D. Polo the Penguin and Trees

Little penguin Polo has got a tree — a non-directed connected acyclic graph, containing n nodes and n - 1 edges. We will consider the tree nodes numbered by integers from 1 to n.

Today Polo wonders, how to find the number of pairs of paths that don't have common nodes. More formally, he should find the number of groups of four integers a, b, c and d such that:

- $1 \leq a < b \leq n$;
- $1 \leq c < d \leq n$;
- there's no such node that lies on both the shortest path from node a to node b and from node c to node d.

The shortest path betweem two nodes is the path that is shortest in the number of edges.

Help Polo solve this problem.

Input

The first line contains integer n $(1 \leq n \leq 80000)$ — the number of tree nodes. Each of the following n - 1 lines contains a pair of integers u_i and v_i $(1 \leq u_i, v_i \leq n; u_i \neq v_i)$ — the i-th edge of the tree.

It is guaranteed that the given graph is a tree.

Output

In a single line print a single integer — the answer to the problem.

Please do not use the %lld specificator to read or write 64-bit numbers in C++. It is recommended to use the cin, cout streams or the %I64d specificator.

Examples

input
4 1 2 2 3 3 4

output
2