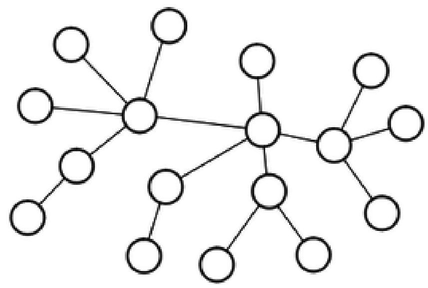
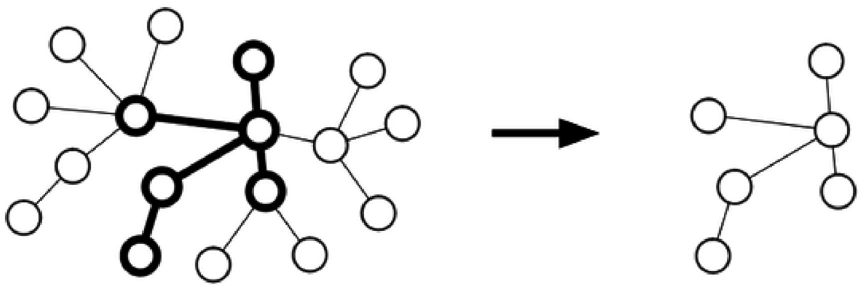
Tree Isomorphism

Little Alexey was playing with trees while studying two new awesome concepts: *subtree* and *isomorphism*.

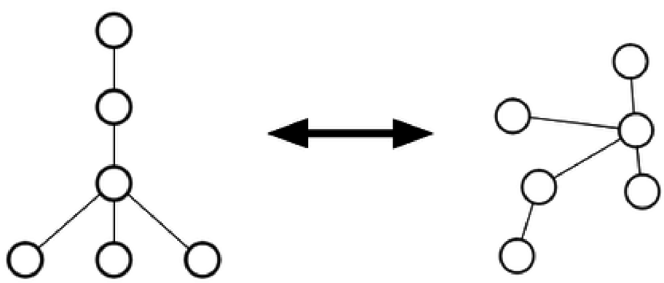
A *tree* is a connected, undirected graph with no cycles. We can denote a tree by a pair (V, E), where V is the set of vertices and E is the set of edges. Here's an example of a tree:



Let V' be a subset of V, and let E' be the set of edges between the vertices in V'. If the graph (V', E') is is a tree, then it is called a *subtree* of G. Here's an example of a subtree of the tree above:



Two trees are said to be *isomorphic* if they contain the same number of vertices and those vertices are connected in the same way. For example, the following two trees are isomorphic:



More formally, two trees $G_1 = (V_1, E_1)$ and $G_2 = (V_2, E_2)$ are said to be isomorphic if there exists a one-to-one correspondence $f : V_1 \to V_2$ such that $(u, v) \in E_1$ if and only if $(f(u), f(v)) \in E_2$.

Now he wonders, how many non-isomorphic trees can he construct using such a procedure? He asks you for help!

Input Format

The first line contains a single integer n denoting the number of vertices of the tree. The number of edges is $n - 1$. The vertices are numbered 1 to n.

The next $n-1$ lines describe the edges of the tree. The i^{th} such line contains two space-separated integers a_i and b_i denoting the vertices that the i^{th} edge connects.

Constraints

- $1 \leq n \leq 19$
- $1 \leq a_i, b_i \leq n$
- It's guaranteed that the given graph forms a tree.

Output Format

Print a single line containing a single integer denoting the number different non-isomorphic trees that Little Alexey can obtain.

Sample Input 0

```
3
1 2
1 3
```

Sample Output 0

```
3
```

Explanation 0

Here are the three trees that Little Alexey can obtain from the tree in this sample:

- a single vertex,
- Two vertices with edge between them, and
- the whole tree.

The following image illustrates it: