

Crossbits

Crossbits are like Crosswords; instead of entering words you enter binary bits 01 in a Crossbit under certain given conditions, assuming that a solution exists. An empty Crossbit of size N is an empty grid of size $N \times N$.

Given a natural number N , consider entering N^2 binary bits in an empty Crossbit, satisfying the following conditions:

- Each square in the grid contains either a 0-bit or a 1-bit with no 1-bit in two major diagonals.
- The total number of 1-bit in each row / column is exactly equal to K , K being a given natural number less than N .
- A 0-bit has at least another adjacent 0-bit either in the same row or in the same column.
- The Crossbit represents the N^2 -bit binary number B formed by placing bits in the 1st, the 2nd, ... the N th row from left to right.

You are required to write a program that enters bits in an empty Crossbit so that the Crossbit represents the least binary number B for given N and K .

As an illustration consider the case with $N = 4$ and $K = 1$. The Crossbit shown below represents the least binary number $B = 001010000010100$ of 16 bits satisfying the specified conditions.

```
0 0 1 0
1 0 0 0
0 0 0 1
0 1 0 0
```

Input

The input may contain multiple test cases.

For each test case parameters N and K of the Crossbit are given in one line. Assume that N does not exceed 10.

The input terminates with a line containing 0 as input.

Output

For each test case, print the Crossbit in N rows; each row contains N bits with a space between two neighbouring bits. Keep a blank line after the last output line of each test case.

Example

Sample Input

```
4 1
6 2
0
```

Sample Output

```
0 0 1 0
1 0 0 0
```

0001
0100

000110
100100
000011
110000
001001
011000