

## Problem K. Adrian the Wonder Child

Input file: *standard input*  
Output file: *standard output*  
Time limit: 2 seconds  
Memory limit: 256 mebibytes

Adrian the Wonder Child started coding for his new project a while ago. Each day he pushed exactly one commit on his Git repository, and now it looks like a tree with  $n$  vertices.

As we all know, Adi has good and bad days (usually one good day followed by ten bad days, but that fact is less important for this problem). So for every node of the tree except the root (initial commit), he labeled **the edge to the parent** with either 0 or 1 depending on whether he pushed the corresponding commit on a bad day or on a good day.

A path between two nodes in the tree is considered *k-alternating* if it has at most  $k$  consecutive edges with the same value.

Looking back on what he did in a particular day, Adi can change his mind on whether it was good or bad; thus, he can flip the label of the edge between the corresponding node and its parent.

Given integers  $m$  and  $k$ , Adi asks himself what is the longest  $k$ -alternating path in the tree that can be obtained by flipping the labels of at most  $m$  edges.

### Input

The first line of the input contains three integers:  $n$ ,  $k$ , and  $m$  ( $3 \leq n \leq 2 \cdot 10^5$ ,  $2 \leq k < n$  and  $0 \leq m < n$ ).

Each of the following  $n - 1$  lines contains three integers:  $x$ ,  $y$ , and  $c$  ( $1 \leq x, y \leq n$  and  $c \in \{0, 1\}$ ) meaning that there is an edge between nodes  $x$  and  $y$  with label  $c$ .

It is guaranteed that the given edges form a tree.

### Output

Output a single number representing the size of the longest  $k$ -alternating path obtained by flipping the label of at most  $m$  edges.

### Example

<i>standard input</i>	<i>standard output</i>
9 3 2 1 2 0 2 3 1 3 4 1 4 5 1 1 6 1 1 7 0 7 8 0 8 9 1	7