Problem Maximum Prime Factor

Input data	`stdin`
Output data	`stdout`

Let X be a strictly positive integer and p be its *maximum prime factor*. For $X = 1$, let $p = 1$. We define two types of operations that can be done on X:

Operation 1. X is divided by p, thus becoming X/p.

Operation 2. X is multiplied by a prime number k such that $p \leq k$, thus becoming $X \cdot k$.

Given Q pairs of strictly positive integers (X, Y), determine for each pair the minimum number of operations of either type required to transform X into Y.

Input Data

The input consists of $Q + 1$ lines. The first line contains the value of Q, representing the number of pairs (X, Y). Each of the following Q lines contains two space separated strictly positive integers X Y.

Output Data

Output Q lines, the i-th of which contains a single integer representing the minimum number of operations for the i-th pair.

Restrictions

- $1 \leq Q \leq 1\,000\,000$
- $1 \leq X, Y \leq 4\,000\,000$
- This problem has individual test scoring. See the notice for more details.

#	Points	Restrictions
1	24	$1 \leq X, Y, Q \leq 1\,000$
2	48	$1 \leq X, Y \leq 100\,000$
3	28	No further constraints.

Examples

Input data	Output data
4 4 10 2 9 6 2 12 12	2 3 1 0

Explanations

For (4, 10): 4 becomes 2 using an Operation 1, then becomes 10 using an Operation 2.

For (2, 9): 2 becomes 1 using an Operation 1, then 3 using an Operation 2, then 9 using an Operation 2.

For (6, 2): 6 becomes 2 using Operation 1.

For (12, 12): The numbers are equal, so no operation is required.