

Problem A. Balanced Strings

Input file: *standard input*
Output file: *standard output*
Time limit: 3 seconds
Memory limit: 256 mebibytes

You are given an undirected tree with n nodes. The nodes are numbered 1 through n . Each node is labeled with either '(' or ')'. Let $l[u \rightarrow v]$ denote the string obtained by concatenating the labels of the nodes on the simple path from u to v . (Note that the simple path between two nodes is uniquely determined on a tree.)

A balanced string is defined as follows:

- The empty string is balanced.
- For any balanced string s , the string “(s)” is balanced.
- For any balanced strings s and t , the string st (the concatenation of s and t) is balanced.
- Any other string is NOT balanced.

Calculate the number of the ordered pairs of the nodes (u, v) such that $l[u \rightarrow v]$ is balanced.

Input

The input consists of a single test case. The input starts with an integer n ($2 \leq n \leq 10^5$), which is the number of nodes of the tree. The next line contains a string of length n , each character of which is either '(' or ')'. The x -th character of the string represents the label of the node x of the tree. Each of the following $n - 1$ lines contains two integers a_i and b_i ($1 \leq a_i, b_i \leq n$), which represents that the node a_i and the node b_i are connected by an edge. The given graph is guaranteed to be a tree.

Output

Display a line containing the number of the ordered pairs (u, v) such that $l[u \rightarrow v]$ is balanced.

Examples

standard input	standard output
2 (1 2	1
4 (() 1 2 2 3 3 4	2
5 (()) 1 2 2 3 2 4 1 5	4