prime}]f(i)=\sum_{i=1}^a[i\in \text{prime}]F(i),a\in \lfloor \frac nx\rfloor$。

由于玄学因素，时间复杂度为 $O(\frac {n^{\frac 34}}{\log n})$。

接下来设 $S(n,k)=\sum_{i=1}^n[mdiv(i)\gt p_k]F(i)$，于是目标就是求 $S(n,0)+F(1)$。

复杂度证明

算法练习