

## 4268 Bonus Adjustment

A bonus plan is one of the most effective tools to motivate employees, therefore most companies take it seriously to decide the amount of the bonus for each employee. BonBonus is a company striving on automatic bonuses calculation software and serves many international companies. For international companies, each employee has two reporting lines, one for the regional office and the other for the functional head. And each employee's performance is based on the evaluations of the regional office and the functional head. For example, Mr. Doe is working in Taiwan office as a marketing manager, then his bonus is decided by the evaluations of Taiwan office and the marketing division in the head quarter.

Let us skip all the complicated details of the bonus calculation. The final output of the BonBonus program is a two dimensional  $m \times n$  table, where each row corresponds to a region and each column corresponds to a functional coding, and the entry in  $a_{i,j}$  is the bonus for the employee in the region  $i$  and for the function  $j$ . As usual, there is no negative bonus. Naturally, the row sums and column sums are the performance for regions and functions. Companies usually announce the performance of regions and functions while they keep the individual's bonus secret.

BonBonus was doing very well until a phone call brought the stormy bad news. A company ran the BonBonus program and happily announced the performance for each region and each function. Later, they found out that the entries in the table were not all integers, it had at most two digits after the decimal point. "Not every country has cents! And we have already announced the performance for regions and functions," shouted at the other end of the phone line. "Didn't you notice the fractional number problem when you announced them?" asked the BonBonus chief technology officer. "The row sums and column sums are all integers!" replied with the mixture of anger and amusement. "How may I help you?" explored the CTO. "You make all the entries integers! Keep the row sums and column sums intact. And moreover, the difference between the final adjusted bonus and the true bonus must be strictly less than one! And you have 24 hours to fix the problem."

That was the last sentence of the phone call. As you had expected, many companies do business in countries without cents. The phone calls rushed in and you, the guru of programming, was called in. "You fix the problem," demanded the CTO. "But, sir. Can it be fixed?" you asked. "If it cannot be fixed, then you have an even bigger problem," said the CTO. You remained silent as if you were pondering. "Fix it and you will get a bonus which will make you beyond happiness." "Okay, sir. I will try my best." You are to write a program to solve the problem above.

### Technical Specification

1. Each company has at least 2 regions and functions and at most 50 regions and functions, i.e.,  $2 \leq n, m \leq 50$ .
2. The entries for the original table are greater than or equal to zero and less than or equal to 100 with at most two digits after the decimal point, i.e.,  $0 \leq a_{i,j} \leq 100$ .

### Input

The first line of the input file contains an integer indicating the number of test cases to follow. The first line for each test case contains two integers  $m, n$  separated by a space and the following  $m$  lines contain  $n$  numbers each. And the  $j$ -th number in the  $i$ -th line is  $a_{i,j}$

**Output**

Output the adjusted bonuses for each case in order. Use a space to separate each column and a line to separate each row. Or output two characters 'no' if no such adjustment exists for that case.

**Sample Input**

```
2
2 2
1.6 1.4
1.4 1.6
2 3
14.29 23.0 21.71
10.71 8.0 15.29
```

**Sample Output**

```
2 1
1 2
15 23 21
10 8 16
```