Robot Race

task: `race`	input file: `stdin`	output file: `stdout`
points: 100	time limit: `2000 ms`	memory limit: `1 GB`

A robot race on a maze will be held in Byteland. The maze has a shape of a rectangle and is divided into $n \times m$ fields arranged in n rows and m columns. Each of the fields is either empty or contains an obstacle.

The contestants register their robots for the competition and come to the maze one after another. Each of the contestants gets random coordinates of the initial field and the target field. The robot is then placed in the initial field and must get to the target field using a sequence of steps.

To make the game more challenging, the rules specify that in each step the robot can only move one field right or one field down in the maze. Moving the robot in any other direction is not allowed.

The contestant whose robot gets fastest from the initial field to the target field wins the competition. If the robot does not get to the target field in the time limit, the contestant in disqualified.

The organizers of the competition realized that if the contestant gets bad coordinates, the robot will not be able to get to the target field using any sequence of moves. In that case they would like to give the contestant another pair of coordinates.

Task

You are given a map of a maze of size $n \times m$ and q pairs of coordinates of the initial and target fields. Determine for each pair of coordinates whether it is possible to get from the initial field to the target field using a sequence of steps right or down.

Input

The first line contains three integers n, m and q: the number of rows and columns of the maze and the number of pairs of coordinates.

Each of the following n lines contains m characters describing the fields of the maze. The character `.` represents an empty field and the character `#` a field with an obstacle.

Then q lines follow describing the pairs of coordinates. The i-th of them contains four integers r_{1i}, c_{1i}, r_{2i}, c_{2i}: the row and the column of the initial field and the row and the column of the target field.

It holds $1 \le n, m \le 1000$ and $1 \le q \le 10^6$.
For all $i \in \{1, 2, \ldots, q\}$ it holds $1 \le r_{1i}, r_{2i} \le n$ and $1 \le c_{1i}, c_{2i} \le m$.
For all $i \in \{1, 2, \ldots, q\}$ the fields in the maze with coordinates (r_{1i}, c_{1i}) and (r_{2i}, c_{2i}) are empty.
Furthermore, in 20 % of the testcases $q \le 300$.

Output

Output q lines. The i-th line should contain the word `YES` if the robot can get from the i-th initial field to the i-th target field, or the word `NO` otherwise.

Sample

input	output

<pre>3 4 5
.#..
.##.
....
1 1 3 4
1 3 3 4
1 1 1 1
1 1 2 4
2 1 2 4</pre>
	<pre>YES
	YES
	YES
	NO
	NO</pre>