

## Problema 1

### Pontos vulneráveis

Tempo limite: 0,25s (C/C++), 1,0s(Java)

Memória limite: 100 MB

Uma empresa de comunicação possui uma rede de comunicação organizada de forma que pontos são distribuídos geograficamente para atender diferentes regiões. As conexões entre diferentes pontos é essencial para que a troca de informação seja efetivada entre diferentes partes da rede.

Um problema que a rede enfrenta é que, quando uma falha ocorre em alguns pontos vulneráveis, isso ocasiona a falta de conectividade de outros pontos ao restante da rede, pois o ponto onde a falha ocorreu é o único ponto por onde informações de um grupo de pontos precisam passar para alcançar pontos fora desse grupo. Por exemplo, na rede ilustrada na figura abaixo, uma falha no ponto 2 causaria a desconexão do ponto 5 com os outros pontos da rede, portanto, o ponto 2 é o ponto vulnerável desta rede.

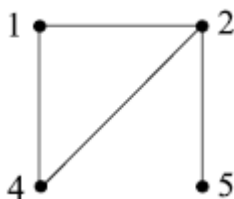


Figura: exemplo de um rede de comunicação.

Essa empresa precisa identificar vulnerabilidades em sua grande rede de comunicação. A sua tarefa é escrever um programa que, dada uma descrição de como os pontos se comunicam em uma dada rede de comunicação, identifique e liste todos os pontos da rede considerados vulneráveis, ou seja, pontos onde o acontecimento de falhas impossibilite a comunicação de outros pontos com partes da rede.

#### Entrada

Uma entrada para o programa descreverá uma dada rede de comunicação. A primeira linha contém dois inteiros  $N$  ( $1 \leq N \leq 100.000$ ) e  $M$  ( $0 \leq M \leq 100.000$ ), representando o número de pontos e o número de conexões entre pares de pontos, respectivamente. Cada ponto é representado por número distinto entre 0 e  $N-1$ . Nas próximas  $M$  linhas, são apresentadas os pares de pontos  $a$   $b$  diretamente ligados na rede.

#### Saída

Seu programa deve imprimir todos os pontos vulneráveis da rede em ordem crescente de identificador, sendo um ponto por linha.

Exemplo de Entrada	Exemplo de Saída
4 4  0 1  0 2  1 2  2 3	2

Exemplo de Entrada	Exemplo de Saída
13 17  0 1  1 2  2 3  3 4  4 1  0 5  0 12  5 12  5 6  5 7  7 8  7 9  8 9  9 10	0  1  5  9

9 11	
9 12	
10 11	