

## Problem A. Alternating Paths

Input file:            standard input  
Output file:          standard output  
Time limit:           3 seconds  
Memory limit:        256 megabytes

Next year, the Overly Complicated Problem Colloquium will take place in the Greatly Magnificent City. In order to impress the guests, several construction projects are underway. This includes painting all the city streets.

As one might expect, the city consists of  $n$  junctions and  $m$  bidirectional streets connecting them. The junctions are indexed  $1 \dots n$ . No pair of junctions is connected by more than one street, no street connects a junction to itself and it is possible to walk from any junction to any other using these streets.

Each street should be painted either red or blue. The mayor thinks that it's a lot more interesting to walk around the city if every street is different from the last. Therefore, the mayor has issued an additional constraint: "if  $p$  and  $q$  are different junctions, then it must be possible to walk from  $p$  to  $q$  such that every street on the path is painted in a different color from the street before it." Such a path may also visit some streets or junctions multiple times.

Your task is to suggest a way to paint all the streets so that the mayor is satisfied or to claim that this isn't possible.

### Input

The first line contains one integer  $t$  ( $1 \leq t \leq 1000$ ) — the number of test cases.  $t$  test cases follow. Each test case is described as follows.

The first line of the test case contains two integers  $n$  and  $m$  — the number of junctions and streets, respectively ( $2 \leq n \leq 100$ ,  $1 \leq m \leq 300$ ). Each of the following  $m$  lines consists of two integers  $u$  and  $v$  ( $1 \leq u \leq n$ ,  $1 \leq v \leq n$ ), denoting a street between junctions  $u$  and  $v$ .

It is guaranteed that in each input file, there are at most 100 test cases where  $n > 50$  or  $m > 150$ .

### Output

For each test case, print the answer on a separate line as follows:

- If satisfying the mayor isn't possible, print **IMPOSSIBLE**.
- Otherwise, print a string of length  $m$ . The  $i$ -th character in the string should be **B** if the  $i$ -th road is blue, and **R** otherwise.

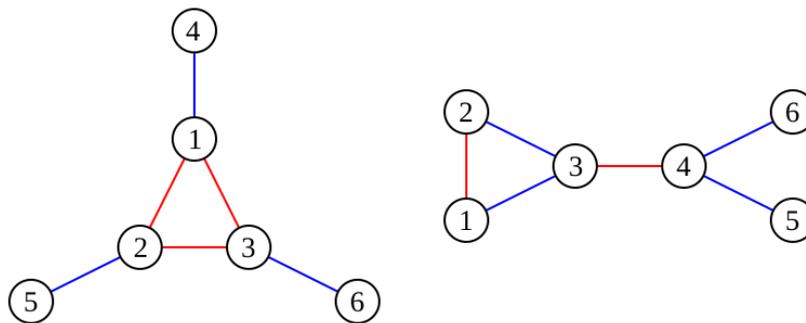
## Example

standard input	standard output
3	RRRBBB
6 6	RBBRBB
1 2	IMPOSSIBLE
2 3	
3 1	
4 1	
5 2	
6 3	
6 6	
1 2	
2 3	
3 1	
3 4	
4 5	
4 6	
4 3	
1 2	
4 2	
2 3	

## Note

The extra line breaks between test cases in the example are to aid with reading, they are not present in the actual input.

The following graphs illustrate a possible coloring for the first two test cases.



In the third test case, however you paint the edges, there will always be a pair of leaves with the same color leading into them, making the condition impossible to satisfy.