

预览输出

时间：2018 年 7 月 9 日 08:00 ~ 13:00

题目名称	整数	蚯蚓排队	泳池
题目类型	传统型	传统型	传统型
目录	integer	queue	pool
可执行文件名	integer	queue	pool
输入文件名	integer.in	queue.in	pool.in
输出文件名	integer.out	queue.out	pool.out
每个测试点时限	1.0 秒	1.0 秒	1.0 秒
内存限制	512 MB	512 MB	512 MB
测试点数目	10	10	1
每个测试点分值	10	10	100

提交源程序文件名

对于 C++ 语言	integer.cpp	queue.cpp	pool.cpp
对于 C 语言	integer.c	queue.c	pool.c
对于 Pascal 语言	integer.pas	queue.pas	pool.pas

编译选项

对于 C++ 语言	-O2 -std=c++11	-O2 -std=c++14	-O2 -std=c++11
对于 C 语言	-O2 -std=c11	-O2 -std=c14	-O2 -std=c11
对于 Pascal 语言	-O2	-O2	-O2

整数 (integer)

【题目描述】

小 w 想要解决一个有关整数的问题。

给定 n ，他想要知道 $\sum_{k=1}^n \sum_{i=1}^k \sum_{j=1}^k \gcd(i, j, k)$ 对 mod 取模的值。

【输入格式】

从文件 *integer.in* 中读入数据。

输入第一行包含两个正整数 n, mod 。

【输出格式】

输出到文件 *integer.out* 中。

输出一行一个整数，表示答案对 mod 取模的值。

【样例 1 输入】

100 998244353

【样例 1 输出】

463999

【子任务】

对于 20% 的数据， $n \leq 100$ 。

对于 60% 的数据， $n \leq 10^7$ 。

对于 100% 的数据， $n \leq 10^9$ 。

蚯蚓排队 (queue)

【题目描述】

小 h 想要给蚯蚓安家。

蚯蚓的家园是一棵 n 个点的树。点从 1 到 n 标号。一共有 m 条蚯蚓。每条蚯蚓的家在树上的某个点上，蚯蚓的家可以重叠。

但是小 h 有 q 个限制。

这 q 个限制分别为第 a_i 个蚯蚓的家和第 b_i 个蚯蚓的家在树上的任何一条路径都经过了点 c_i 。

小 h 发现他并不会处理这些麻烦的限制，所以他向你求助。数据保证一定有解。

【输入格式】

从文件 *queue.in* 中读入数据。

第一行三个正整数 n, m, q 。

接下来 $n - 1$ 行，每行两个正整数 x, y ，表示树上的一条边。

接下来 q 行，每行三个正整数 a_i, b_i, c_i ，意义如题面所示。

数据保证有解。

【输出格式】

输出到文件 *queue.out* 中。

一行 m 个正整数，其中第 i 个表示第 i 个蚯蚓的家所在的地方。

【样例 1 输入】

2 2 1

1 2

1 2 1

【样例 1 输出】

1 1

【子任务】

对于 20% 的数据, $n, m, q \leq 5$ 。

对于 40% 的数据, $n, m, q \leq 15$ 。

对于 100% 的数据, $n, m \leq 250, q \leq 5 \times 10^4$ 。

泳池 (pool)

【题目描述】

小 x 在搭积木。

她的积木由

【输入格式】

从文件 *pool.in* 中读入数据。

第一行三个正整数 n, m, q 。

接下来 $n - 1$ 行，每行两个正整数 x, y ，表示树上的一条边。

接下来 q 行，每行三个正整数 a_i, b_i, c_i ，意义如题面所示。

数据保证有解。

【输出格式】

输出到文件 *pool.out* 中。

一行 m 个正整数，其中第 i 个表示第 i 个蚯蚓的家所在的地方。

【样例 1 输入】

This is the input of the first sample case.

See this file at `down/1.in`.

【样例 1 输出】

This is the output of the first sample case.

See this file at `down/1.ans`.

【子任务】

对于 20% 的数据， $n, m, q \leq 5$ 。

对于 40% 的数据， $n, m, q \leq 15$ 。

对于 100% 的数据， $n, m \leq 250, q \leq 5 \times 10^4$ 。