Question **1**

Correct

Marked out of 10.00

You will be given two integers representing row count and column count of a matrix and a matrix of that size.

From that Matrix we have to find the possible 2*2 matrix where each of those matrices should follow the given rule:

· Each element of matrix should be divisible by sum of its digits.

Input Format

First Line: Row count Column Count

Matrix of size [Row count x Column Count]

Output Format

Print all the sub matrices that follow the given rule one after the other.

Sample Test Case

Input:

4 3

40 42 2

30 24 27

180 190 40

11 121 13

Output:

40 42

30 24

42 2

24 27

30 24

180 190

24 27

190 40

Note: There are four matrices of sizes 2*2 in the above output.

Print all the matrices of a particular row, and then move on to the next row.

For example:

Input	Result
4 3 40 42 2 30 24 27 180 190 40 11 121 13	40 42 30 24 42 2 24 27 30 24 180 190 24 27 190 40

Answer: (penalty regime: 0 %)

```
def sum_of_digits(n):
    return sum(int(digit) for digit in str(n))

def is_valid(matrix):
    for row in matrix:
        for num in row:
            if num%sum_of_digits(num)!=0:
                return False
    return True

def extract(matrix,rows,cols):
    res=[]
    for i in range(rows-1):
```

```
        for j in range(cols-1):
            submat=[[matrix[i][j],matrix[i][j+1]],[matrix[i+1][j],matrix[i+1][j+1]]]
            if is_valid(submat):
                res.append(submat)
    return res

def printf(matrices):
    for matrix in matrices:
        for row in matrix:
           print(" ".join(map(str,row)))



rows,cols=map(int,input().split())
matrix=[list(map(int,input().split())) for _ in range(rows)]
valid=extract(matrix,rows,cols)
printf(valid)
```

	Input	Expected	Got	
✔	4 3 40 42 2 30 24 27 180 190 40 11 121 13	40 42 30 24 42 2 24 27 30 24 180 190 24 27 190 40	40 42 30 24 42 2 24 27 30 24 180 190 24 27 190 40	✔

Passed all tests! ✔