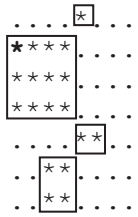
2. Blobs

Program Name: Blobs.java Input File: blobs.dat

Johnny is studying different shapes in a plane. For this particular study, he refers to the shapes as blobs even though they are solid rectangles. He represents his blobs in a rectangular grid as a collection of one or more contiguous asterisks (*). Contiguous means that the asterisks must be adjacent either horizontally or vertically. Non-blob characters are represented by periods (.). In the diagram below, there are 4 blobs.



You are to write a program that will determine the location of the uppermost, leftmost character of a blob given the coordinates of a given character in a grid. The uppermost, leftmost character of the largest blob (bolded *****) in the example above is row 2, column 1 or `2 1`. Rows and columns are numbered beginning with one.

Input

The first line of input will contain a single integer `n` that indicates the number of data sets to follow. For each data set:

- the first line will contain three integers in the form `r c s`
  - `r` ≥ 3 is the number of rows in the grid
  - `c` ≥ 3 is the number of columns in the grid
  - `s` is the number of test cases for that grid
- the next `r` lines will contain the grid
- the next `s` lines will each contain an ordered pair `x y`, 1 ≤ `x` ≤ `r` and 1 ≤ `y` ≤ `c` which is the location of a character in the grid

Output

For each ordered pair `x y`, you will print the coordinates in the form `j k` of the uppermost, leftmost character of the blob where 1 ≤ `j` ≤ `r` and 1 ≤ `k` ≤ `c`. If the test case falls on a square that is not part of a blob, print `NO BLOB`.

Example Input File

```
2
7 8 2
....*...
****....
****....
****....
....**..
..**....
..**....
2 1
5 3
5 6 3
...***
***...
...***
**.***
**.***
5 6
1 6
5 1
```

2. Blobs (cont.)

Example Output to Screen

```
2 1
NO BLOB
3 4
1 4
4 1
```