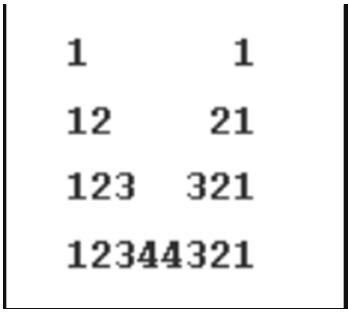
Pattern - 12: Number Crown Pattern

Problem Statement: Given an integer **N,** print the following pattern :



Here, N = 5.

Examples:

Input Format: N = 3
Result:
```
1    1
12  21
123321
```

Input Format: N = 6
Result:
```
1          1
12        21
12       321
1234    4321
12345  54321
123456654321
```

Solution

Disclaimer: *Don't jump directly to the solution, try it out yourself first.*

Problem Link

Approach:

There are 4 general rules for solving a pattern-based question :

- We always use nested loops for printing the patterns. For the outer loop, we count the number of lines/rows and loop for them.

- Next, for the inner loop, we focus on the number of columns and somehow connect them to the rows by forming a logic such that for each row we get the required number of columns to be printed.
- We print the numbers inside the inner loop.
- Observe symmetry in the pattern or check if a pattern is a combination of two or more similar patterns or not.

In this problem, we want to print a combination of a numbered pyramid and a reverse-numbered pyramid. So, as per our observation in each row, numbers are printed from 1 to the row number and then some spaces and then again numbers from 1 to the row number but in reverse order. So, the outer loop will run from 1 to N and there will be three inner loops for numbers, spaces, and then again numbers.

The first inner loop will have numbers printed from 1 to the row number, the second will print the spaces (8 spaces in row 1, 6 spaces in row 2, and so on) and then the last loop will run from row number to 1 in decreasing manner. For spaces, we can say that initially, spaces are 2*(N-1) for Row 1 where N is the total no. of rows and then the spaces decrease by 2 in each iteration till the last row is reached.

Code:

C++Java

```
class Main {

   static void pattern12(int N)
{
    // initial no. of spaces in row 1.
    int spaces = 2*(N-1);

     // Outer loop for the number of rows.
     for(int i=1;i<=N;i++){

         // for printing numbers in each row
         for(int j=1;j<=i;j++){
         System.out.print(j);
          }

         // for printing spaces in each row
```

```
            for(int j = 1;j<=spaces;j++){
            System.out.print(" ");
            }

            // for printing numbers in each row
            for(int j=i;j>=1;j--){
             System.out.print(j);
            }

            // As soon as the numbers for each iteration are printed,
we move to the
            // next row and give a line break otherwise all numbers
            // would get printed in 1 line.
            System.out.println();

            // After each iteration nos. increase by 2, thus
            // spaces will decrement by 2.
            spaces-=2;
        }
}

    public static void main(String[] args) {

        // Here, we have taken the value of N as 5.
        // We can also take input from the user.
        int N = 5;
        pattern12(N);
    }
}
```

Output

```
1         1
12       21
```

```
123       321
1234    4321
12345543 21
```