

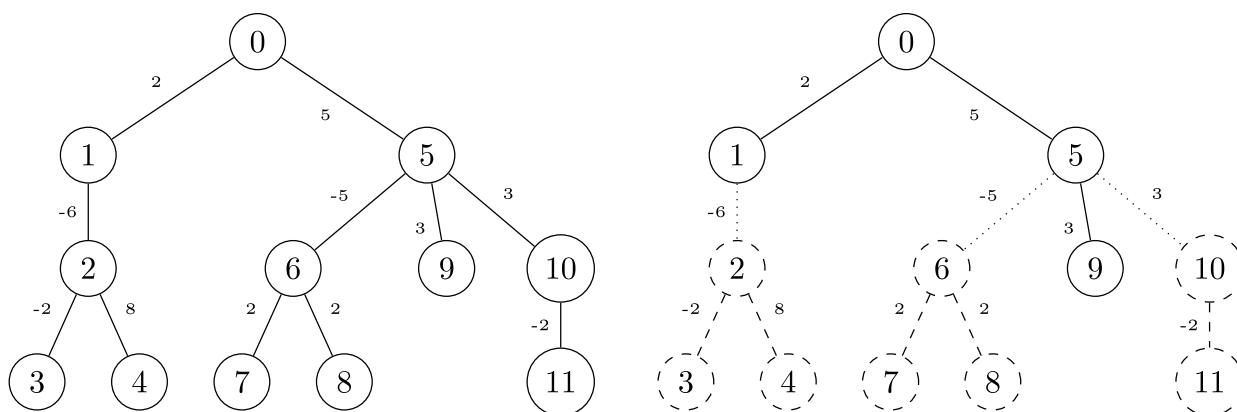
## Problema A. Árvore de Natal

Arquivo-fonte: `arvore.c`, `arvore.cpp` ou `arvore.java`  
 Tempo limite: 3 segundos  
 Autor: Lucas Maciel

Roberto é apaixonado por árvores de natal. Todo ano, ele enfeita as árvores de Itajubá com os mais bonitos enfeites. Por isso, esse ano ele foi convidado pelo prefeito de Belo Horizonte a enfeitar as árvores de natal que serão colocadas no final do ano. Para isso, Roberto irá podar galhos das árvores que ele acha que estão feios. Para cada galho  $i$  da árvore, Roberto definiu o quão bonito ele é por um número  $w_i$ . Valores negativos representam que o galho é feio. A beleza de uma árvore de natal é então definida pela soma das belezas de seus galhos.

O processo de poda de uma árvore é simples. Roberto corta um galho da árvore, e todos os galhos que partiam dele caem também. Na figura 1 temos um exemplo de árvore e de uma poda feita por Roberto. Em cada galho da árvore na figura temos o valor  $w_i$  que descreve o quão bonito é o galho. Na árvore à direita, os galhos cortados por Roberto estão pontilhados, enquanto os nós e galhos que caíram após o corte estão tracejados. No exemplo da figura 1, tanto a árvore sem corte, quanto a cortada possuem um valor 10 de beleza.

Figura 1: Corte de Roberto



Sabendo da VI Maratona Mineira de Programação, Roberto pediu que você fizesse um programa que o ajudasse a definir quais galhos ele deverá cortar da árvore para que ela fique a mais bonita possível de acordo com sua definição.

### Entrada

A entrada consiste de um inteiro  $N$  que é a quantidade de nós da árvore. Nas próximas  $N - 1$  linhas, temos quatro inteiros  $d_i$ ,  $a_i$ ,  $b_i$  e  $w_i$  representando o identificador do galho  $i$ , que ele conecta o nó  $a_i$  ao nó  $b_i$ , e que ele possui  $w_i$  de beleza pela classificação de Roberto. A árvore é sempre enraizada no nó 0. É garantido que o grafo da entrada é conexo e não possui ciclos.

### Saída

A saída deve conter dois inteiros,  $D$  e  $M$  que representam o quão bonito é a árvore com o corte ótimo e quantos cortes precisam ser feitos, respectivamente. Se o número de cortes for maior do que 0, na próxima linha, imprima  $M$  inteiros  $d_j$  em ordem crescente e separados por espaço, onde  $d_j$  representa o identificador de cada galho  $j$  a ser cortado.

Caso exista mais de uma árvore com o mesmo grau de beleza, imprima aquela que possui menos galhos. Se ainda existir mais de uma árvore que satisfaça os mesmos critérios, imprima a que possua menos cortes.

### Limites

$$2 \leq N \leq 10^6$$

$$0 \leq d_i \leq N - 2$$

$$0 \leq a_i, b_i \leq N - 1$$

$$-1000 \leq w_i \leq 1000$$

## Exemplos

Entrada	Saída
12 0 0 1 2 1 1 2 -6 2 2 3 -2 3 4 2 8 4 0 5 5 5 5 6 -5 6 6 7 2 7 8 6 2 8 5 9 3 9 5 10 3 10 10 11 -2	15 3 2 5 10

A figura 2 apresenta o corte ótimo do caso de exemplo. Note que foram feitos o menor número de cortes para obter essa árvore de beleza 15.

Figura 2: Corte ótimo

