

N-RookKnight

Filename: rookknight

Consider a new chess piece called the "rookknight". It can move either like a rook or a knight. Clearly, on an $n \times n$ chess board, we can place no more than n rookknights without having some pair attacking each other. Each possible valid arrangement of n rookknights on an $n \times n$ chessboard with no two attacking each other can be represented with a permutation.

The Problem

Consider all solutions for placing n rookknights on an $n \times n$ board. Of these solutions, determine the k^{th} one, when we sort the solutions by lexicographical ordering of their underlying permutations.

The Input

The first line of the input file will contain a number n ($1 \leq n \leq 100$) indicating the rookknight problems to solve. Each of the following n lines will contain two integers, s ($2 \leq s \leq 1000$) and k ($1 \leq k \leq 10000$), separated by spaces representing the size of one dimension of the board and the rank of the permutation desired. It is guaranteed that there will be at least k solutions for a $s \times s$ board for each query.

The Output

For each query, output the 1-based permutation that is its solution, with a space after each value, on a line by itself.

Sample Input

```
4
5 16
3 2
4 3
8 932
```

Sample Output

```
5 1 2 3 4
3 2 1
2 1 4 3
3 6 7 8 2 1 4 5
```