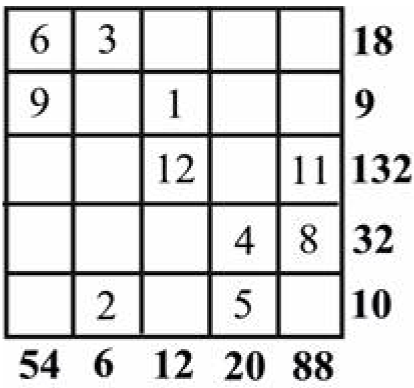
Peter takes part in a computational mathematics contest. The last round consists of the following problem.

Given an array $N \times N$. A positive integer is written besides each column and row. Let X_i be a number written near a column i, and Y_j a number written near a column j. In order to be a winner a few numbers should be put in the array. Moreover, its absolutely necessary to follow all the conditions described lower:

1. No two numbers in the array may be equal.
2. Each column and each row must contain exactly two numbers.
3. The product of the numbers of i-th column must be equal to X_i.
4. The product of the numbers of j-th row must be equal to Y_j

Figure 1 shows the example of a solution of such a problem.



6	3				**18**
9		1			**9**
		12		11	**132**
			4	8	**32**
	2		5		**10**
54	**6**	**12**	**20**	**88**	

Figure - 1

Please! Help Peter.

Input

The input file contains several test cases. The description of each test case is given below:

The first line of each test case contains N $(2 \leq N \leq 10)$. The second line contains N numbers, $X_1, X_2, \ldots, XN$ $(1 \leq X_i \leq 1000$ for all $1 \leq i \leq N)$. The third line contains N numbers $Y_1, Y_2, \ldots, Y_N$ $(1 \leq Y_i \leq 1000$ for all $1 \leq i \leq N)$.

Input is terminated by a case whose $N = 0$. This case should not be processed.

Output

The description of output for each test case is given below:

For each set of input print the $(N \times N)$ grid after placing the numbers in it. The array should be printed by lines: first N numbers of the first row, then N numbers of the second row, etc. If the cell has a number – print it; however, if it does not – print just '0'. All the numbers should be separated by blanks and/or new line symbols. Its not necessary to care about nice array. If there are many solutions, find just one of it. It's guaranteed that at least one solution will always exist.

Print a blank line after the output for each set of input.

Sample Input

```
2
2 12
3 8
3
5 8 18
2 30 12
5
54 6 12 20 88
18 9 132 32 10
10
2 12 30 56 90 132 182 240 306 380
19 36 51 64 75 84 91 96 99 200
0
```

Sample Output

```
1   3
2   4

1   2   0
5   0   6
0   4   3

6   3   0   0   0
9   0   1   0   0
0   0  12   0  11
0   0   0   4   8
0   2   0   5   0

1   0   0   0   0   0   0   0   0  19
2   0   0   0   0   0   0   0  18   0
0   3   0   0   0   0   0   0  17   0
0   4   0   0   0   0   0  16   0   0
0   0   5   0   0   0   0  15   0   0
0   0   6   0   0   0  14   0   0   0
0   0   0   7   0   0  13   0   0   0
0   0   0   8   0  12   0   0   0   0
0   0   0   0   9  11   0   0   0   0
0   0   0   0  10   0   0   0   0  20
```