Help Mr. Heckles!

Input file:	`standard input`
Output file:	`standard output`
Time limit:	2 seconds
Memory limit:	512 megabytes

Note: Use Fast I/O for both java and c++.

With the festive season just around the corner, the demand for jam bottles has increased immensely in the city of Baradoland.
Mr. Heckles, known for making different flavoured jams owns a huge shop - highly prominent in Baradoland.

He has **M** shelves, each shelf containing **N** distinct flavoured jam bottles denoted by a unique id a_{ij}. To ease out the process, he arranged the jam bottles in the **increasing order of their ids in each row and each column of the entire stack of jam bottles.**

Can you help Mr. Heckles find the exact position (i, j) of jams ordered by **Q** customers, as he is too old to handle the crowd? Here i represents the shelf number from the top(or row number) and j represents the position of the jam bottle in shelf i from the left (or column number).
As soon as a jam bottle is ordered, a new jam bottle of the same flavour (same id) is placed at the same position. Assume infinite supply of each available jam bottle.

Output **-1** if the jam bottle with a given id is out of stock.

Input

The first line contains 2 integers **M** (1<=M<=10^5) and **N** (1<=N<=10^5) , (M x N <=10^6)
, **(min(M, N) <=10^2)** - the number of rows and the number of columns in the stack of jam bottles respectively.
Each of the next **M** lines contain **N** distinct numbers in increasing order (a_{i1}, a_{i2}, ..., a_{iN}), (0<=a_{ij}<=10^{18}) - the ids of jam bottles.

The next line contains an integer **Q**, (0<=Q<=10^5) - the number of customers.
Each of the next **Q** lines contain a number - q^{th} line denoting the id of the jam ordered by the q^{th} customer.

Output

Output **Q** lines, q^{th} line containing 2 space-separated integers i j denoting the position of the jam bottle ordered by the q^{th} customer.
Output **-1** if the the jam bottle ordered is out of stock.

Example

standard input	standard output
3 4 11 25 90 128 34 56 99 151 56 87 123 187 3 99 156 128	2 3 -1 1 4