Task ?. Connect

Along with all the travelling around the country for **IATI 2022**, the tour guide company of Kyusho was once again overwhelmed with calls. After all, not everyone has the luxury of travelling by train.

The area covered by his company consists of **N** cities, numbered 1 through **N**. Between them, there are **M** one-way road segments, numbered 1 through **M**, each connecting a pair of different cities. There may be more than one direct road in a given direction between two cities.

The office receives all kinds of calls from drivers on the road, but one of them requires significantly more time to deal with: "Can you always go from city **A** to city **B** and then go back to city **A** even if one of the road segments is closed?".

Kyusho keeps in mind that you're an experienced programmer, so he asks you to write a program that answers the incoming calls on a given road network.

Input

On the first line of the standard input, the numbers **N** and **M** are given. Each of the next **M** lines contains two numbers, **A** and **B**, specifying that there is a one-way direct road from city **A** to city **B**. On the next line, the number **Q** is given—the number of calls, followed by **Q** lines, each consisting of two numbers **A** and **B,** describing a call from a driver asking about cities **A** and **B**, respectively. Sometimes the calls are so many that Kyusho thinks it's best to get the answers for each unordered pair of cities (**A**, **B**). For convenience, we will denote this case as **Q** = 0. Then, by default the calls (1,1) (1,2) (1,3) … (1,**N**) (2,2) (2 ,3) ... (2,**N**) (3,3) ... (3,**N**) ... (**N**,**N**) must be answered in that order.

Output

To each call we will assign a number:
- 0 → When either road is closed, there is always a route between the two cities in both directions.
- **M**+1 → Even without closing one of the road segments, there isn't a route of the described type.
- The number of the road whose closure will lead to no route in some direction. If there are several such, the answer is the one with the lowest number.

Let the resulting numbers be $s_1, s_2, s_3, \dots s_Q$. On a single line of the standard output, print the remainder of the number $P = s_1 \times B^{Q-1} + s_2 \times B^{Q-2} + s_3 \times B^{Q-3} + \cdots + s_Q \times B^0$ modulo 10^9+7 where $B = 2 \times 10^5$.
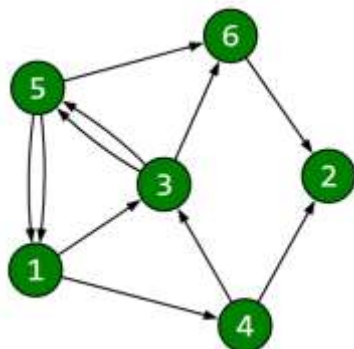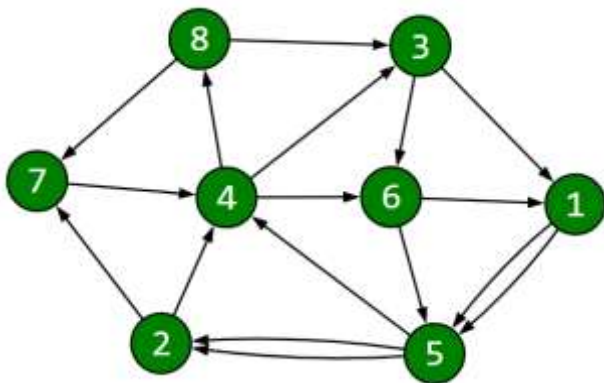
Constraints

➢ $2 \le$ **N** ≤ 2000 , $1 \le$ **A**, **B** $\le$ **N**
➢ $1 \le$ **M** $\le 10^5$
➢ $0 \le$ **Q** $\le 10^5$

Subtasks

Subtask	Points	N	M	Q	Further constraints	Required subtasks
0	0	–	–	–	Sample testcases	–
1	13	≤ 200	≤ 1000	≤ 500 and **Q** ≠ 0	None	–
2	18	≤ 2000	≤ 10^5	≤ 10^5	The answers to all of the calls are either 0 or **M**+1	–
3	11	≤ 500	≤ 8×10^3	≤ 10^5	None	0 – 2
4	12	≤ 2000	≤ 10^5	≤ 10^4 and **Q** ≠ 0	There exists a road segment between cities p and p+1 in both directions for all $1 \le p \le N - 1$	1 – 3
5	25	≤ 2000	≤ 10^5	≤ 10^4 and **Q** ≠ 0	None	1 – 4
6	10	≤ 2000	≤ 8×10^3	≤ 10^5	None	0 – 3
7	11	≤ 2000	≤ 10^5	≤ 10^5	None	0 – 6

The points for a subtask are given only if all the tests for the subtask are passed.

Sample testcases

Input	Output	Explanation
6 11 1 4 4 3 3 5 5 1 1 3 3 6 5 6 6 2 4 2 5 1 3 5 0	575589257	Call for (1,6): There is no route from city 6 to city 1 even without closing one of the road segments. Call for (1,3): Whichever road is closed the routes in both directions will continue to exist. Call for (4,5): If the road segment from city 1 to city 4 is closed, there will be no route from city 5 to city 4. The assigned numbers to the calls: 12 0 12 1 12 0 12 0 0 0 0 12 1 12 0 12 1 0 12 0 12

Input	Output	Explanation
8 17 5 2 6 5 4 6 4 8 4 3 3 1 2 7 7 4 5 4 8 7 8 3 3 6 6 1 2 4 1 5 1 5 5 2 4 4 6 8 1 5 2 6 7	995598902	Second call: If the road segment from city 4 to city 8 is closed, there will be no route from city 1 to city 8. Fourth call: If the road segment from city 7 to city 4 is closed, there will be no route from city 7 to city 6. The assigned numbers to the calls: 18 4 18 8