

## PRINT 1 TO N

Given **n** (number of rows), print the following pattern:

**n = 5**

```
1  
2  
3     for(int i=1; i<=n ;i++)  
4     {  
5         cout << i << endl;  
6     }
```

**n = 6**

```
1  
2  
3  
4  
5  
6
```

**n = 7**

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

## 2 STARS

Given **n** (number of rows), print the following pattern:

**n = 5**

```
**  
**     for(int i=1; i<=n ;i++)  
**     {  
**         cout << "**<< endl;  
**     }
```

**n = 6**

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

**n = 7**

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

# 4 STARS

Given **n** (number of rows), print the following pattern:

**n = 5**

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

**n = 6**

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

**n = 7**

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

```
1 #include <bits/stdc++.h>  
2 using namespace std;  
3  
4 int main()  
5 {  
6     int n;  
7     cin >> n;  
8  
9     for(int i = 1; i <= n; i++)  
10    {  
11        // we have to print star 'n' times  
12        for(int j = 1; j <= n; j++)  
13        {  
14            cout << "*";  
15        }  
16        cout << endl;  
17    }  
18}  
19
```

5

Output

Status : Successfully executed

Time:  
0.0000 secs

Memory:  
3.52 Mb

Sample Input

5

Your Output

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

# M STARS (RECTANGLE)

Given  $n$  (no. of rows) and  $m$  (no. of cols) , print the following pattern:

$n = 5, m = 7$

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

$n = 6, m = 3$

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

$n = 7, m = 4$

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

```
1 #include <bits/stdc++.h>  
2 using namespace std;  
3  
4 int main()  
5 {  
6     int n;  
7     cin >> n;  
8  
9     int m;  
10    cin >> m;  
11  
12    for(int i = 1; i <= n; i++)  
13    {  
14        // cout << "****" << endl;  
15        // we have to print star 'm' times  
16        for(int j = 1; j <= m; j++)  
17        {  
18            cout << "*";  
19        }  
20        cout << endl;  
21    }  
22}  
23
```

10 3

Output

Status : Successfully execut

Time: 0.0000 secs | Memory: 3.568 M

Sample Input

10

Your Output

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

# SQUARE

Given **n** (no. of rows and cols), print the following pattern:

**n = 5**

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

**n = 6**

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

**n = 7**

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

```
1 #include <bits/stdc++.h>
2 using namespace std;
3
4 int main()
5 {
6     int n;
7     cin >> n;
8
9     for(int i = 1; i <= n; i++)
10    {
11        // we have to print star 'n' times
12        for(int j = 1; j <= n; j++)
13        {
14            cout << "*";
15        }
16        cout << endl;
17    }
18 }
19
```

5

Output

Status : Successfully executed

Time: 0.0000 secs | Memory: 3.52 Mb

Sample Input

5

Your Output

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

# PYRAMID

Given  $n$  (no. of rows), print the following pattern:

$n = 5$

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

$n = 6$

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

$n = 7$

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

# PYRAMID

Given  $n$  (no. of rows), print the following pattern:

$n = 5$

```
•  
..  
...  
...  
....
```

$n = 6$

```
•  
..  
...  
...  
...  
....
```

$n = 7$

```
•  
..  
...  
...  
...  
....  
....
```

$i$

1

2

3

4

5

6

7

# stars

1  
2  
3

4  
5

6  
7

```
for( i=1; i<=n; i++ )  
{  
    for( j=1; j<=i; j++ )  
        cout << "*";  
    cout << endl;  
}
```

(i)  $\Rightarrow$  (i)

# INVERTED PYRAMID

Given **n** (no. of rows), print the following pattern:

**n = 5**

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

**n = 6**

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

**n = 7**

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

```
#include <bits/stdc++.h>  
using namespace std;  
  
int main()  
{  
    int n;  
    cin >> n;  
  
    for(int i = n; i >= 1; i--)  
    {  
        // we have to print star 'n' times  
        for(int j = 1; j <= i; j++)  
        {  
            cout << "*";  
        }  
        cout << endl;  
    }  
}
```

5

Output

Status : Successfully executed

Time: 0.0000 secs | Memory: 3.392 Mb

Sample Input

5

Your Output

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

# HOLLOW SQUARE

Boundary will exist

Given  $n$  (no. of rows and cols), print the following pattern:

$n = 5$

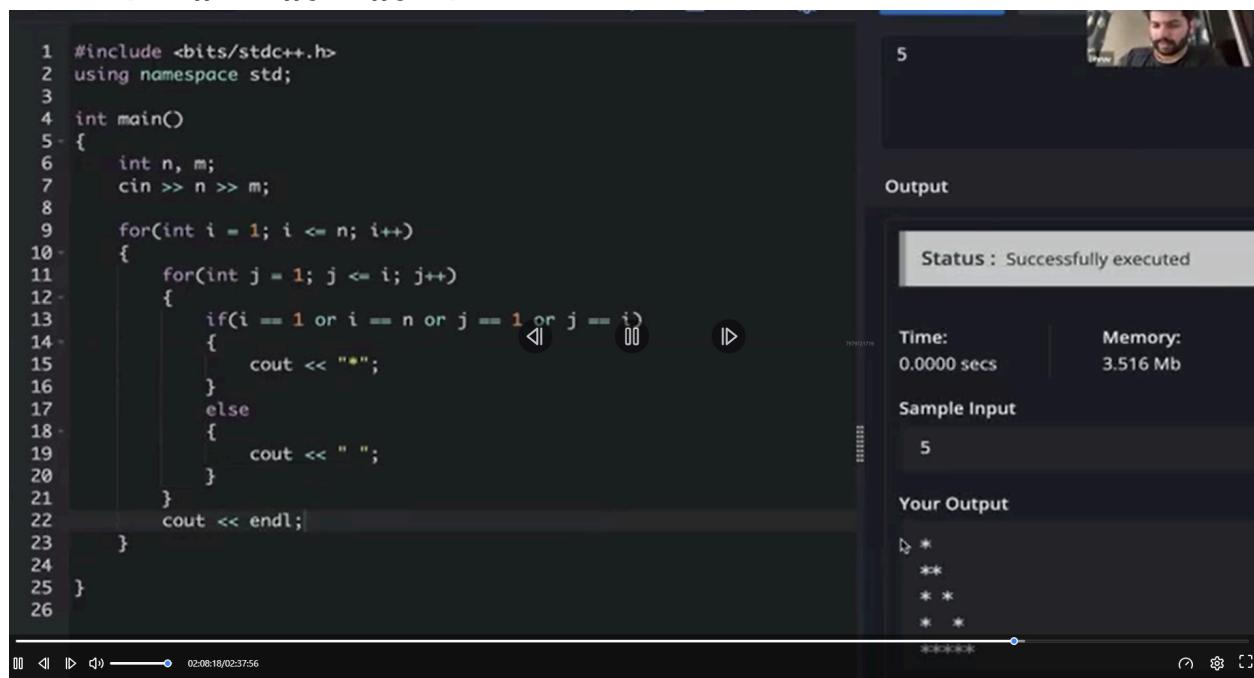
```
i=1 * * * *  
* * * *  
* * * *  
* * * *  
i=5 * * * * *
```

$n = 6$

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

## Note:

1. To solve the Hollow square problem Focus on boundary points i.e.  
 $\text{if}(i=1 \text{ || } i=n \text{ || } j=1 \text{ || } j=n)$



The screenshot shows a code editor with the following C++ code:

```
1 #include <bits/stdc++.h>
2 using namespace std;
3
4 int main()
5 {
6     int n, m;
7     cin >> n >> m;
8
9     for(int i = 1; i <= n; i++)
10    {
11        for(int j = 1; j <= i; j++)
12        {
13            if(i == 1 or i == n or j == 1 or j == i)
14            {
15                cout << "*";
16            }
17            else
18            {
19                cout << " ";
20            }
21        }
22        cout << endl;
23    }
24
25 }
```

The code is part of a live coding session, as evidenced by the video camera icon in the top right corner of the interface.

The execution results are displayed in the right panel:

- Status: Successfully executed
- Time: 0.0000 secs | Memory: 3.516 Mb
- Sample Input: 5
- Your Output:  
\*  
\*\*  
\* \*  
\* \*

The screenshot shows a C++ code editor interface. The code in the editor is as follows:

```
1 #include <bits/stdc++.h>
2 using namespace std;
3
4 int main()
5 {
6     int n, m;
7     cin >> n >> m;
8
9     for(int i = n; i >= 1; i--)
10    {
11        for(int j = 1; j <= i; j++)
12        {
13            if(i == 1 || i == n || j == 1 || j == i)
14            {
15                cout << "*";
16            }
17            else
18            {
19                cout << " ";
20            }
21        }
22        cout << endl;
23    }
24
25 }
```

The output window shows the status "Status : Successfully executed" and the output "10". The sample input was "10" and the output produced is:

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

## NUMBERED RECTANGLE

Given **n** (no. of rows) and **m** (no. of cols) , print the following pattern:

**n = 5, m = 7**

1111111	x x x x x x x
2222222	x x x x x x x
3333333	x x x x x x x
4444444	x x x x x x x
5555555	x x x x x x x

1) draw the star pattern

2) replace \* → given elem

**n = 6, m = 3**

111
222
333
444
555
666

**n = 7, m = 4**

1 1 1 1
2 2 2 2
3 3 3 3
4 4 4 4
5 5 5 5
6 6 6 6
7 7 7 7