The International Corporation for Protection and Control (ICPC) develops efficient technology for, well, protection and control. Naturally, they are keen to have their own headquarters protected and controlled. Viewed from above, the headquarters building has the shape of a convex polygon. There are several suitable places around it where cameras can be installed to monitor the building. Each camera covers a certain range of the polygon sides (building walls), depending on its position. ICPC wants to minimize the number of cameras needed to cover the whole building.

Input

The input file contains several test cases, each of them as described below.

Its first line contains two integers n and k ($3 \leq n \leq 10^6$ and $1 \leq k \leq 10^6$), where n is the number of walls and k is the number of possible places for installing cameras. Each of the remaining k lines contains two integers a_i and b_i ($1 \leq a_i, b_i \leq n$). These integers specify which walls a camera at the i-th place would cover. If $a_i \leq b_i$ then the camera covers each wall j such that $a_i \leq j \leq b_i$. If $a_i > b_i$ then the camera covers each wall j such that $a_i \leq j \leq n$ or $1 \leq j \leq b_i$.

Output

For each test case, the output must follow the description below, on a line by itself.

Display the minimal number of cameras that suffice to cover each wall of the building. The ranges covered by two cameras may overlap. If the building cannot be covered, display '`impossible`' instead.

Sample Input

```
100 7
1 50
50 70
70 90
90 40
20 60
60 80
80 20
8 2
8 3
5 7
8 2
8 4
5 7
```

Sample Output

```
3
impossible
2
```