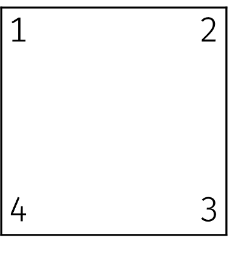
Given a matrix of m x n and a folding direction, determine what the matrix will look like when folded in the given way.

Given four points on a piece of paper, assume the top left corner has a value of 1, and going in a clockwise direction, each corner increases in value by 1.



**Input Format**

- The first line is the number of test cases (t)  
- The first line of each test case is the size of the matrix (m n)  
- The second line of each test case is the folding direction from point c to d  
- For the next m lines, each line is the row of that matrix with n values

**Constraints**

- 1 ≤ c, d ≤ 4 and c ≠ d  
- 1 ≤ m, n, t ≤ 1,000,000  
- 0 ≤ value inside the matrix (v) ≤ 9  
- When folded, if the value of vij that is folded over vi'j' are not the same, we considered it asymmetrical (see Sample 3)  
- If a side has the size of 1, we cannot fold perpendicularly to that side.

**Output Format**

For each test case, print the number of the test case starting with 1 followed by the list the resulting matrix of the folded paper. Use a hyphen symbol (-) for disappearing spaces. If the paper cannot be folded symmetrically, list “error.”

**Sample Input 1**

1  
1 1  
1 2  
0

**Sample Output 1**

1  
error

**Sample Output 1 Explanation**

The matrix has size 1x1, so it’s not foldable.



**Sample Input 2**

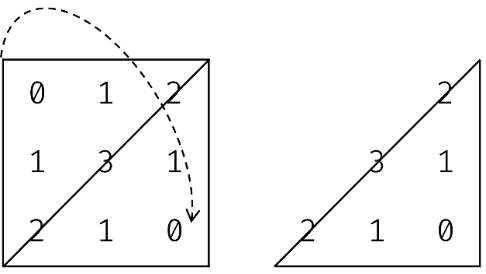
1  
3 3  
1 3  
0 1 2  
1 3 1  
2 1 0

**Sample Output 2**

1  
--2  
-31  
210

**Sample Output 2 Explanation**

If we fold it from corner 1 to corner 3, the numbers match the opposite numbers, so we can fold it symmetrically. The top left part has disappeared, so it is replaced with a hyphen symbol (-) in the output.



**Sample Input 3**

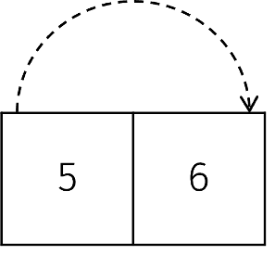
1  
1 2  
1 2  
5 6

**Sample Output 3**

1  
error

**Sample Output 3 Explanation**

We have a foldable matrix. However, when we fold from corner 1 to corner 2, the values inside the matrix are different (5 and 6), so we cannot fold it symmetrically.



**Sample Input 4**

1  
1 3  
1 2  
1 2 1

**Sample Output 4**

1  
-21

**Sample Output 4 Explanation**

The numbers match each other in the opposite side.

