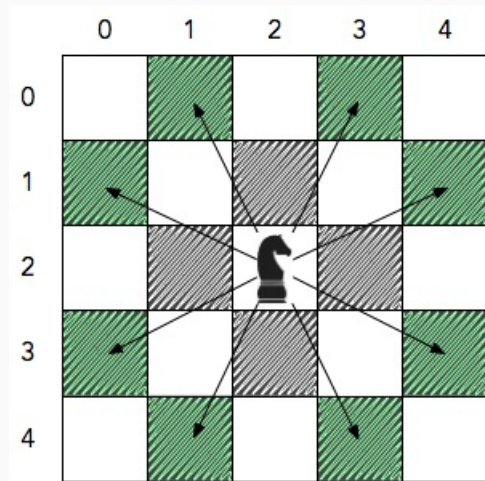


KnightL is a chess piece that moves in an L shape. We define the possible moves of ***KnightL***(a, b) as any movement from some position (x_1, y_1) to some (x_2, y_2) satisfying either of the following:

- $x_2 = x_1 \pm a$ and $y_2 = y_1 \pm b$, or
- $x_2 = x_1 \pm b$ and $y_2 = y_1 \pm a$

Note that (a, b) and (b, a) allow for the same exact set of movements. For example, the diagram below depicts the possible locations that ***KnightL***(1, 2) or ***KnightL***(2, 1) can move to from its current location at the center of a 5×5 chessboard:



Observe that for each possible movement, the Knight moves **2** units in one direction (i.e., horizontal or vertical) and **1** unit in the perpendicular direction.

Given the value of n for an $n \times n$ chessboard, answer the following question for each (a, b) pair where $1 \leq a, b < n$:

- What is the minimum number of moves it takes for ***KnightL***(a, b) to get from position $(0, 0)$ to position $(n - 1, n - 1)$? If it's not possible for the Knight to reach that destination, the answer is -1 instead.

Then print the answer for each ***KnightL***(a, b) according to the *Output Format* specified below.

Input Format

A single integer denoting n .

Constraints

- $5 \leq n \leq 25$

Output Format

Print exactly $n - 1$ lines of output in which each line i (where $1 \leq i < n$) contains $n - 1$ space-separated integers describing the minimum number of moves ***KnightL***(i, j) must make for each respective j (where $1 \leq j < n$). If some ***KnightL***(i, j) cannot reach position $(n - 1, n - 1)$, print -1 instead.

For example, if $n = 3$, we organize the answers for all the (i, j) pairs in our output like this:

```
(1,1) (1,2)
(2,1) (2,2)
```

Sample Input 0

5

Sample Output 0

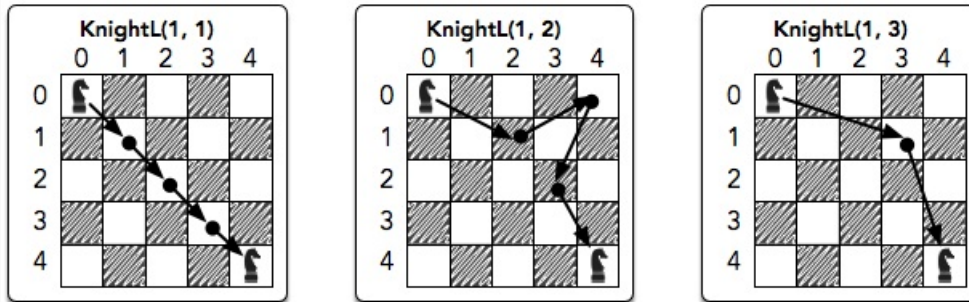
```

4 4 2 8
4 2 4 4
2 4 -1 -1
8 4 -1 1

```

Explanation 0

The diagram below depicts possible minimal paths for $\text{KnightL}(1, 1)$, $\text{KnightL}(1, 2)$, and $\text{KnightL}(1, 3)$:



One minimal path for $\text{KnightL}(1, 4)$ is:

$$(0, 0) \rightarrow (1, 4) \rightarrow (2, 0) \rightarrow (3, 4) \rightarrow (4, 0) \rightarrow (0, 1) \rightarrow (4, 2) \rightarrow (0, 3) \rightarrow (4, 4)$$

We then print 4 4 2 8 as our first line of output because $\text{KnightL}(1, 1)$ took 4 moves, $\text{KnightL}(1, 2)$ took 4 moves, $\text{KnightL}(1, 3)$ took 2 moves, and $\text{KnightL}(1, 4)$ took 8 moves.

In some of the later rows of output, it's impossible for $\text{KnightL}(i, j)$ to reach position (4, 4). For example, $\text{KnightL}(3, 3)$ can only move back and forth between (0, 0) and (3, 3) so it will never reach (4, 4).