Problem B

Money Matters

Our sad tale begins with a tight clique of friends. Together they went on a trip to the picturesque country of Molvania. During their stay, various events which are too horrible to mention occurred. The net result was that the last evening of the trip ended with a momentous exchange of "I never want to see you again!"s. A quick calculation tells you it may have been said almost 50 million times!

Back home in Scandinavia, our group of ex-friends realize that they haven't split the costs incurred during the trip evenly. Some people may be out several thousand crowns. Settling the debts turns out to be a bit more problematic than it ought to be, as many in the group no longer wish to speak to one another, and even less to give each other money.

Naturally, you want to help out, so you ask each person to tell you how much money she owes or is owed, and whom she is still friends with. Given this information, you're sure you can figure out if it's possible for everyone to get even, and with money only being given between persons who are still friends.

Input specifications

The first line contains two integers, n ($2 \le n \le 10000$), and m ($0 \le m \le 50000$), the number of friends and the number of remaining friendships. Then n lines follow, each containing an integer o ($-10000 \le o \le 10000$) indicating how much each person owes (or is owed if $o < 0$). The sum of these values is zero. After this comes m lines giving the remaining friendships, each line containing two integers x, y ($0 \le x < y \le n-1$) indicating that persons x and y are still friends.

Output specifications

Your output should consist of a single line saying "POSSIBLE" or "IMPOSSIBLE".

Sample input 1	Sample output 1
5 3 100 -75 -25 -42 42 0 1 1 2 3 4	POSSIBLE

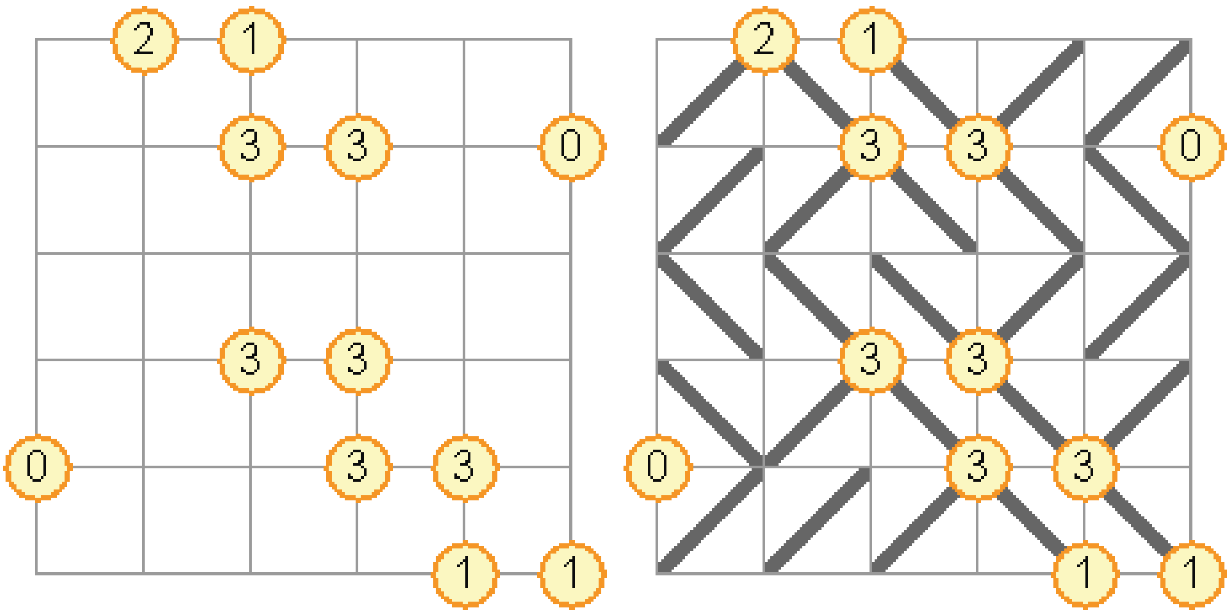
Sample input 2	Sample output 2
4 2 15 20 -10 -25 0 2 1 3	IMPOSSIBLE

Problem F

Gokigen Naname

Gokigen Naname is a Japanese puzzle game played on a square grid in which numbers in circles appear at some of the intersections on the grid.

The objective is to draw diagonal lines in each cell of the grid, such that the number in each circle equals the number of lines extending from that circle. Additionally, it is forbidden for the diagonal lines to form an enclosed loop.



The first figure shows the start position of a puzzle. The second figure shows the solution to the same puzzle. A *Gokigen Naname* puzzle always has exactly one solution.

Input specifications

The first line of the input contains a single integer n ($2 \leq n \leq 7$), the number of cells along each of the sides in the square grid. Then follow $n + 1$ lines containing the contents of the intersections of the grid cells. Each such line will contain a string of $n+1$ characters, either a digit between 0 and 4, inclusive, or a period ('.') indicating that there is no number at this intersection (arbitrarily many lines may connect to it).

Output specifications

The output should contain n lines, each line containing exactly n characters. Each character should either be a slash or a backslash, denoting how the corresponding grid cell is filled.

<table>
<tr><th>Sample input 1</th><th>Sample output 1</th></tr>
<tr><td>

```
3
1.1.
...0
.3..
..2.
```

</td><td>

```
\//
\\\
/\/
```

</td></tr>
</table>

<table>
<tr><th>Sample input 2</th><th>Sample output 2</th></tr>
<tr><td>

```
5
.21...
..33.0
......
..33..
0..33.
....11
```

</td><td>

```
/\\//
//\\\
\\\//
\/\\/
///\\
```

</td></tr>
</table>

Problem G

Flight Planning

The airline company NCPC Airways has flights to and from n cities, numbered from 1 to n, around the entire world. However, they only have $n-1$ different flights (operating in both directions), so in order to travel between any two cities you might have to take several flights. In fact, since the management has made sure that it's possible to travel between any pair of cities, there is exactly one set of flights a passenger have to take in order to travel between two cities (assuming you want to use the same airline).

Recently many of NCPC Airways frequent flyers have complained that they have had to change flights too often to get to their final destination. Since NCPC Airways doesn't want to loose their customers to other airline companies, but still keep the nice property of their flights, they have decided to cancel one of their current flights and replace it with another flight. Help the company by writing a program which finds the best flight to cancel and the best new flight to add so that the maximum number of flight changes a passenger might have to make when travelling between any pair of cities in which NCPC Airways operates is minimized.

The input will be constructed so that it is always possible to improve the maximum number of flight changes needed.

Input specifications

The first line in the input contains the integer n ($4 \leq n \leq 2500$), the number of cities NCPC Airways operates in. Then follow $n-1$ lines specifying the flights. Each flight is given as a pair of cities a and b ($1 \leq a, b \leq n$).

Output specifications

The output should consist of three lines. The first line should contain an integer, the minimum number of flights needed to take when travelling between any pair of cities after changing one of the flights. The second line should contain two integers, specifying the two cities between which the flight should be canceled. The third line should contain two integers, specifying the two cities where a new flight should be added.

If there are more than one optimal solution, any one of them will be accepted.

Sample input 1	Sample output 1
4 1 2 2 3 3 4	2 3 4 2 4

Sample input 2	Sample output 2
14 1 2 1 8 2 3 2 4 8 9 8 10 8 11 4 5 4 6 4 7 10 12 10 13 13 14	5 1 8 2 10