

1147 Crime

The city of X (for reasons that will become clear, the name of this city is kept secret) is one of the safest cities in the country. However, the mayor of X has decided to eliminate crime completely.

The Perfect Police Patrol System (PPPS)

The Perfect Police Patrol System is the mayor's response to crime. According to the mayor's analysis, the city consists of intersections and streets, where every street connect two intersections. Crime must be eliminated from the streets.

The mayor's idea is to install PPPS stations at some intersections. The officers in a station can patrol all the streets that enter the intersection where the station is, and only those. To achieve maximal security every street should be patrolled. That is, *there must be a PPPS station at one of the two ends of every street*. However, in the past, there were unfortunate incidents where two police patrols fought with each other in dark streets, because they could not recognize that they were both from the police. To avoid such untimely incidents, *there cannot be PPPS stations at both ends of a street*. Finally, as PPPS stations are very expensive and as the mayor is not willing to increase taxes, *one should build as few PPPS stations as possible*.

Write a program that determines whether it is possible to design such a Perfect Police Patrol System. If so, the program should output the minimal number of stations that have to be built.

Input

The input contains several test cases, each of them consists of lines of integers. The first line contains two integers n and m ($1 \leq n \leq 1000$, $1 \leq m \leq 10000$) where integer n is the number of intersections and integer m is the number of streets. The next m lines describe the m streets. Each of these lines contains two integers that identify the two intersections at the two ends of the streets (intersections are numbered $1, 2, \dots, n$). Because of the various tunnels, bridges, temporarily closed streets etc. you cannot assume anything about the topology of the city.

Output

For each test case, the output contains one single line. If the PPPS can be designed, then the program should output one integer, the minimal number of PPPS stations required. Otherwise, if no PPPS can be designed, the program should output the line 'Impossible' (without quotes).

Sample Input

```
7 4
1 2
1 3
4 5
6 5
5 6
1 2
1 3
2 4
3 4
3 5
4 5
```

Sample Output

2

Impossible