

Graffiti

Input file: **standard input**
Output file: **standard output**
Time limit: **2 seconds**
Memory limit: **512 megabytes**

Jeroen has found a beautiful tree consisting of n nodes. Help him put graffiti on the tree. Jeroen applies graffiti to the tree by spraypainting a letter on each node. He has a favourite word and he wants to maximize the number of directed paths in the tree $u \rightarrow v$, such that the letters on the path u, p_2, p_3, \dots, v , spell out his favourite word.

Note that a directed path is a sequence of 1 or more nodes, where adjacent nodes in the sequence are neighbours in the tree, and no node appears more than once in the sequence. The tree itself is undirected, so each edge can be traversed in both directions.

Input

The first line of the input contains a single integer n ($1 \leq n \leq 300\,000$) — the number of nodes in the tree.

The next line contains a string w ($1 \leq |w| \leq 3$) consisting only of lowercase letters — Jeroen's favourite word.

Each of the next $n - 1$ lines contains the description of an edge. Each line contains two integers u_i and v_i ($1 \leq u_i, v_i \leq n, u_i \neq v_i$) — an edge connects node u_i to node v_i .

It is guaranteed that the $n - 1$ edges form a tree.

Output

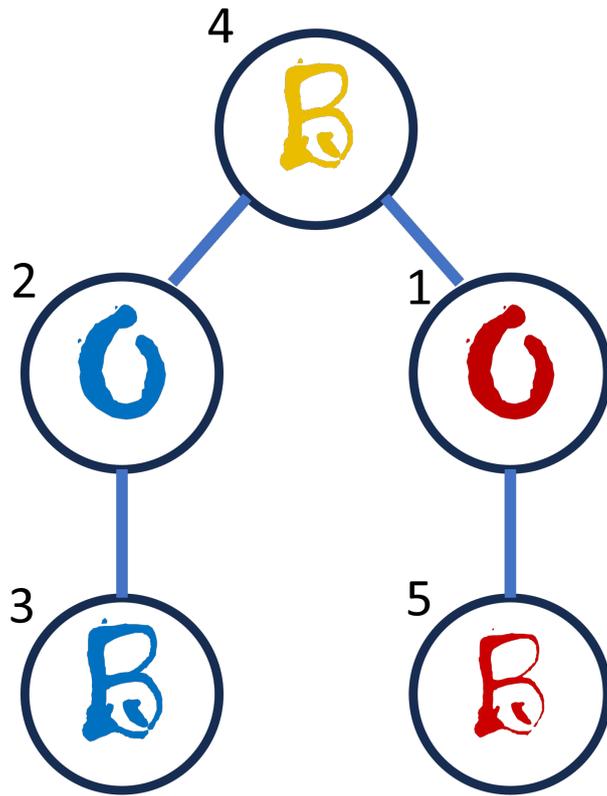
Output a single integer, the maximum number of directed paths that spell out Jeroen's favourite word, if you spraypaint the letters optimally.

Examples

standard input	standard output
1 a	1
3 orz 1 2 2 3	1
2 ab 1 2	1
5 bob 3 2 5 1 1 4 2 4	4

Note

For the last sample, one way to optimally spraypaint the letters is:



Visualization of example 4

Here the directed paths that contain the word bob, are:

- $4 \rightarrow 1 \rightarrow 5$
- $5 \rightarrow 1 \rightarrow 4$
- $4 \rightarrow 2 \rightarrow 3$
- $3 \rightarrow 2 \rightarrow 4$

In total, this gives 4 directed paths, with the word bob, which is optimal.