

A. Ir e vir

Time limit: 0.237s

Memory limit: 1536 MB

Numa certa cidade há N intersecções ligadas por ruas de mão única e ruas com mão dupla de direção. É uma cidade moderna, de forma que muitas ruas atravessam túneis ou têm viadutos. Evidentemente é necessário que se possa viajar entre quaisquer duas intersecções, isto é, dadas duas intersecções V e W , deve ser possível viajar de V para W e de W para V .

Sua tarefa é escrever um programa que leia a descrição do sistema de tráfego de uma cidade e determine se o requisito de conexidade é satisfeito ou não.

Entrada

A entrada contém vários casos de teste. A primeira linha de um caso de teste contém dois números inteiros N e M , separados por um espaço em branco, indicando respectivamente o número de intersecções ($2 \leq N \leq 2000$) e o número de ruas ($2 \leq M \leq N(N-1)/2$). O caso de teste tem ainda mais M linhas, que contêm, cada uma, uma descrição de cada uma das M ruas. A descrição consiste de três inteiros V , W e P , separados por um espaço em branco, onde V e W são identificadores distintos de intersecções ($1 \leq V, W \leq N, V \neq W$) e P pode ser 1 ou 2; se $P = 1$ então a rua é de mão única, e vai de V para W ; se $P = 2$ então a rua é de mão dupla, liga V e W . Não existe duas ruas ligando as mesmas intersecções.

O ultimo caso de teste é seguido por uma linha que contém apenas dois números zero separados por um espaço em branco.

Saída

Para cada caso de teste seu programa deve imprimir uma linha contendo um inteiro G , onde G é igual a 1 se o requisito de conexidade está satisfeito, ou G é igual a 0, caso contrário.

Exemplo

Entrada:

```
4 5
1 2 1
1 3 2
2 4 1
3 4 1
4 1 2
3 2
1 2 2
1 3 2
3 2
1 2 2
1 3 1
4 2
1 2 2
3 4 2
0
```

Saída:

```
1
1
0
0
```

Primeira fase da Maratona de Programação - 2010