B. Begginer's Zelda

1 second, 256 megabytes

You are given a tree[†]. In one *zelda-operation* you can do follows:

- Choose two vertices of the tree u and v;
- Compress all the vertices on the path from u to v into one vertex. In other words, all the vertices on path from u to v will be erased from the tree, a new vertex w will be created. Then every vertex s that had an edge to some vertex on the path from u to v will have an edge to the vertex w.

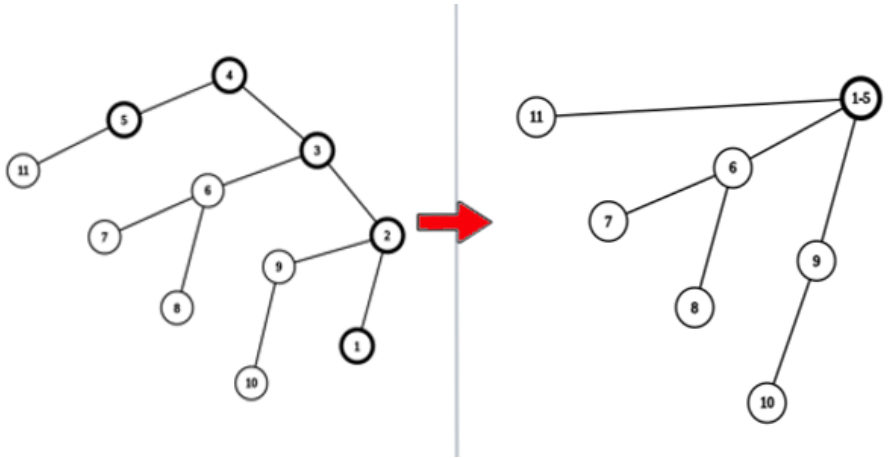

Illustration of a zelda-operation performed for vertices 1 and 5.

Determine the minimum number of zelda-operations required for the tree to have only one vertex.

[†]A tree is a connected acyclic undirected graph.

Input

Each test consists of multiple test cases. The first line contains a single integer t ($1 \le t \le 10^4$) — the number of test cases. The description of the test cases follows.

The first line of each test case contains a single integer n ($2 \le n \le 10^5$) — the number of vertices.

i-th of the next $n-1$ lines contains two integers u_i and v_i ($1 \le u_i, v_i \le n, u_i \neq v_i$) — the numbers of vertices connected by the i-th edge.

It is guaranteed that the given edges form a tree.

It is guaranteed that the sum of n over all test cases does not exceed 10^5.

Output

For each test case, output a single integer — the minimum number of zelda-operations required for the tree to have only one vertex.

input

```
4
4
1 2
1 3
3 4
9
3 1
3 5
3 2
5 6
6 7
7 8
7 9
6 4
7
1 2
1 3
2 4
4 5
3 6
2 7
6
1 2
1 3
1 4
4 5
2 6
```

output

```
1
3
2
2
```

In the first test case, it's enough to perform one zelda-operation for vertices 2 and 4.

In the second test case, we can perform the following zelda-operations:

1. $u = 2, v = 1$. Let the resulting added vertex be labeled as $w = 10$;
2. $u = 4, v = 9$. Let the resulting added vertex be labeled as $w = 11$;
3. $u = 8, v = 10$. After this operation, the tree consists of a single vertex.