You are playing a single player game where you can convert one integer from another through a sequence of moves. The integer Y is reachable from X in a single move if the following is satisfied.

$$Y = \frac{X \times P_2^k}{P_1^k}$$

where k is a positive integer, P_1 and P_2 are prime numbers and X is divisible P_1^k.

For example 18 is reachable from 8 in one move, because you can divide 8 by 4 and then multiply by 9. But 6 is not reachable from 8. Given two integers A and B calculate the minimum number of moves necessary to transform A into B. Both A and B can be very large. So each of them is needed to be expressed as a multiplication of a sequence of small integers: $A = \prod_{i=1}^{N} a_1$ and $B = \prod_{i=1}^{M} b_1$

Both of the sequences a_i and b_i will be given as inputs.

Input

First line of the input contains T ($1 \leq T \leq 40$) the number of test cases. Then T blocks of test cases follows. First line of the test case contains N ($1 \leq N \leq 300$) followed by N integers. N is the length of the sequence a_i and the following N integers form the sequence a_i. The second line starts with an integer M ($1 \leq M \leq 300$). M is the length of the sequence b_i and the following M integers form the sequence b_i. Each of integers in these two sequences will be between 2 and 200 (inclusive).

Output

For each case of input, print the serial of output followed by an integer: the minimum number of moves required to transform A to B. If it is impossible to transform A to B with any number of moves output '`-1`' instead. If the minimum number of moves is greater than or equal to 20 print a '`-1`' as well.

Sample Input

```
4
1 4
1 9
2 2 2
2 3 3
1 8
1 6
2 32 11
3 27 25 13
```

Sample Output

```
Case 1: 1
Case 2: 1
Case 3: -1
Case 4: 3
```