You have a chessboard with N rows and N columns. You want to color each of the cells with exactly N colors (colors are numbered from 0 to $N-1$). A coloring is valid if and only if no row or no columns contains a duplicate color. That means in the final coloring each cell should be colored by a color from 0 to $N-1$. No row will contain two cells of the same color and same for the no column.

Your friend already colored some of the cells in the top-left corner (cells that are in the first R row and in the first C columns). Your job is to color the rest of the cells or tell it is impossible to color.

Input

Input starts with an integer T ($1 \le T \le 50$), the number of test cases. Each test case starts with a line containing 3 integers N ($1 \le N \le 50$), R ($0 \le R \le N$) and C ($0 \le C \le N$). Each of the next R line contains C integers in each line. The j-th integer in the i-th line indicates the color of the cell located at row i and column j. The initial coloring will be always valid. That means No row or column will have duplicate integers. The initial colors are between 0 to $N-1$ inclusive.

Output

First line of each test case will be 'YES' when it is possible to color the rest of the cells or 'NO' otherwise. In case it is possible to reach a valid coloring the next N line should contain N integers in each line denoting the colors of the cells(in one of the valid final colorings). Each of these N integers should be separated by a single space. There can be multiple valid colorings; you can output any valid coloring. When the solution is 'NO', you do not have to output additional line.

Sample Input

```
3
4 2 2
1 2
2 0
4 2 3
0 1 2
2 0 1
3 1 1
0
```

Sample Output

```
YES
1 2 0 3
2 0 3 1
3 1 2 0
0 3 1 2
NO
YES
0 1 2
2 0 1
1 2 0
```