MEXing

Do you remember the problem "First Missing Positive" from the first boot camp?

If you do, that a great help. If you don't it's fine don't worry.

So, let's define something called MEX as bellow:

The MEX (minimum excluded) of an array is the smallest not-negative integer that does not belong to the array. For example:

The MEX of [2,2,1] is 0 because 0 does not belong to the array.

The MEX of [3,1,0,1] is 2 because 0 and 1 belong to the array, but 2 does not.

The MEX of [0,3,1,2] is 4 because 0,1,2 and 3 belong to the array, but 4 does not.

Now you are given a tree with n nodes. For each node, you either color it in 0, or 1.

The value of a path (u, v) is equal to the MEX of the colors of the nodes from the shortest path between u and v.

The value of a coloring is equal to the sum of values of all paths (u, v) such that 1<=u<=v<=n.

What is the maximum possible value of any coloring of the tree?

Input format:

Each test contains multiple test cases. The first line of input contains a single integer t (1<=t<=10^4) the number of test cases. The description of test cases follows.

The first line of each test case contains a single integer n (1<=n <=2.10^5) the number of nodes in the tree.

The following n-1 lines of each test case contain 2 integers ai and bi (1<=ai, bi<=n, ai!=bi) indicating an edge between vertices ai and bi.

It is guaranteed that the given edges from a tree.

It is guaranteed that the sum of n across all test case does not exceed 2.10^5.

Output format:

For each test case, print the maximim possible value of any coloring of the tree.

Time limit: 3 seconds

Example:

4

3

1 2

2 3

4

1 2

1 3

1 4

10

1 2

1 3

3 4

3 5

1 6

5 7

2 8

6 9

6 10

1

Output:

8

15

96

1

Explain:

In the first sample, we will color vertex 2 in 1 and vertices 1,3 In 0.

After this, we consider all paths:

(1, 1) with value 1

(1, 2) with value 2

(1, 3) with value 2

(2, 2) with value 0

(2, 3) with value 2

(3, 3) with value 1

We notice the sum of values is 8 which is the maximum possible.