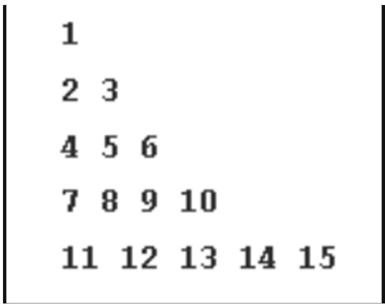
Pattern - 13: Increasing Number Triangle Pattern

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



```
1
2 3
4 5 6
7 8 9 10
11 12 13 14 15
```

Here, N = 5.

Examples:

```
Input Format: N = 3
Result:
1
2 3
4 5 6

Input Format: N = 6
Result:
1
2  3
4  5  6
7  8  9  10
11  12  13  14  15
16  17  18  19  20  21
```

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 just have to print the right-angled number pyramid but here, we also have to increase the number each time we print it. For printing, the right-angled pyramid as we know the outer loop runs for N times and the inner loop runs for i times. Now, to print an increasing number pyramid we just have to increment the number inside the inner loop so that after printing the number each time it increases by 1.

Code:

C++Java

```
class Main {

   static void pattern13(int N)
{
     // starting number.
     int num =1;

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

          // Inner loop will loop for i times and
          // print numbers increasing by 1 each time.
          for(int j=1;j<=i;j++){
              System.out.print(num + " ");
              num =num+1;
          }
          // 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();

      }
}

    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;
        pattern13(N);
    }
}
```

Output

```
1
2  3
4  5  6
7  8  9  10
11  12  13  14  15
```