

Overview: Find the diameter of a tree.

Description: Given a tree (a connected, undirected, unweighted acyclic graph) with vertices numbered from 1 to n , we want to find the maximum distance between any pair of vertices. This distance is called the diameter of the tree. The given algorithm works as follows. Firstly, we choose an arbitrary vertex u and perform depth-first search from u to find the vertex v that is furthest from u . Then we perform depth-first search again from vertex v to find vertex w that is furthest from v . The distance between v and w is the diameter of the tree. The proof of correctness of this algorithm is left as an exercise to the reader.

Filename: bug2.{java, cpp, py}

Input: The first line contains a single integer n , the number of vertices in the tree. The next $n-1$ lines contain two integers a and b , which means that vertex a and b are connected by an edge.

Output: Print the diameter of the tree.

Assumptions: The input graph is a valid tree.
 $1 \leq n \leq 500$
 $1 \leq a, b \leq n$

Sample Input: 3
1 2
2 3

Sample Output: 2