2016 Spring CS300 Programming Assignment #1

(Due at Apr. 13 PM 11:59)

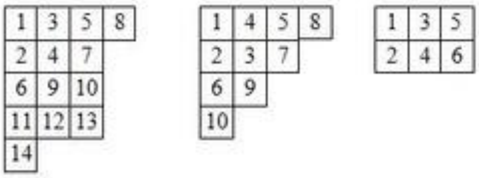
A ***Young diagram*** is an arrangement of boxes in rows and columns conforming to the following rules:

- the boxes in each row and each column are contiguous,
- the left borders of all rows are aligned, and
- each row is not longer than the one above.

Young tableau is obtained by filling in the boxes of the ***Young diagram*** with numbers. Each box in ***Young tableau*** is filled according to the following rules:

- each box contains a single integer,
- each integer is greater than or equal to the integer in the box to its left, and
- each integer is strictly greater than the integer in the box above.

Here are some examples of ***Young tableaus*** :



1	3	5	8
2	4	7	
6	9	10	
11	12	13	
14			

1	4	5	8
2	3	7	
6	9		
10			

1	3	5
2	4	6

Let's assume there is an $m \times n$ matrix which represents a ***Young tableau.*** For some of the entries which do not exist in the ***Young tableau,*** we mark it as ∞ in the matrix.

Your task is to keep the ***Young tableau*** after an **extraction of minimum number** or insertion of a value in $O(m+n)$ time. Your algorithm should use a recursive subroutine that solves an $m \times n$ problem by recursively solving either an $(m-1) \times n$ or $m \times (n-1)$ subproblem.

Input

Input starts with a line containing three integers m, n, and k $(1 \leq m, n \leq 1{,}000;\ 1 \leq \mathrm{k} \leq 1{,}000{,}000)$, describing the size of $m \times n$ matrix and the number of extractions and insertions. Next k lines contain the guide of insertions and extractions. In the case of extraction, single character 'E' exists to represent the extraction. In the case of insertion, two information, alphabet 'I' and number to insert, exist to represent the insertion. Next n lines contain the matrix. An infinite elements in the matrix is replaced with a sharp(#). There is no duplicated number in the matrix.

Output

Display a matrix after k extractions and insertions in order.

Sample Input	Output for Sample Input
3 2 1 E 1 3 5 2 4 6	2 3 5 4 6

4 4 3
E
I 5
E
1 3 7 11
2 8 13 #
6 9 # #
10 # # #
	3 7 11 13
	5 8
	6 9
	10