Pattern - 8: Inverted Star Pyramid

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

```
********
 ******
  *****
   ***
    *
```

Here, N = 5.

Examples:

Input Format: N = 3
Result:
```
*****
 ***
  *
```
Input Format: N = 6
Result:
```
**********
 ********
  *******
   *****
    ***
     *
```

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 '*' 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 particular pattern, we run the outer loop for N times as we have to print N rows as usual. Now, the question arises what will be the logic behind the inner loop?

Similar to the last pattern, we can clearly observe that for each row there are some spaces that get printed then some stars, and then again some spaces giving it an inverse pyramidal look. For eg: In the first row (i=0) there are 0 spaces, 9 stars, then again 0 spaces. In the second row (i=1) there is 1 space, 7 stars, then again 1 space so we can say that there are i spaces, 2*N - (2*i+1) stars, and then again i space for each row where i is the row index. We thus simply run 3 inner loops, first for printing the spaces, then the stars, and then the spaces again.

Code:

C++Java

```cpp
#include <bits/stdc++.h>
using namespace std;


void pattern8(int N)
{
    // This is the outer loop which will loop for the rows.
    for (int i = 0; i < N; i++)
    {
        // For printing the spaces before stars in each row
        for (int j =0; j<i; j++)
        {
            cout <<" ";
        }


        // For printing the stars in each row
        for(int j=0;j< 2*N -(2*i +1);j++){
```

```cpp
            cout<<"*";
        }

        // For printing the spaces after the stars in each row
        for (int j =0; j<i; j++)
        {
            cout <<" ";
        }

        // As soon as the stars for each iteration are printed, we move to the
        // next row and give a line break otherwise all stars
        // would get printed in 1 line.
        cout << endl;
    }
}

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

    pattern8(N);

    return 0;
}
```

Output

```
*********
 *******
  *****
   ***
    *
```