Problem A - Tracking Users
UAC1 2015

An online streaming website is interested to know at what times their services are used the most. This is useful information because they can sell advertising at a more expensive rate at such times.
In order to do that, they keep track of whenever a user connects or disconnects on their website. Each day they collect a lot of data and at the end of the day they need to analyze it.
Their data is a set of n sintervals $[start_i, end_i]$ representing that someone got connected at minute $start_i \in \{1, \ldots, 1440\}$ and disconnected at the end of minute $end_i \in \{1, \ldots, 1440\}$. Note that one day has 1440 minutes and they start to count from minute 1.

Your job is to find the times in the day that have the maximum number of users connected to the website.

Input

The first line of the input contains a single integer n representing the number of intervals. Then follow n lines each with two integers separated by a single space, $start_i$ and end_i.

Constraints

- $1 \leq n \leq 1000$
- $1 \leq start_i \leq end_i \leq 1440$

Output

A single line with the minutes of the day that reach the maximum number of users. Print the minutes in **increasing order**.

Sample Test Cases

Sample Input 1

```
3
1 1
```

2 2
3 3

Sample Output 1

1 2 3

Sample Input 2

4
1 10
2 9
3 8
4 7

Sample Output 2

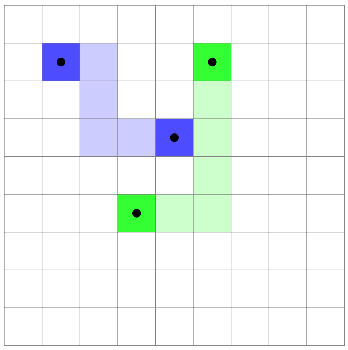
4 5 6 7

Problem B - Moving Money
UAC1 2015

Two rival mafia organizations in Grid-City want to plan routes to move money between their two main houses. As its name suggests, Grid-City is a city whose steets form a grid. All the steets in Grid-City have the same length. To avoid conflicts between them, they would like to define non-intersecting routes to move money. However, moving money is a risky operation because the longer it takes the more likely it is that they will get cought by the police. Hence they want their routes to be shortest paths.

Can you help them figure out if it is possible to find such routes?

For example, the following picture shows an example where it is possible:



Input

Two lines, each with 4 integers separated by single spaces. The first line contains $x_1^1, y_1^1, x_2^1, y_2^1$ and the second $x_1^2, y_1^2, x_2^2, y_2^2$. The pairs (x_1^1, y_1^1) and (x_2^1, y_2^1) represent the coordinates of the two houses of the first mafia organization and (x_1^2, y_1^2) and (x_2^2, y_2^2) represent the coordinates of the houses of the second mafia organization.

Constraints

- $0 \leq x_1^1, y_1^1, x_2^1, y_2^1, x_1^2, y_1^2, x_2^2, y_2^2 \leq 1000$

Output

A single line with `yes` if it is possible to find two non-intersecting shortest paths between the houses or `no` otherwise.

Sample Test Cases

Sample Input 1

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

Sample Output 1

```
yes
```

Sample Input 2

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

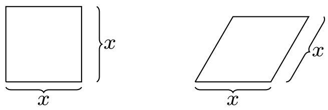
Sample Output 2

```
no
```

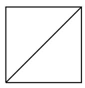
Problem C - Cross Bracing
UAC1 2015

Many engineering structures are constructed using triangles. Why is this?

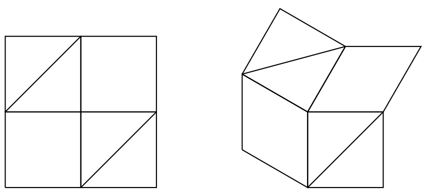
The simple answer is that triangles are stable. A triangle has three sides and, if these are fixed in length, there is only one configuration they can be in. There is no flexibility or freedom. On the other hand, a quadrilateral has more degrees of freedom. Without changing any of the lengths of the sides the shape can be deformed.



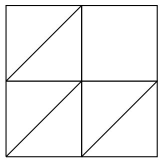
Triangles help to keep structures rigid. So to make a square rigid we can use a cross section sub-dividing it into two triangles as shown in the figure.



In this problem we are interested to know if a given grid structure is rigid or not. A grid structure is rigid if no square can be deformed. For instance the following grid structure is not rigid as it can be deformed while maintaining the lengths of all non-crossed squares and the integrity of the crossed squares.



By adding a cross section on the lower left square, we obtain a rigid structure.



Given a grid with some cross sections, determine what is the minimum number of cross sections that must be added so that it becomes rigid.

Input

The first line contains two integers r and c giving the number of rows and columns of the grid structure. Then follow r lines each containing a binary string s_i of length c such that $s_{ij} = 1$ if and only if square at row i and column j is braced.

Constraints

- $1 \le r, c \le 1000$

Output

A single line with a non-negative integer giving the minimum number of cross sections that must be added in order to make the structure rigid.

Sample Test Cases

Sample Input 1

```
2 2
11
01
```

Sample Output 1

```
0
```

Sample Input 2

```
9 7
0000001
0000100
0000100
0101010
0000010
0100001
1010000
```

0001000
0010000

Sample Output 2

2

Problem D - Elevator Ride
UAC1 2015

You have two friends that live in a very tall building with n floors. The building has two elevators. Your friends decide to race home from work. They both arrived at the same time at the building and after fighting for a while to try to be the first into the elevator, they both decide that it is smarter that each of them uses one of the two elevators.

Lack of luck, or perhaps the fact that the elevators are quite old, both the elevators get stuck on their way up.
Fortunately they both have cellphone reception and are able to call each other. Scared, they don't care about the race anymore. All they want is to meet each other in some floor. After a few minutes of button bashing one realizes that that the elevators can still move but only every k_1 floor. This means that if he is in floor f_1 he can go to $f_1 \pm t \cdot k_1$ for any t (within the bounds of the building). He asks your other friend if his elevator also does that and after trying for a while he says that his elevator can move every k_2 floors, that is, he can go to $f_2 \pm t \cdot k_2$ for any t (within the bounds of the building).

Knowing you are their smartest friend they called you to help them figure out if they can meet in some floor.

Input

A single line with 5 integers n, f_1, f_2, k_1 and k_2 separated by a single space.

Constraints

- $1 \leq n \leq 2^{63} - 1$
- $0 \leq f_1, f_2 \leq \min(n, 2^{31} - 1)$
- $1 \leq k_1, k_2 \leq \min(n, 2^{31} - 1)$

Output

A single line with the word `yes` if they can meet in a floor f such that $0 \leq f \leq n$ or `no` otherwise.

Sample Test Cases

Sample Input 1

6
1 2 2 3

Sample Output 1

yes

Sample Input 2

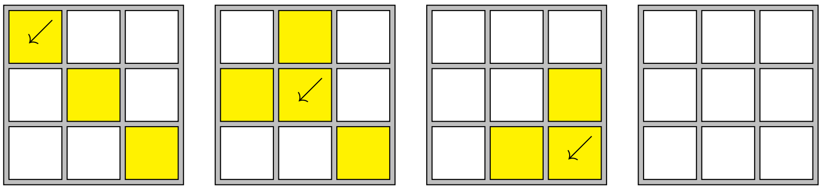
8
0 1 2 2

Sample Output 2

no

Problem E - Lights Out
UAC1 2015

The 'lights out puzzle' is a puzzle that consists of a grid of lights, some turned on and the others turned off. The objective is to turn all the lights off. Each light has a switch that, when pressed, changes the state of that light and all of its neighboring lights (horizontally and vertically).
The next image shows an example with a 3 by 3 grid of lights. The initial configuration in the one shown on the left and the switches that are pressed are indicated with an arrow ($\swarrow$).



In this problem we consider a variant of this puzzle where we consider that if a button is on the edge of the grid then positions wrap around. For instance, the neighbors of the button at the top left corner (1, 1) in a n by n grid are $(2, 1)$, $(1, 2)$, $(n, 1)$ and $(1, n)$.

Given the initial state of the puzzle, your job is to figure out if there is a solution to the puzzle.

Input

The first line contains one integer n representing the size of the puzzle. Then follow n lines each containing a string s_i over the alphabet $\{\texttt{.}, \texttt{o}\}$ of length n such that $s_{ij} = \texttt{o}$ if and only if the light at row i and column j is on.

Constraints

- $1 \leq n \leq 50$

Output

A single line with a string: `yes` if the puzzle is solvable and `no` otherwise.

Sample Test Cases

Sample Input 1

```
3
.o.
oo.
.oo
```

Sample Output 1

```
no
```

Sample Input 2

```
3
ooo
o..
o..
```

Sample Output 2

```
yes
```