

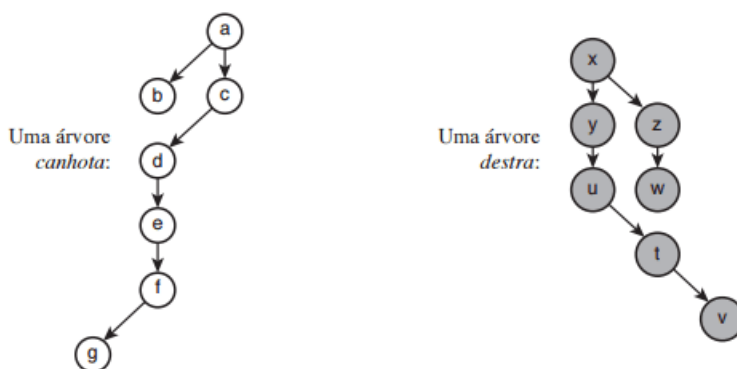
# Fundindo Árvores

Por Maratona de Programação da SBC – 2016  Brazil

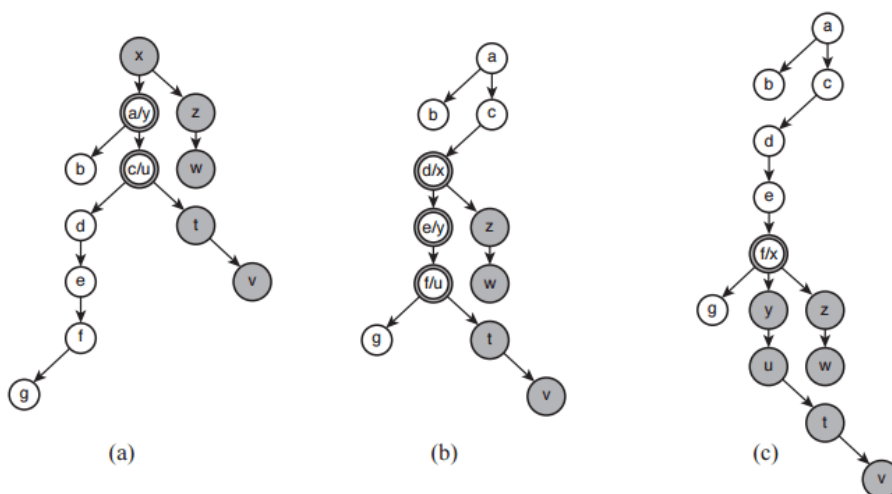
**Timelimit: 1**

Em Computação árvores são objetos estranhos: a raiz está no topo e as folhas estão embaixo! Uma árvore é uma estrutura de dados composta de **N** vértices conectados por **N-1** arestas de forma que é possível chegar de um vértice a qualquer outro vértice seguindo as arestas. Em uma árvore enraizada, cada aresta conecta um vértice pai a um vértice filho. Um único vértice não tem pai, e é chamado de raiz. Assim, partir da raiz é possível chegar a qualquer outro vértice da árvore seguindo as arestas na direção de pai para filho.

Em uma árvore ternária cada vértice pode ter até três vértices filhos, chamados esquerdo, central e direito. Uma árvore ternária canhota é uma árvore ternária enraizada em que nenhum vértice tem filho direito. Uma árvore ternária destra é uma árvore ternária enraizada em que nenhum vértice tem filho esquerdo. A raiz de uma árvore ternária é sempre um vértice central. A figura abaixo mostra exemplos de uma árvore canhota e de uma árvore destra.



Em uma árvore ternária cada vértice pode ter até três vértices filhos, chamados esquerdo, central e direito. Uma árvore ternária canhota é uma árvore ternária enraizada em que nenhum vértice tem filho direito. Uma árvore ternária destra é uma árvore ternária enraizada em que nenhum vértice tem filho esquerdo. A raiz de uma árvore ternária é sempre um vértice central. A figura abaixo mostra exemplos de uma árvore canhota e de uma árvore destra.



Note que na Figura (a) a raiz é o vértice  $x$  (da árvore destra) e os pares de vértices  $(a, y)$  e  $(c, u)$  são superpostos. Na Figura (b) a raiz é o vértice  $a$  (da árvore canhota) e os pares de vértices  $(d, x)$ ,  $(e, y)$  e  $(f, u)$  são superpostos. Na Figura (c) a raiz também é o vértice  $a$  (da árvore canhota) e o par de vértices  $(f, x)$

é superposto.

Dadas uma árvore canhota e uma árvore destra, sua tarefa é determinar o número mínimo de vértices necessários para construir uma árvore ternária que é uma superposição das árvores dadas.

### Entrada

A primeira linha de um caso de teste contém um inteiro **N** indicando o número de vértices da árvore canhota ( $1 \leq N \leq 10^4$ ). Vértices nesta árvore são identificados por números de 1 a **N**, e a raiz é o vértice de número 1. Cada uma das **N** linhas seguintes contém três inteiros **I**, **L** e **K**, indicando respectivamente o identificador de um vértice **I**, o identificador do filho esquerdo **L** de **I** e o identificador do filho central **K** de **I** ( $0 \leq I, L, K \leq N$ ). A linha seguinte contém um inteiro **M** indicando o número de vértices da árvore destra ( $1 \leq M \leq 10^4$ ). Vértices nesta árvore são identificados por números de 1 a **M**, e a raiz é o vértice de número 1. Cada uma das **M** linhas seguintes contém três inteiros **P**, **Q** e **R**, indicando respectivamente o identificador de um vértice **P**, o identificador do filho central **Q** de **P** e o identificador do filho direito **R** de **P** ( $0 \leq P, Q, R \leq N$ ). O valor zero indica um vértice não existente (usado quando um vértice não tem um ou ambos os seus filhos).

### Saída

Imprima o número mínimo de vértices de uma árvore que é a superposição das duas árvores dadas na entrada.

Exemplos de Entrada	Exemplos de Saída
7 1 2 3 2 0 0 3 4 0 4 0 5 5 0 6 6 7 0 7 0 0 7 1 2 3 2 4 0 3 5 0 4 0 6 5 0 0 6 0 7 7 0 0	11
5 1 2 3 2 4 5 3 0 0 4 0 0 5 0 0 3 1 2 3 2 0 0 3 0 0	6

3	3
3 0 2	
2 0 0	
1 0 3	
2	
2 0 0	
1 2 0	