

Problem B

Budget

source: budget.*

I have a problem. I was supposed to make a budget proposal for this competition and I haven't started yet. It must be done at the end of the competition or the others will notice that I haven't done it. This is where I need you.

The budget proposal is a matrix where the rows represent the different kinds of expenses and the columns represent the different sites.

We had a meeting about this some time ago where we discussed the sums over different kind of expenses and sums over different sites. There was also some talk about special constraints: someone mentioned that Stockholm would need at least SEK 2000 for food and someone from Lund argued they wouldn't use more than SEK 100 for paperclips. Anyway, I'm sure there was more, I'll go and try to find some notes from that meeting if you write a program that can solve the problem when we get the data.

And, by the way, no one really reads budget proposals anyway, so you'll just have to make sure that it sums up properly and meets all constraints.

Input specifications

The first line of the input contains an integer N giving the number of test cases.

The first line of each test case contains two integers m, n giving the number of rows and columns ($m \leq 200, n \leq 20$). The second line contains m integers, giving the row sums of the matrix. The third line contains n integers, giving the column sums of the matrix. The fourth line contains an integer c giving the number of constraints. The next c lines contain the constraints.

Each constraint consists of two integers r, c specifying some entry (or entries) in the matrix (the upper left corner is 1 1 and 0 is interpreted as all, i.e. 4 0 means all entries on the fourth row and 0 0 means the entire matrix), one element from the set $\{<, =, >\}$ and one integer v , with the obvious interpretation. For instance, the constraint 1 2 > 5 means that the entry in the 1st row and 2nd column must have an entry strictly greater than 5, and the constraint 4 0 = 3 means that all elements in the fourth row should be equal to 3.

Output specifications

For each case output a matrix of non-negative integers meeting the above constraints or the string "IMPOSSIBLE" if no legal solution exists.

Output an empty line between cases.

Sample input

2

2 3

8 10

5 6 7

4

0 2 > 2

2 1 = 3

2 3 > 2

2 3 < 5

2 2

4 5

6 7

1

1 1 > 10

Output for sample input

2 3 3

3 3 4

IMPOSSIBLE