

# Cipher

Input file:            **standard input**  
Output file:           **standard output**  
Time limit:            2 seconds  
Memory limit:         1024 megabytes

Alice designed a cipher system: it encodes the plaintext as a 64-bit unsigned integer  $x$ , and randomly selects a 64-bit unsigned integer  $a$  as the public key, encrypting the plaintext to get ciphertext  $b = a^x \bmod 2^{64}$ . In particular, it is defined that  $0^0 = 1$ .

Now, Bob has intercepted  $n$  sets of encrypted information  $(a_i, b_i)$  ( $i = 1, 2, \dots, n$ ). To decrypt all the information, Bob needs to find the **smallest** integer  $x_i$  ( $0 \leq x_i < 2^{64}$ ) for each set of encrypted information such that  $a_i^{x_i} \equiv b_i \pmod{2^{64}}$ , or determine that the information has been corrupted (i.e., there is no  $x_i$  that satisfies the condition). Please write a program to help Bob accomplish this task.

## Input

The first line of input contains an integer  $n$  ( $1 \leq n \leq 10^5$ ), representing the number of messages.

The next  $n$  lines, the  $i$ -th line contains two integers  $a_i$  and  $b_i$  ( $0 \leq a_i, b_i < 2^{64}$ ), representing the public key and ciphertext of the  $i$ -th set of information, respectively.

## Output

Output  $n$  lines, for the  $i$ -th line:

- If there exists at least one  $x_i$  that satisfies the conditions stated in the problem, output the smallest one;
- Otherwise, output a line **broken message**.

## Example

standard input	standard output
5	2
2 4	3
3 27	994996658310742016
1000000007 998244353	broken message
4 2	broken message
1000000007 1000000009	