

搜索课程

网络教室 > CIRNO > 数论和其他数学 > 7.最小素因数和 > 查看



导航



网络教室

个人主页

当前课程

CIRNO

成员

勋章

数论和其他数学



7.最小素因数和

查看

提交

结果

提交历史

报表

相似度

我的课程



系统管理



Programming practice

administration

更改设置

查看

提交

结果

提交历史

测试环境

报表

相似度



添加一个版块

添加...

7.最小素因数和

成绩	0	开启时间	2018年06月28日 星期四 00:05
折扣	0.8	折扣时间	2050年06月28日 星期二 00:05
允许迟交	否	关闭时间	2050年06月28日 星期二 00:05

题目描述

令 $f(x)$ 为 x 的最小素因数，给出两个数 L 和 R ，求 L 到 R 之间所有的 $f(x)$ 的和。

$$ans = \sum_{x=L}^R f(x), \quad f(x) \triangleq \min\{p > 1 | p \mid x\}$$

输入

一组用例一行2个数 L, R

$$2 \leq L \leq 1000, L^2 < R \leq 10^9$$

其实 L 和 R 是可以在 10^{10} 以下任意取的，这里加这些限制条件是为了省掉一些罗嗦的判断。

输出

答案

Manager

1120132001

题目来源

- 局部分配的角色
- 权限
- 检查权限
- 过滤器
- 日志
- 备份
- 恢复

课程管理

切换角色到...

经典问题

题解

这题太难了，直接上题解吧。

要是 $R < 10^7$ ，直接Euler线性筛就很爽。但现在R超了。于是首先应该确定思路：素数和合数应该分别算。

合数这里，最基本的想法是容斥原理。对于每一个根号R以下的素数的集合 $S = \{p_1, \dots, p_k | p_1 < \dots < p_k\}$ ，给答案增加或减去 p_k 乘上L到R之内能被S内每个素数整除的数之个数。 k 是奇数加，偶数减。原理请自行思考。其实素数的和也是可以容斥的，只是比较慢。这个算法的复杂度是R下所有无平方因子数的个数，是个玄学复杂度。但是很遗憾，这个只能搞 10^8 ，再大会TLE。

于是合数的做法参考这篇博客，观察到如果一个数x最小的素因数是p，那么用p去除x商不含p以下的素数。给个阈(读ㄣˊ, yu4)值 $K = 100$ ，对比K小的素数还用容斥做，比K大的素数p，用筛法筛出 $(L - 1)/K$ 到 R/K 的所有不能被p以下素数所整除之数，乘上p更新答案即可。

求素数和有不少很快的算法但是不好写。因为主要复杂度在合数那边，这里写个DP就行。DP的方法也很简单，一开始设 $S[n, 1]$ 为2到n所有整数的和，从2开始逐个素数地筛：

$$S[n, p_i] \leftarrow S[n, p_{i-1}] - pS[n/p_i, p_{i-1}] - S[p_i - 1, p_{i-1}]$$

可以用滚动数组记。注意我们最终求的是 $S(N, p_{\leq \sqrt{N}})$ ，所以S的下标只考虑DP的时候会从N经过的数就可以，也就

$$\text{是 } \frac{N}{1}, \frac{N}{2}, \dots, \frac{N}{\sqrt{N}-1}, \frac{N}{\sqrt{N}}, \sqrt{N}, \sqrt{N}-1, \sqrt{N}-2, \dots, 1$$

最后合并两个答案，注意把多算的素数减掉就行。如果实现够快，可以跑进1s。

	测试输入 ?	期待的输出 ?	时间限制 ?	内存限制 ?	额外进程 ?
测试用例 1	以文本方式显示 1. 2 5↵	以文本方式显示 1. 12↵	1秒	64M	0
测试用例 2	以文本方式显示 1. 2 20↵	以文本方式显示 1. 101↵	1秒	64M	0

