Game Ranking

Time limit: 1 sec

Everyone love game, so are the students of the algorithm class. We have **N** students in the class (numbered 0 to **N**-1). When bored, they find a one-on-one game and play it several times. To increase the fun, they set up a playing season where students compete with each other and the result of every game is recorded. The result of a game between student A and student B is either "A win" or "A lose". There is no draw. Moreover, since the time is limited, it is possible that some pair of students might have no chance to play a game.

From the result of all games, we wish to create a ranking system. The ranking can be described as follows.

- The rank is numbered from 1 onward. Rank 1 is the best players.
- Each student should have the best rank they can have that does not violate the following rules
- It is possible that two or more student might have the same rank.
- If the result of the game between A and B is "A win", the rank of A must be better than the rank of B. However, there is some exception as follows.
  - If A won against B, we consider that A "indirectly won" against B and everyone that lost to B.
  - If A "indirectly win" against B and at the same time B "indirectly win" against A, we consider A and B to have the same rank. For example, if we have a record that A win against B and B win against C and C win against A, we consider A,B,C to have the same rank.

Your task is to calculate the best ranking of every student.

Input

- The first line of input contains two integers **N** and **P** which indicate the number of students and the number of game played (1 ≤ **N** ≤ 5,000 และ 0 ≤ **P** ≤ 20,000).

- The following P lines indicate the recorded results. Each line contains two integers A and B indicating that A won against B. It is guaranteed that there is no repetitive line.

Output

The output contains K integers where K is the total number of distinct rank. The first number indicates the number of student that have rank 1, the second number indicates the number of students that have rank 2 and so on until the Kth number that indicates the number of student that have rank K.

Example

Input	Output
10 0	10
10 2 5 6 0 1	8 2
4 3 0 1 1 2 2 3	1 1 1 1
4 5 0 1 1 2 2 0 2 3 1 3	3 1
3 2 0 1 1 0	3
4 3 2 1 2 0 1 3	1 2 1