

# Day 11/180: Pattern Printing- Super Hard

First Pattern: Take an input n from the user, and create a pattern like below, for n=5, we have output like this.

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

Solution:

```
#include <iostream>
using namespace std;

int main() {
    int n;
    cin >> n;

    for (int i = 1; i <= n; i++) {
        // Print spaces to align the asterisks to the right
        for (int j = 1; j <= n - i; j++) {
            cout << " ";
        }

        // Print asterisks for this row
        for (int k = 1; k <= i; k++) {
            cout << "*" // Add extra spaces for spacing
        }

        cout << endl;
    }
    return 0;
}
```

Second Pattern: Take an input n from the user, and create a pattern like below, for n=5, we have output like this.

```
    1
  1 2 3
1 2 3 4 5
1 2 3 4 5 6 7
1 2 3 4 5 6 7 8 9
```

Solution:

```
#include<iostream>
using namespace std;

int main()
{
    int n;
    cin >> n; // Input the number of rows for the pattern
    for (int i = 1; i <= n; i++){
        // Loop to print spaces before the numbers
        for (int j = 1; j <= n - i; j++){
            cout << "  "; // Print two spaces
        }
        // Loop to print numbers in increasing order
        for (int j = 1; j <= 2 * i - 1; j++){
            cout << j << " "; // Print the current value of
            'j' followed by a space
        }
        cout << endl; // Move to the next line after each
        row is printed
    }
    return 0;
}
```

Third Pattern: Take an input n from the user, and create a pattern like below, for n=5, we have output like this

```
A
A B A
A B C B A
A B C D C B A
A B C D E D C B A
```

Solution:

```
#include<iostream>
using namespace std;

int main(){
    int n;
    cin >> n; // Input the number of rows for the pattern
    for (int i = 1; i <= n; i++){
        // Loop to print spaces before the characters
        for (int j = 1; j <= n - i; j++){
            cout << " "; // Print two spaces
        }
        // Loop to print characters in increasing order
        for (int j = 1; j <= i; j++){
            cout << (char)('A' + j - 1) << " "; // Print the
character ('A' + j - 1) followed by a space
        }
        // Loop to print characters in decreasing order
(excluding the last character)
        for (int j = i - 1; j >= 1; j--){
            cout << (char)('A' + j - 1) << " "; // Print the
character ('A' + j - 1) followed by a space
        }
        cout << endl; // Move to the next line after each row is
printed
    }
    return 0;
}
```

Fourth Pattern: Take an input n from the user, and create a pattern like below, for n=5, we have output like this.

```

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

Solution:

```
#include<iostream>
using namespace std;

int main()
{
    int n;
    cin >> n; // Input the number of rows for the pattern
    //Upper Half
    for (int i = 1; i <= n; i++){
        // Loop to print spaces before the numbers
        for (int j = 1; j <= n - i; j++){
            cout << " "; // Print two spaces
        }
        // Print asterisks for this row
        for (int k = 1; k <= i; k++) {
            cout << "* "; // Add extra spaces for spacing
        }
        cout << endl; // Move to the next line after each
        row is printed
    }
}
```

```

    }
    //Lower Half
    for (int i = n; i >= 1 ; i--){
        // Loop to print spaces before the numbers
        for (int j = 1; j <= n - i; j++){
            cout << " "; // Print two spaces
        }
        // Print asterisks for this row
        for (int k = 1; k <= i; k++) {
            cout << "* "; // Add extra spaces for spacing
        }
        cout << endl; // Move to the next line after each
row is printed
    }
    return 0;
}

```