

Problem L. Shika Bika

Program: shikabika.(cpp|java)
Input: shikabika.in
Balloon Color: Yellow

Shika and Bika are two cute little sisters. They like to play together all the time. These days, they are learning about integers. They developed the following game. Shika chooses a certain range, without telling Bika about it, then the game goes in rounds. In each round, Shika calls out one integer from the range she chose, then Bika replies with another integer (it needn't be in the range, just any integer). Shika ends the game only after making sure that she called out each integer in the range at least once. This means Shika can call out an integer more than once and Bika can reply with different integer each time. For example, Shika might call out 3 in a round and Bika replies with 4, then later in another round, Shika calls out 3 again but Bika replies with 3 this time. After every round, they write down a pair consisting of the integer that Shika called out in the round and the integer Bika replied with. The write up has the following problems:

- They do not write the pair in a certain order. So, if Shika says x , and Bika replies with y , they sometimes write this pair as (x, y) and other times as (y, x) .
- In some rounds, they forget to write the pair at all.

Next day after the game, Shika asks Bika questions like: "Did I call out integer q yesterday?". Bika has the write up and she is allowed to answer with one of the following answers. "YES", "NO", or "NOT SURE". Given the written pairs, can you help Bika answer Shika's questions?

Input

Your program will be tested on one or more test cases. The first line of the input will be a single integer T ($1 \leq T \leq 100$).

Each test case represents a game and starts with a line that contains two space separated integers:

- **N**: The number of written pairs ($1 \leq N \leq 1000$)
- **Q**: The number of queries ($1 \leq Q \leq 1000$)

Followed by **N** lines each containing 2 space separated integers **a** and **b** representing the i^{th} pair ($-1,000,000 \leq a, b \leq 1,000,000$)

Followed by **Q** lines each containing a single integer **A** representing the j^{th} query ($-1,000,000 \leq A \leq 1,000,000$)

Output

For each test case, print Q lines, where the i_{th} line answers the i_{th} query with "YES", "NO", or "NOT SURE".

Examples

shikabika.in	stdout
1	NOT SURE
2 3	NOT SURE
10 7	YES
20 14	
3	
8	
11	

Note

In the test case, we can see that we have the following possibilities:

- Round 1: Shika called out 10, Bika replied with 7
 - Round 2: Shika called out 20, Bika replied with 14
- Round 1: Shika called out 10, Bika replied with 7
 - Round 2: Shika called out 14, Bika replied with 20
- Round 1: Shika called out 7, Bika replied with 10
 - Round 2: Shika called out 20, Bika replied with 14
- Round 1: Shika called out 7, Bika replied with 10
 - Round 2: Shika called out 14, Bika replied with 20

For the first query, Bika would never be able to decide if 3 was called out by Shika or not, since some rounds' pairs may be missing. For the second query, in possibilities 1 and 2, Bika can't guarantee that Shika called out 8. For the third query, 11 was for sure called out by Shika, as it is between the two integers called out by Shika in all possibilities.