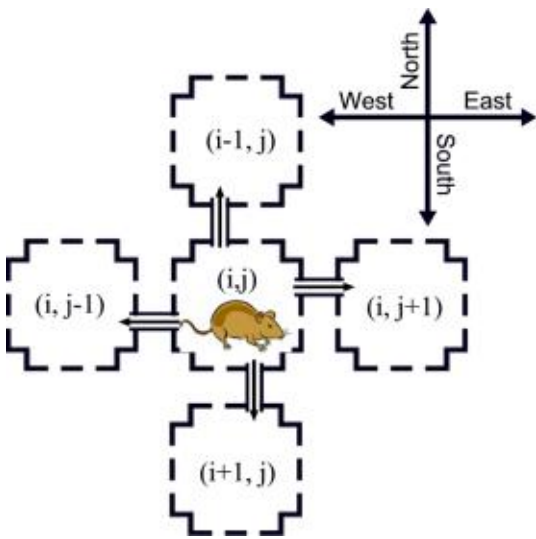


Rats like many other rodents are used in the labs for scientific experiments. But unlike many cruel surgical experiments our experiment on the rats would be a rewarding psychological one. We are trying to establish the fact that the rats have a good cognitive map in their brain they have a good sense of directions. Here we would like to give an empirical proof to support our claim.



In our experiment, we build a maze for our rats. This maze is basically a rectangular grid of cells. There are tunnels that lead from one cell to another. Because of the rectangular arrangement, the tunnels can run in the four major directions only. From the cell at location  $(i, j)$  the rats may go East to  $(i, j + 1)$ , West to  $(i, j - 1)$ , North to  $(i - 1, j)$  and South to  $(i + 1, j)$  as shown in the figure on the right. These tunnels are directed. Thus a tunnel that takes a rat from cell from  $(i, j)$  to  $(i, j - 1)$  not necessarily provides a path to get back to cell  $(i, j)$ . Some of these tunnels are open ended, if a rat goes through that tunnel, it would fall out of the maze. Now we allure the rats by putting mouth watering cheese cuts in some particular cells. Our experiment requires the rats to be trained to remember the sequence of **E** (East), **W** (West), **N** (North), **S** (South) moves that lead them to these cheese cuts.



We train a rat by releasing it in a designated start cell in the maze. Then it goes through the maze at its own will — taking the **EWNS** turns whimsically. But at some point in time it is bound to lose its vigor and stop moving. Then we take the rat out from the maze. If it stopped on a cell that contained a cheese cut we reward it with another cheese cut, leaving the original one in the maze; but if it stops on a cell that does not have any cheese cut, it goes unrewarded. One may wonder what would happen if the rat falls out of the board by following an open ended tunnel. Our explanation is simple, we put cheese cuts only in some particular cells in the maze there is no cheese cuts in the world outside. So if a rat falls out it doesnt get any cheese.

We have seen that after training a rat several times in this way it seems to remember the rewarding move sequences correctly. Our experiment requires that we release the rat in the same starting cell and the cheese cuts are placed in exactly the same cells every time we put it in the maze. So if the rat finds that some “**EESSWNNS**” leads it to a cheese cut, it would always find a cheese cut with that sequence of moves. Our smart rat only needs to differentiate the moves that are rewarding from the ones that are not. However, your task is not as simple as that.

We plan to train our rat in a maze and test its learning ability in a different maze. But to do that we need to make sure that these two mazes are identical. For our purpose identical mazes not necessarily mean identically constructed mazes. What we need is that they would be identical to the rats. **If a rat has a sequence of moves leading to a cheese cut in the training maze, it would also get a cheese cut with the same sequence of moves in the new maze. And if a sequence of moves leads it to nothing in the training maze, the same sequence must be unrewarding for the rat in the new maze.** This is where we need your help. Your task is to take the configuration of the two mazes as input, and tell us if they are identical from the rats perspective.

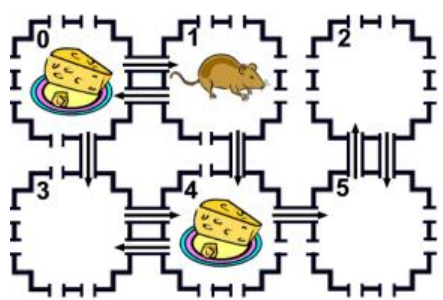
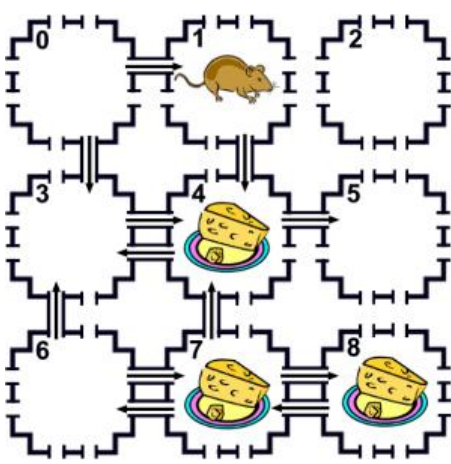
## Input

The input file contains several test cases. The first line of the input gives you the number of test cases,  $T$  ( $1 \leq T \leq 25$ ). Then  $T$  pairs of maze configuration will follow. The first line of a maze configuration starts with the dimension – the number of rows  $R$  ( $1 \leq R \leq 20$ ) and the number of columns  $C$  ( $1 \leq C \leq 20$ ) for the maze. Each of the next  $R$  lines would describe the  $C$  cells in that column. Each cell is represented by a 4-bit number (numbers in the range 0 to 15). These numbers allows us to list all possible outgoing tunnels. The outgoing tunnels for North, East, South and West are represented by the 0 – *th*, 1 – *st*, 2 – *nd* and 3 – *rd* bits respectively. If a bit is set to zero then the tunnel in that direction is open ended leading the rat out of the maze; whereas a bit set to one indicates that the rat can go to the next cell in that particular direction. We label the cells in the maze in row major order starting from 0. After the description of the maze cells, the next line would give you the label of the starting cell for that maze. The first integer in the next line would give you the number of cells that would contain cheese cuts. Then that many cell labels will follow in the same line. You may find it helpful to relate the images in the illustration section to the second sample input.

## Output

For each set of input print the test case number first. Then print ‘Yes’ if our rat would find the pair of mazes to be identical, otherwise print ‘No’. The Sample Input/Output section will clarify the formatting issues.

**Illustration:** The following pictures illustrate the second sample test case. The numbers in the cells denote the cell labels, the rat shows the starting cell and the cheese cuts show the cells that would contain the cheese cuts. In the first maze the rat falls out of the maze if it takes a west bound move, thus fails to get any cheese. But in the second maze a west bound move would earn it a cheese cut.



## Sample Input

```
2
3 3
6 4 0
2 10 0
3 11 8
1
3 4 7 8
2 3
6 4 4
2 10 1
1
2 0 4
3 3
6 4 0
2 10 0
3 11 8
0
3 4 7 8
2 3
6 12 4
2 10 1
1
2 0 4
```

## Sample Output

```
Case 1: Yes
Case 2: No
```