7th Balkan Olympiad on Informatics

Ioannina - Greece

Problems

DAY 1
Problem 1: Paving Roads (30 points)

In an isolated area of a poor country there exist N towns connected by roads. Every pair of towns is not necessarily connected by a direct road but there is a path connecting them. All these roads are unpaved. The government is interested in paving some of these roads, so that with these paved roads:

* there is a paved path from any of these towns to any other town, and
* this does not hold if any of these (paved) roads is left unpaved

In order to find the most preferable paving, the government wants to obtain the number of all the different possible pavings.

INPUT
Your program should read the input from the file INPUT.TXT, as follows: The first line contains the number of towns N (2 <= N <= 20) and the number of roads R (N-1 <= R <= 190) as two positive integer numbers separated by a space character. The next R lines describe the town connections as pairs of integers K and L, denoting a road connecting the K-th and L-th town (0 <= K, L <= N-1).

OUTPUT
Your program should produce its output into the file OUTPUT.TXT as follows:
There will be only one line of text, containing the number of different possible pavings.

EXAMPLE

INPUT.TXT	OUTPUT.TXT
4 4 0 2 0 1 0 3 1 3	3

Time Limit per Test: 5 seconds

DAY 1
Problem 2: The Flip Game (40 points)

There is an ancient solitaire game named "the flip game". It consists of an array of M rows and 9 columns of two-colored pegs, with a black and a white side. When a peg with its white side showing is flipped, it shows

its black side, and the other way around.

In each move of the game the player flips an entire row or an entire column.

The objective of the game is to leave as few pegs on their black side on the board as possible, by doing any number of moves.

INPUT
Your program should read the input from the file INPUT.TXT. The first line contains one positive integer number M (1 <= M <= 1000), denoting the number of rows in the game board. The next M consecutive lines contain exactly 9 characters, which are "0"s or "1"s, separated by one space character, where "0" means a peg showing its white side and "1" means a peg showing its black side.

OUTPUT
Your program should produce its output into the file OUTPUT.TXT as follows:
There will be only one line of text, containing the minimum possible number of pegs showing their black side, which are left on the game board.

EXAMPLE

INPUT.TXT	OUTPUT.TXT
4 1 1 1 1 1 1 1 1 1 1 0 0 0 0 0 0 0 0 1 0 0 0 0 0 0 0 0 1 0 0 0 0 0 0 0 0	1

Time Limit per test: 3 seconds

DAY 1
Problem 3: Guarding a Gallery (30 points)

Gallery 'El Greco' consists of a single room whose walls have either a north-south direction or a west-east direction. Moreover its floor plan forms a single closed polygonal line and there are no objects of any kind inside the room. There are precious paintings on every wall and the gallery owner wants to hire a guard to guard them. So, he's interested in determining whether there is any point in the gallery from where the guard can see all the walls. Note that a wall is still visible even if the guard is located along the line of the wall.

INPUT
Your program should read the input from the file INPUT.TXT, as follows: The first line contains the number N (4 <= N <= 1000) of the corners of the gallery. On the N following lines there are pairs of integers X and Y denoting the coordinates of the corners in a clock-wise order (0 <= X, Y <= 1000).

OUTPUT
Your program should produce its output into the file OUTPUT.TXT as follows:
There will be only one line of text, containing the integer coordinates of a point in the gallery from where the guard can
see all the walls; the coordinates should be separated by one space character. There may be many possible such points, output only one. If no such point exists then your program should output "NO POINT".