

<b>Status</b>	Finished
<b>Started</b>	Monday, 3 November 2025, 3:35 PM
<b>Completed</b>	Monday, 3 November 2025, 3:45 PM
<b>Duration</b>	9 mins 50 secs

Question **1**

Correct

The number of rows N is passed as the input. The program must print the half pyramid using asterisk \*.

**Input Format:**

The first line contains N.

**Output Format:**

N lines representing the half pyramid pattern using \* (A single space is used to separate the \*)

**Boundary Conditions:**

$2 \leq N \leq 100$

**Example Input/Output 1:**

Input:

5

Output:

```
*
```

```
**
```

```
***
```

```
****
```

```
*****
```

**Example Input/Output 2:**

Input:

3

Output:

```
*
```

```
**
```

```
***
```

**For example:**

Input	Result
5	* * * * * * * * * * * * * * *
3	* * * * * *

**Answer:** (penalty regime: 0 %)

```

1 #include<stdio.h>
2 int main()
3 {
4     int n;
5     scanf("%d", &n);
6     for(int i = 1;i <= n; i++){
7         for(int j = 1; j <= i; j++)
8             printf("* ");
9         printf("\n");
10    }
11    return 0;
12 }
```

	Input	Expected	Got	
✓	5	* * * * * * * * * * * * * * *	*	✓
✓	3	* * * * * *	*	✓

Passed all tests! 

**Question 2**

Correct

The number of rows N is passed as the input. The program must print the half pyramid using the numbers from 1 to N.

**Input Format:**

The first line contains N.

**Output Format:**

N lines representing the half pyramid pattern using the numbers from 1 to N. (A single space is used to separate the numbers)

**Boundary Conditions:**

$2 \leq N \leq 100$

**Example Input/Output 1:**

Input:

5

Output:

1  
1 2  
1 2 3  
1 2 3 4  
1 2 3 4 5

**Example Input/Output 2:**

Input:

3

Output:

1  
1 2

1 2 3

**For example:**

Input	Result
5	1 1 2 1 2 3 1 2 3 4 1 2 3 4 5
3	1 1 2 1 2 3

**Answer:** (penalty regime: 0 %)

```

1 #include<stdio.h>
2 int main()
3 {
4     int n;
5     scanf("%d", &n);
6     for(int i = 1; i <= n; i++){
7         for(int j = 1; j <=i; j++)
8             printf("%d ", j);
9         printf("\n");
10    }
11 }
12 }
```

	Input	Expected	Got	
✓	5	1 1 2 1 2 3 1 2 3 4 1 2 3 4 5	1 1 2 1 2 3 1 2 3 4 1 2 3 4 5	✓
✓	3	1 1 2 1 2 3	1 1 2 1 2 3	✓

Passed all tests! 