

# Task: NET

## Network

english

BOI 2015, day 1. Available memory: 256 MB.

30.04.2015

The government of Byteland has decided that it is time to connect their little country to the Internet, so that all citizens can participate in programming competitions and watch videos of cute cats. When it was time to build the network backbone of the country, they assigned the company Internet Optimists Inc. with connecting all the  $n$  computers of Byteland. The connections were made as direct links between pairs of computers in such a way that any pair of computers are connected by a sequence of links.

Byteland is not a rich country by any means, so to minimize costs the network topology was built in the form of a *tree* (i.e. there are exactly  $n - 1$  direct links between computers). Far too late, it was realised that this solution suffers from a serious drawback. If just a single link is broken, the computers of Byteland will be partitioned so that some computers cannot communicate with each other!

To improve the reliability of Byteland's network, it was decided that it should at least tolerate if a single link is broken. Your task is to help Internet Optimists Inc. to improve the network in a cheapest way. Given the network topology of Byteland (i.e. which  $n - 1$  pairs of computers are connected by direct links), find the minimum number of links that need to be added so that the network will still be connected if any single link is broken.

### Input

The first line of input contains a positive integer  $n$  ( $n \geq 3$ ), the number of computers in Byteland. For simplicity, all computers are numbered from 1 to  $n$ . Each of the following  $n - 1$  lines contains a pair of integers  $a$  and  $b$  ( $1 \leq a, b \leq n, a \neq b$ ) that describes a direct link between computers  $a$  and  $b$ .

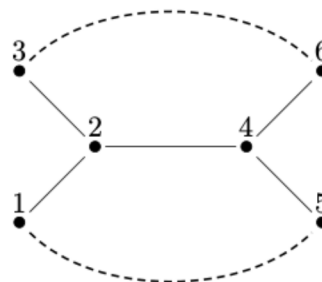
### Output

In the first line of output your program should write an integer  $k$ , the minimal number of links that should be added to the network. In each of the following  $k$  lines your program should write a pair of integers  $a$  and  $b$  ( $1 \leq a, b \leq n, a \neq b$ ) that denote the numbers of computers that should be connected by a link. The links can be written in any order. If there is more than one solution, your program should output any one of them.

### Examples

For the input data:

```
6
1 2
2 3
2 4
5 4
6 4
```



a correct result is:

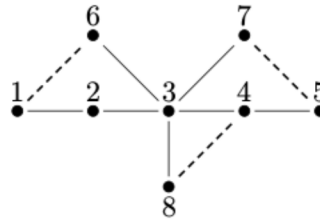
```
2
1 5
3 6
```

v. 3

Network 1/2

For the input data:

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



a correct result is:

3  
1 6  
5 7  
8 4

## Grading

Subtask	Conditions	Points
1	$n \leq 10$	18
2	$n \leq 2000$	45
3	$n \leq 500\,000$	37