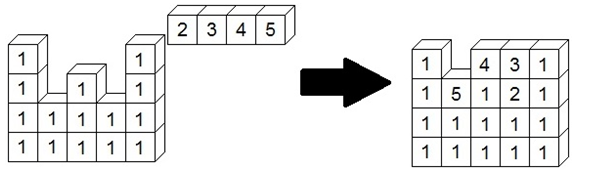
# Pushing Blocks

The Company Blocks Regular Inventing Usefulness of Something, better known as Brisa , build blocks, always the same size . One detail that stands out is the manner in which the blocks are stored in stock, after manufactured . They are formed by a row of cells . Withdrawal of a stock box is somewhat cluttered when , for choosing a cell at random and cut up some top block it. However , the storage medium is somewhat interesting: a conveyor located at the top of the stack straight rightmost stock is used . With this, it forms a queue with the new blocks . The belt right wheel to the left. So there is a vacant space in one of two cells, the block will be inserted in it, if there is not, it progresses to the following cells. Below is an example of insert blocks .



**Input**

There will be several test cases . Each test case have three integers , M , P and F , indicating the rightmost stack height , the number of stacks of blocks and the size of the row of blocks to be inserted . Following this, M lines are read with P values ​​, with values ​​1 , which is represented block, and 0 , representing which does not block. Next, a line is read with F values ​​representing the queue with the new blocks . The last test case is represented by three zeros , and should not be processed.

**Output**

For each test case , print the cells after the addition of new blocks. In some cases, a row of new blocks is more than sufficient for all the cells remain the same size. In this case , disregard the blocks that are left in the queue.

| **Input Sample** | **Output Sample** |
| --- | --- |
| 4 5 4  1 0 0 0 1  1 0 1 0 1  1 1 1 1 1  1 1 1 1 1  2 3 4 5  5 3 6  1 0 1  1 0 1  1 0 1  1 0 1  1 0 1  4 5 6 7 8 9  0 0 0 | 1 0 4 3 1  1 5 1 2 1  1 1 1 1 1  1 1 1 1 1  1 8 1  1 7 1  1 6 1  1 5 1  1 4 1 |

**Tasks:** Your specific tasks are as follows:

(1) Provide a detailed description of the working of the algorithm along with an example.

(2) Provide a pseudo code for the algorithm.

(3) Analyze the theoretical run-time complexity of the algorithm and show that it is Θ(nlogn).

(4) Analyze the theoretical space complexity of the algorithm.

(5) Discuss the correctness of your algorithm

(6) Implement the algorithm