Problem H. Roads of the Empire

Input file:	*standard input*
Output file:	*standard output*
Time limit:	2 seconds
Memory limit:	512 mebibytes

King is dead. Long live the king!

The young emperor has just inherited the vast empire of his father. After the successful reign, it consists of such many cities that it is plausible to consider that there are infinitely many of them.

However, the transportation system of the empire is pretty poor. As all new rulers do, the emperor wants to change his predecessor's policies. Thus, instead of war, he decides to build some new roads in the empire. However, the country's religious beliefs require the emperor to follow a specific ritual of building new roads.

First, he has to choose a positive number n. Then, n cities are taken, numbered from 1 to n. After that, for all pairs of cities with numbers x and y such that $1 \le x < y \le n$, the road between them is built if and only if $x + n$ is evenly divisible by y.

After all those manipulations, the emperor has to choose two cities u and v, satisfying $1 \le u, v \le n$, and find the number of roads in the shortest path between them. Moreover, he must choose them randomly and equiprobably to be blessed by God, the holy protector of the empire. Knowing this length, the Religious Council will decide whether the plan is acceptable.

The emperor is worried before the meeting with the Council, so he prepared several such plans. For every plan, answer the question of the length of the shortest path!

Input

The first line contains one integer T ($1 \le T \le 2 \cdot 10^5$) — the number of plans proposed by the emperor.

Each of the following T lines contains three integers n, u, v ($1 \le n \le 10^{18}, 1 \le u, v \le n$) — the number of cities chosen for the plan, and the two cities for which you are asked to find the length of the shortest path in the described graph.

It is guaranteed that u and v were chosen randomly and equiprobably after n had been selected for each test.

Output

For every query, print a number on a separate line — the length of the shortest path between the corresponding cities. If the path does not exist, print -1.

Examples

standard input	standard output
4 5 1 2 8 2 5 7 7 2 6 2 5	1 1 -1 2
1 88 14 2	2