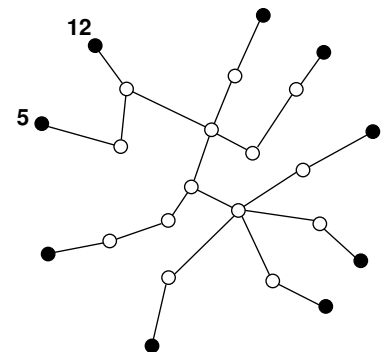


# Capitais

Nome do arquivo: `capitais.c`, `capitais.cpp`, `capitais.pas`, `capitais.java`, `capitais.js` ou `capitais.py`

A Linearlândia construiu uma rede de ferrovias de alta velocidade, ligando certos pares de cidades, de modo que: é possível viajar entre qualquer par de cidades usando apenas ferrovias; e há apenas um caminho de ferrovias (sequência de ferrovias) entre qualquer par de cidades. Existe muita disputa entre as capitais dos estados da Linearlândia e, por isso, ficou decidido que cada capital seria ligada por ferrovia a apenas uma outra cidade, e que toda cidade que não é capital seria ligada a outras cidades por duas ou mais ferrovias. Dessa forma, nenhuma viagem entre um par de capitais usando apenas ferrovias passa por uma terceira capital.



Vamos definir como *distância-ferrovia* entre duas cidades o número de ferrovias que é necessário utilizar para viajar entre essas duas cidades. Dada apenas a informação sobre quais pares de cidades estão ligados por uma ferrovia, você deve escrever um programa para computar a menor distância-ferrovia entre todos os pares de capitais. Na figura acima, há nove capitais e a menor distância-ferrovia entre qualquer par de capitais é 3, entre as capitais 5 e 12.

## Entrada

A primeira linha da entrada contém um inteiro  $N$ , o número de cidades. As cidades são identificadas por inteiros de 1 a  $N$ . As  $N - 1$  linhas seguintes contém, cada uma, dois inteiros  $U$  e  $V$ , representando um par de cidades ligadas por uma ferrovia.

## Saída

Seu programa deve produzir uma única linha, contendo um único inteiro, a menor distância-ferrovia entre todos os pares capitais.

## Restrições

- $2 \leq N \leq 10^5$

## Informações sobre a pontuação

- Em um conjunto de casos de teste cuja soma é 40 pontos,  $N \leq 10^4$ ;

## Exemplos

<b>Entrada</b> 8 1 2 2 3 3 4 5 3 6 5 6 7 8 7	<b>Saída</b> 3
<b>Entrada</b> 2 1 2	<b>Saída</b> 1