Problem A
Acorn Quarrels
Time Limit: 1 Second, Memory Limit: 2G


Paul Danese, CC BY-SA 4.0, via Wikimedia Commons

There are N squirrels storing acorns in a tree. The (botanical) oak tree is also a graph-theoretic tree: a connected graph with N vertices labeled from 1 to N and $N-1$ undirected edges. Each squirrel sits at a different vertex of the tree and two squirrels are *neighbors* if their vertices share an edge.

In ascending order of vertex label starting with the squirrel at vertex 1, each squirrel steals one acorn from the neighboring squirrel that currently has the most acorns. If multiple neighbors are tied for having the most acorns, the squirrel steals one acorn from each of them!

To limit the fallout of these shenanigans, you want to distribute acorns to the squirrels so that each squirrel begins with at least N acorns (so that no squirrel runs out of acorns due to thefts) and ends with the same number of acorns after all N squirrels are done stealing from each other as they had originally. It can be shown that such a distribution exists where every squirrel begins with at most $2N-1$ acorns. Find any distribution satisfying these conditions.

Input

The first line of input contains an integer N ($2 \leq N \leq 10^5$), the number of vertices (and squirrels).

Each of the following $N-1$ lines contains two space-separated integers u and v ($1 \leq u, v \leq N, u \neq v$), indicating that an edge exists between vertices u and v. There is at most one edge between any pair of vertices, and the edges form a tree.

Output

Print N space-separated integers $d_1, d_2, \ldots, d_N$ satisfying $N \leq d_i \leq 2N-1$, where d_i is the number of acorns you would like to distribute to the squirrel at vertex i. Your solution will be accepted if every squirrel ends with the same number of acorns as they started with after all N thefts have occurred. It can be proved that such a distribution of acorns always exists.

Sample Input 1

```
5
1 2
1 3
1 4
2 5
```

Sample Output 1

```
6 5 7 7 9
```

Sample Input 2

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

Sample Output 2

```
14 15 10 9 13 14 14 14
```