**NODX Query**

**Input:** Standard Input, **Output:** Standard Output

**Time Limit:** 1 second(s)

**Memory Limit:** 256 megabytes

**Problem Statement:**

You are an expert in number theory, and so you can solve any problems related to it. Other side, Ohi bhai is the master of data structure. He can easily make any problem difficult by adding some queries and ranges. So, Ohi bhai thinks that you may get troubled if he adds some range queries with any number theory problem. Now he comes with a number theory merged with data structure problem.

You are given Q queries. Each query has three integers L, R, and X. You have to find the number of integers in range L, R (inclusive) having NOD is X.

In number theory, NOD means Number of Divisors. Example:

NOD(4) = 3 as {1 , 2 , 4} are divisors of 4.

NOD(5) = 2 as {1 , 5} are divisors of 5.

NOD(6) = 4 as {1, 2 , 3, 6} are divisors of 6.

**Input:**

The first line contains one integer **Q**  — the number of queries.

Each query consists of threeintegers **L , R** and **X**.

**Constraints:**

**1 <= Q <= 105**

**1 <= L <= R <= 106**

**1 <= X <= 106**

**Output:**

Output one integer for each queries — **the number of integers in range L , R (inclusive) having NOD is X.**

**Sample Input/Output:**

|  |  |
| --- | --- |
| **Sample Input** | **Sample Output** |
| 2  1 10 2  1 10 4 | 4  3 |

Explanation:

1st Case: [2, 3, 5 , 7] total of these 4 have NOD 2 in range 1 , 10

2nd Case: [6, 8, 10] total of these 3 have NOD 4 in range 1 , 10