
Bits Reverse

Input file: **standard input**
Output file: **standard output**
Time limit: 1 second
Memory limit: 256 megabytes

Now given two integers x and y , you can reverse every consecutive three bits in arbitrary number's binary form (any leading zero can be taken into account) using one coin. Reversing $(1, 2, 3)$ means changing it into $(3, 2, 1)$.

Could you please find a way that minimize number of coins so that $x = y$? If you can, just output the minimum coins you need to use.

Input

The first line of input file contains only one integer T ($1 \leq T \leq 10000$) indicating number of test cases.

Then there are T lines followed, with each line representing one test case.

For each case, there are two integers x, y ($0 \leq x, y \leq 10^{18}$) described above.

Output

Please output T lines exactly.

For each line, output **Case d :** (d represents the order of the test case) first. Then output the answer in the same line. If there is no way for that, print **-1** instead.

Example

| standard input | standard output |
|----------------|-----------------|
| 3 | Case 1: -1 |
| 0 3 | Case 2: 1 |
| 3 6 | Case 3: 2 |
| 6 9 | |

Note

- Sample 1: Considering following two binary string:

0: 0...0000

3: 0...0011

There is no way to achieve the goal.

- Sample 2: Considering following two binary string:

3: 0...0011

6: 0...0110

You can reverse the lowest three digits in 3 so that 3 is changed into 6.

You just need to perform one reverse so that the minimum coin you need to use is 1.