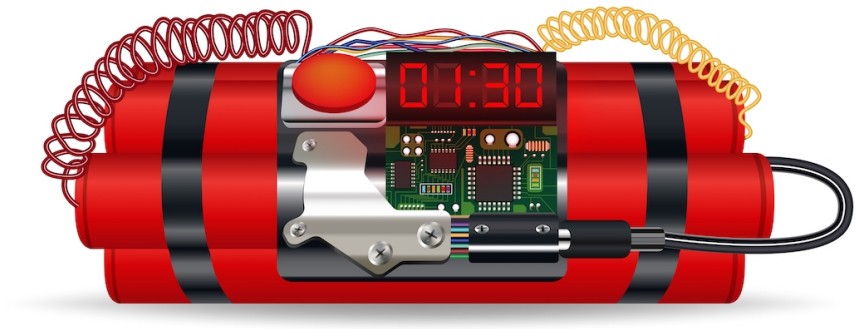


Statement

You are called urgently in the middle of the night, a particularly complex bomb has been found and you are the only one who can defuse it! You arrive on site and discover the machine: several inputs are connected to a single output by a series of logical gates.

All inputs are initially in state 1, and the state of the output depends on the logical circuit. If you cut an input wire, the value of this input becomes 0. To defuse the bomb, you need to determine which input wire to cut for the output of the last gate to become 0. **You can only cut one wire** and your goal is to determine all possible solutions to defuse the bomb by cutting only one input wire.

To understand how logical gates work, you can see this table: https://en.wikipedia.org/wiki/Logic_gate#Symbols

Data format

Input

Row 1: two integers separated by a space, **N** between 1 and 500 000 representing the number of inputs in of the logical circuit of the bomb, and **M** between 1 and 500 000, representing the number of logical gates in the circuit.
Rows 2 to **M** - 1: the description of a logical circuit gate, in the type format *input1 (entrance2*):

- The possible types of gates are: NOT, AND, OR, XOR
- NOT gates take only one entrance, gates of all other types take two entrances
- The entrances to each gate can be an input (noted *I*...), or an output from another gate (noted *G*...).

The gates are numbered from *G0* to *G-<**M**-1>*, the number corresponds to the position of the gate in the list of descriptions. The inputs are numbered from *I0* to *I< **N**-1>.*

For example *XOR I1 G3* represents an XOR gate whose input I1 and the output of the fourth gate in the list

Output

Your code will have to display on a row the input identifiers which by cutting only their wire will lead output of the last gate to 0, in ascending order and separated by spaces (e.g. *I0 I4 I6 I7 I18).* Every input of your solution must be enough to defuse the bomb: so in the previous example, your solution means that if I only cut the wire of the Input 18 or the wire of input 4 (or of any input mentioned in the solution) then the exit of the last gate will be 0.
If there is no input wire to defuse the bomb, return IMPOSSIBLE.

Comments

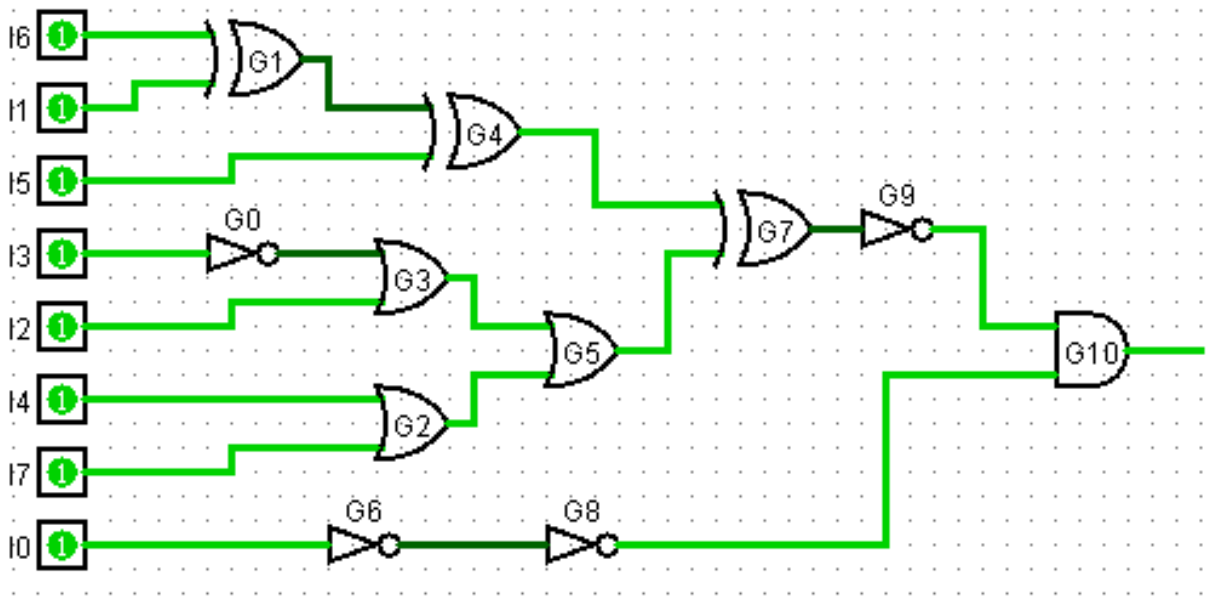
- The logical gates are given in a topological order: a gate never enters the exit of a logical gate that is after itself in the list
- Every output from a logical gate and each input wire is used **exactly once** as the input of another gate, with the exception of the output of the last gate that corresponds to the bomb.
- It is possible that the bomb is already defused in its original state. In this case, you must list all the inputs for which cutting the wire will leave the output at 0.

Example

Example of a network

```
8 11
NOT I3
XOR I6 I1
OR I4 I7
OR G0 I2
XOR G1 I5
OR G3 G2
NOT I0
XOR G5 G4
NOT G6
NOT G7
AND G9 G8
```

Matches the circuit



Example 1 input and solution

For input:

```
3 3
AND I0 I1
NOT I2
OR G0 G1
```

When all inputs are at 1, the output is also at 1.

If I0 or I1 goes to 0, the output of G0 will be 0. G2 will be 0 OR 0, so the output will move to 0.
If I2 goes to 0, the output of G0 being 1, G2 will be 1 OR 0. The output remains in this case at 1.

So for this example your code will have to return:

```
I0 I1
```

Example 2 input and solution

For entry:

```
2 2
AND I0 I1
NOT G0
```

No matter which wire you cut, the bomb's output will go from 0 to 1. For this example, your code will have to return

```
Impossible
```

Example 3 input and solution

The following entry corresponds to the logical circuit outlined above in the statement:

```
8 11
NOT I3
XOR I6 I1
OR I4 I7
OR G0 I2
XOR G1 I5
OR G3 G2
NOT I0
XOR G5 G4
NOT G6
NOT G7
AND G9 G8
```

The expected response is:

```
I0 I1 I5 I6
```