

Énoncé

On vous appelle en urgence au milieu de la nuit, une bombe particulièrement complexe a été trouvée et vous êtes le seul à pouvoir la désamorcer ! Vous arrivez sur place et découvrez l'engin : plusieurs fils d'entrée (inputs) sont reliés à une unique sortie par une série de portes logiques.

Tous les inputs sont initialement à l'état 1, et l'état de la sortie dépend du circuit logique. Si vous coupez un fil d'input, la valeur de cet input passe à 0. Pour désamorcer la bombe, vous devez déterminer quel fil d'input couper pour que la sortie de la dernière porte passe à 0. **On ne peut couper qu'un seul fil** et votre objectif est de déterminer toutes les solutions possibles pour désamorcer la bombe en ne coupant qu'un seul fil d'input.

Pour comprendre comment fonctionnent les portes logiques, vous pouvez consulter ce tableau : https://fr.wikipedia.org/wiki/Fonction_logique#Représentation_graphique

Format des données

Entrée

Ligne 1 : deux entiers séparés par un espace, **N** compris entre 1 et 500 000 représentant le nombre de fils en entrée du circuit logique de la bombe, et **M** compris entre 1 et 500 000, représentant le nombre de portes logiques dans le circuit.

Lignes 2 à **M** + 1 : la description d'une porte logique du circuit, au format type entrée1 (entrée2)- Les types de portes possibles sont : NOT, AND, OR, XOR

- Les portes NOT prennent une seule entrée, les portes de tous les autres types prennent deux entrées
- Les entrées de chaque porte peuvent être un input (notés I...), ou une sortie d'une autre porte (notés G...).

Les portes sont numérotées de G_0 à G_{M-1} , le numéro correspond à la position de la porte dans la liste des descriptions. Les inputs sont numérotés de I_0 à I_{N-1} .

Par exemple *XOR I1 G3* représente une porte XOR dont les entrées sont l'input I1 et la sortie de la quatrième porte dans la liste

Sortie

Votre code devra afficher sur une ligne les identifiants des Inputs dont il suffit de couper le fil pour que la sortie de la dernière porte soit à 0, par ordre croissant et séparés par des espaces (par exemple *I0 I4 I6 I7 I18*). Chaque input de votre solution doit suffire pour désamorcer la bombe : donc dans l'exemple précédent, votre solution signifie que si je ne coupe que le fil de l'Input 18 ou que le fil de l'input 4 (ou que le fil de n'importe quel input mentionné dans la solution) alors la sortie de la dernière porte sera 0.

Si aucun fil d'input ne permet de désamorcer la bombe, renvoyez IMPOSSIBLE.

Commentaires

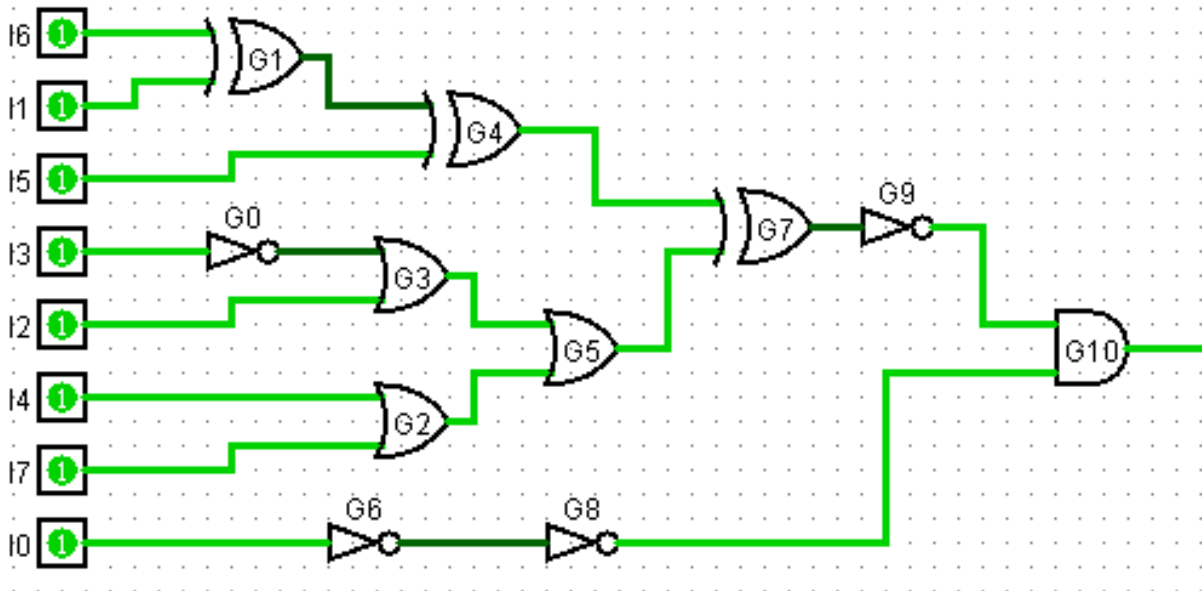
- Les portes logiques sont données par ordre topologique : une porte ne prend jamais en entrée la sortie d'une porte logique qui se situe après elle-même dans la liste
- Chaque sortie d'une porte logique et chaque fil d'input sont utilisés **exactement une fois** en entrée d'une autre porte, à l'exception de la sortie de la dernière porte qui correspond à la sortie de la bombe.
- Il est possible que la bombe soit déjà désamorcée dans son état initial. Dans ce cas, vous devez lister tous les inputs dont couper le fil permettra de laisser la sortie à 0.

Exemple

Exemple de réseau

```
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
```

Correspond au circuit



Exemple 1 d'entrée et de solution

Pour l'entrée :

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

Lorsque que tous les inputs sont à 1, la sortie est également à 1.

Si I0 ou I1 passe à 0, la sortie de G0 sera 0. G2 sera donc 0 OR 0, et la sortie passera donc à 0.

Si I2 passe à 0, la sortie de G0 étant 1, G2 sera donc 1 OR 0. La sortie reste dans ce cas à 1.

Ainsi pour cet exemple votre code devra renvoyer :

```
I0 I1
```

Exemple 2 d'entrée et de solution

Pour l'entrée :

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

Quel que soit le fil que l'on coupe, la sortie de la bombe passera de 0 à 1. Pour cet exemple, votre code devra donc renvoyer

```
IMPOSSIBLE
```

Exemple 3 d'entrée et de solution

L'entrée suivante correspond au circuit logique schématisé plus haut dans l'énoncé :

```
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
```

La réponse attendue est :

```
I0 I1 I5 I6
```