

Sweet Tooth

You are preparing giveaway presents for your child's birthday party. For this, you have bought ridiculous amounts of various sweets. Now you want to pack a present for each of the children. It is mandatory that each child receives the same amount of sweets to prevent grievance. Thus, you do this: Out of the s sweets, you make c piles for each of the c children, and each of these piles contains $\lfloor s/c \rfloor$ sweets, and then you eat the remaining $s - c \cdot \lfloor s/c \rfloor$ yourself. With this general plan in mind, you set out to make customized piles for all children since each child prefers different sweets.

Input: The first line contains c , the number of children. The second line contains t , the number of different types of sweets. The third line contains t numbers, the i th number says how many sweets of type i you have (this means that s is the sum of the numbers in the third line).

The following c lines each contain t numbers between 1 and c which indicate the preferences of the children for each type of sweet (a lower number indicates that the sweet is higher in the priority list; priorities can be used multiple times).

Output: Compute the minimum p such that it is possible to cluster the sweets into c piles for the children and one pile for yourself such that each child receives $\lfloor s/c \rfloor$ sweets with a priority of $\leq p$.

Sample Input:

```
5
5
1 1 1 1 1
1 2 3 4 5
2 3 4 5 1
1 1 1 1 1
2 2 2 3 5
5 4 3 2 1
```

Sample Output:

```
2
```