

After a long summer's march through the rough terrain of northern America, the indian tribe had found a place where they hopefully would be left alone. The chief proclaimed that this would be the new place for their village, despite the rocky nature of the landscape. They set a temporary camp for the night, content with the piece of land they had discovered. The very next day however, it stood clear that some effort planning the locations of the Teepee tents had to be made. It was simply too great a difference in altitude between the tents, making the walk along some paths extremely tiresome. Therefore, the chief ordered his witty son, Fast Thought, to find a connected area in their vicinity, large enough to host all tents of the tribe, having as small difference between the highest and lowest point as possible.

The task called for some altitude measurements of their whereabouts, which caused no problem for Fast Thought, since he was wise in the ways of trigonometry. He divided the land into squares big enough to host a tent each and estimated the altitude of each square. Now the problem was reduced to finding a connected region containing at least as many squares as there were tents, having the smallest difference between the highest and lowest altitude. He drew a rectangular matrix A , representing the area, where the entry $a_{i,j}$ at row i and column j , was the altitude of the square with coordinates (i, j) . He considered an entry $a_{i,j}$ adjacent to the entries $a_{i,j+1}$, $a_{i+1,j}$, $a_{i,j-1}$ and $a_{i-1,j}$, and called a set of entries connected if for every pair of entries in the set, there was a connecting path of adjacent entries in it. Since the size of the tribe altered with time, Fast Thought decided to solve the problem for a general number of tents. Thus the problem left to solve for him was to find a set of at least k connected entries $a_{i,j}$ in the matrix A , such that the difference between the largest and the smallest entry in the set was minimized.

Input

On the first line of input there are two integers, r, c (≤ 40), giving the dimension of the matrix A . The following r lines, each containing c integers between 0 and 99, are the entries $a_{i,j}$ of the matrix. The next line contains a single integer $n \leq 100$, and is followed by n lines each holding a single positive integer $k_i \leq rc$.

Output

For each k_i , output one line containing the minimum difference between the largest and the smallest entry for any connected set of at least k_i entries.

Sample Input

```
5 10
0 0 3 46 0 46 0 0 12 12
0 0 13 50 49 46 11 10 10 11
0 51 51 49 99 99 89 0 0 10
0 0 48 82 70 99 0 52 13 14
51 50 50 51 70 35 70 10 14 11
6
1
5
10
12
47
50
```

Sample Output

```
0
0
3
4
89
99
```