

The Easiest Problem

You are given an unweighted, undirected tree T . We say T is special iff it has this property:

"All nodes of degree greater than or equal to 3 are surrounded by at most two nodes of degree two or greater."

Finding maximal size subtree of this tree so that it's a special tree.

Input

The first line of the input file contains one integer N --- number of nodes in the tree ($0 < N \leq 10^6$). Next $N-1$ lines contain $N-1$ edges of that tree --- Each line contains a pair (u, v) means there is an edge between node u and node v ($1 \leq u, v \leq N$).

Output

At the first line, output number of nodes in the optimal subtree you found. Next lines, print all edges belong to that subtree, each line contains a pair $u v$ means an edge between node u and node v .

Example

Input:

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

Output:

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