Problem L. Noblesse Code

Input file:	standard input
Output file:	standard output
Time limit:	10 seconds
Memory limit:	512 megabytes

You will be given n noblesse code pairs $(a_1, b_1), (a_2, b_2), \ldots, (a_n, b_n)$ and q queries. In each query, you will be given a pair (A, B), you need to figure out how many noblesse code pairs can be transformed from the given pair (A, B). Every time you can transform the current pair (A, B) into $(A + B, B)$ or $(A, A + B)$. You can do the transform operation for arbitrary times (or do nothing).

Input

The first line contains a single integer T ($1 \le T \le 100$), the number of test cases. For each test case:

The first line of the input contains two integers n and q ($1 \le n, q \le 50\,000$), denoting the number of noblesse code pairs and the number of queries.

In the next n lines, the i-th line contains two integers a_i and b_i ($1 \le a_i, b_i \le 10^{18}$), describing the i-th noblesse code pair.

In the next q lines, the i-th line contains two integers A and B ($1 \le A, B \le 10^{18}$), describing the pair in the i-th query.

It is guaranteed that the sum of all n is at most 500 000, and the sum of all q is at most 500 000.

Output

For each query, print a single line containing an integer, denoting the number of noblesse code pairs that can be transformed from the given pair. Note that two noblesse code pairs $(a_i, b_i), (a_j, b_j)$ are considered to be different if and only if $i \neq j$.

Example

standard input	standard output
2 3 4 6 9 5 3 1 1 6 3 1 2 2 1 5 3 2 2 7 14 7 14 7 7 7 14	1 0 1 1 2 2