

B. Art Union

time limit per test: 1 second

memory limit per test: 256 megabytes

input: standard input

output: standard output

A well-known art union called "Kalevich is Alive!" manufactures objects d'art (pictures). The union consists of n painters who decided to organize their work as follows.

Each painter uses only the color that was assigned to him. The colors are distinct for all painters. Let's assume that the first painter uses color 1, the second one uses color 2, and so on. Each picture will contain all these n colors. Adding the j -th color to the i -th picture takes the j -th painter t_{ij} units of time.

Order is important everywhere, so the painters' work is ordered by the following rules:

- Each picture is first painted by the first painter, then by the second one, and so on. That is, after the j -th painter finishes working on the picture, it must go to the $(j + 1)$ -th painter (if $j < n$);
- each painter works on the pictures in some order: first, he paints the first picture, then he paints the second picture and so on;
- each painter can simultaneously work on at most one picture. However, the painters don't need any time to have a rest;
- as soon as the j -th painter finishes his part of working on the picture, the picture immediately becomes available to the next painter.

Given that the painters start working at time 0, find for each picture the time when it is ready for sale.

Input

The first line of the input contains integers m, n ($1 \leq m \leq 50000$, $1 \leq n \leq 5$), where m is the number of pictures and n is the number of painters. Then follow the descriptions of the pictures, one per line. Each line contains n integers $t_{i1}, t_{i2}, \dots, t_{in}$ ($1 \leq t_{ij} \leq 1000$), where t_{ij} is the time the j -th painter needs to work on the i -th picture.

Output

Print the sequence of m integers r_1, r_2, \dots, r_m , where r_i is the moment when the n -th painter stopped working on the i -th picture.

Examples

input
5 1 1 2 3 4 5
output
1 3 6 10 15

input
4 2 2 5 3 1 5 3 10 1
output
7 8 13 21