



Last Updated: Sep 22, 2023

Easy

Star Patterns Program in C

Author
Vaibhav Mani

0 upvote

Share

Introduction

Star Patterns Program in C involve printing various patterns made of asterisks (*) using loops. These programs are common in learning C programming as they help grasp the concept of nested loops and control structures.. These are the most triggering questions because it opens your mind and helps you develop your logical thinking. We will be doing this question using [C Language](#).



Star Patterns Program in C

**Set your learning plan**
Important for focused learningPrepare for tech interviews ☐Learn competitive programming ☐Learn and practise coding ☐

Already setup?

Next

1/2

What is Star Pattern in C?

In C, an asterisk pattern refers to a series of asterisks (*) arranged in a particular pattern. These patterns are often printed on consoles and terminals and can be used for various decorative and educational purposes. Star patterns are a popular programming exercise for beginners because they help improve your logic and looping skills.



There are many types of star patterns that can be created using loops in C, such as Square pattern, triangle pattern, diamond pattern, etc.

Each pattern requires a different set of loops and logic to achieve the desired output.

Square Star Pattern

This code asks the user to enter the number of rows (n) in the square pattern. Two nested loops are used to print the stars. Iterate the outer loop n times to create a row, then iterate the inner loop n times to create a star in each row.

The `printf("* ")` statement prints an asterisk and a space for each iteration, creating a pattern of squares.

Code

C

```
#include <stdio.h>
int main() {
    int n;
    printf("Enter the number of rows: ");
    scanf("%d", &n);

    for (int i = 0; i < n; i++) {
        for (int j = 0; j < n; j++) {
            printf("* ");
        }
        printf("\n");
    }
    return 0;
}
```

 You can also try this code with [Online C Compiler](#)

[Run Code](#)

Output

```
/tmp/DXSZDRP8kK.o
Enter the number of rows: 5
* * * * *
* * * * *
* * * * *
* * * * *
* * * * *
```

Set your learning plan

Important for focused learning



Prepare for tech interviews



Learn competitive programming



Learn and practise coding



[Already setup?](#)

[Next](#)

1/2

Hollow Square Star Pattern



Similar to the square pattern, the code asks the user to enter the number of rows (n) for the hollow square pattern. The code prints asterisks and spaces using two nested loops. It iterates the outer loop n times to create a row, then iterates the inner loop n times to create an asterisk and a space on each row.

The condition if `(i == 1 || i == n || j == 1 || j == n)` checks if the current position is on the boundary of the square. In that case the code prints an asterisk (*). Otherwise, two spaces () are printed to create a hollow effect inside the square.


Code

C

```
#include <stdio.h>
int main() {
    int n;
    printf("Enter the number of rows: ");
    scanf("%d", &n);

    for (int i = 1; i <= n; i++) {
        for (int j = 1; j <= n; j++) {
            if (i == 1 || i == n || j == 1 || j == n) {
                printf("* ");
            } else {
                printf(" ");
            }
        }
        printf("\n");
    }

    return 0;
}
```

 You can also try this code with [Online C Compiler](#)

[Run Code](#)

Output

```
/tmp/DXSZDRP8kK.o
Enter the number of rows: 5
* * * * *
*           *
*           *
*           *
*           *
* * * * *
```

Hollow Square Pattern with
Diagonal



Set your learning plan

Important for focused learning



Prepare for tech
interviews



Learn competitive
programming



Learn and practise
coding



[Already setup?](#)

[Next](#)

1/2

Similar to the hollow square pattern, the code asks the user to enter the number of rows (n) for the diagonal hollow square pattern. It prints asterisks and spaces using two nested loops. It iterates the outer loop n times to create a row, then iterates the inner loop n times to create an asterisk and a space on each row.

The condition if $(i == 1 \parallel i == n \parallel j == 1 \parallel j == n \parallel i == j \parallel j == (n - i + 1))$ if the current position is on the border of the square or diagonally from top left to bottom right and top right to bottom

Code

C

```
#include <stdio.h>
int main() {
    int n;
    printf("Enter the number of rows: ");
    scanf("%d", &n);

    for (int i = 1; i <= n; i++) {
        for (int j = 1; j <= n; j++) {
            if (i == 1 || i == n || j == 1 || j == n || i == j ||
                j == (n - i + 1)) {
                printf("* ");
            } else {
                printf(" ");
            }
        }

        printf("\n");
    }
    return 0;
}
```

 You can also try this code with [Online C Compiler](#)

[Run Code](#)

Output

```
/tmp/DXSZDRP8kK.o
Enter the number of rows: 5
* * * * *
* *   * *
*   *   *
* *   * *
* * * * *
```

Set your learning plan
Important for focused learning



Prepare for tech interviews



Learn competitive programming



Learn and practise coding



[Already setup?](#)

[Next](#)

1/2

Rhombus Star Pattern



This code asks the user to enter the number of rows (n) in the diamond pattern. Then it prints spaces and asterisks using three nested loops. It creates rows by repeating the outermost loop n times.

The first inner loop (**controlled by the space variable**) outputs the space to move the star so that it forms a diamond shape.

The second inner loop (**controlled by variable j**) outputs 'n' stars on each line. The `printf("*")` statement prints an asterisk and a space at each iteration, creating a diamond pattern.

Code

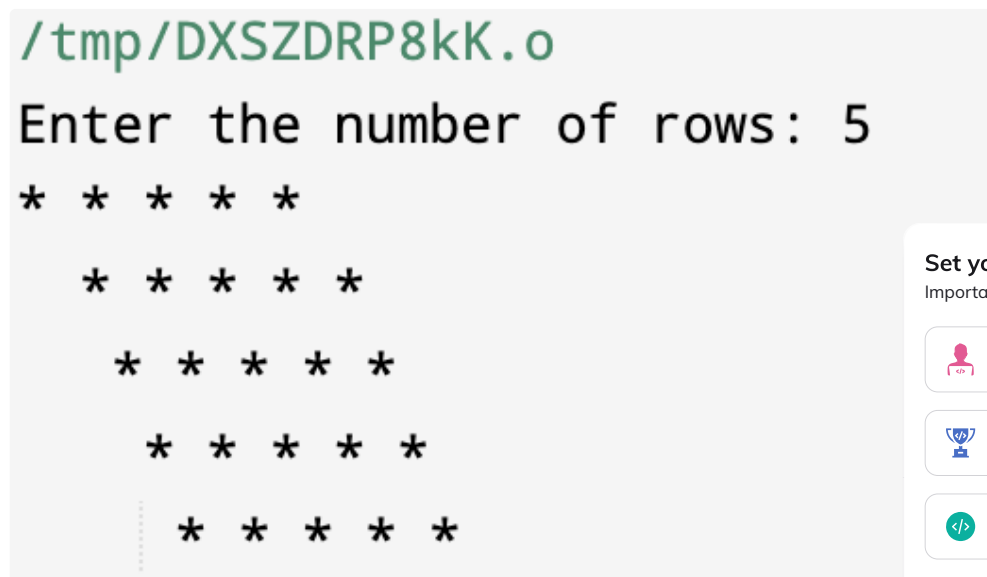
C

```
#include <stdio.h>
int main() {
    int n;
    printf("Enter the number of rows: ");
    scanf("%d", &n);
    for (int i = 1; i <= n; i++) {
        for (int space = 1; space <= i; space++) {
            printf(" ");
        }
        for (int j = 1; j <= n; j++) {
            printf("* ");
        }
        printf("\n");
    }
    return 0;
}
```

 You can also try this code with [Online C Compiler](#)

[Run Code](#)


Output





Hollow Rhombus Star Pattern

Similar to the diamond pattern, the code asks the user to enter the number of rows (n) for the hollow diamond pattern.

Set your learning plan
Important for focused learning

 Prepare for tech interviews ☐

 Learn competitive programming ☐

 Learn and practise coding ☐

[Already setup?](#) [Next](#)

1/2

The condition `if (i == 1 || i == n || j == 1 || j == n)` checks if the current position is on the border of the diamond. In that case the code prints an asterisk (*). Otherwise, two spaces (" ") are printed to create a hollow effect inside the diamond.

Code

C

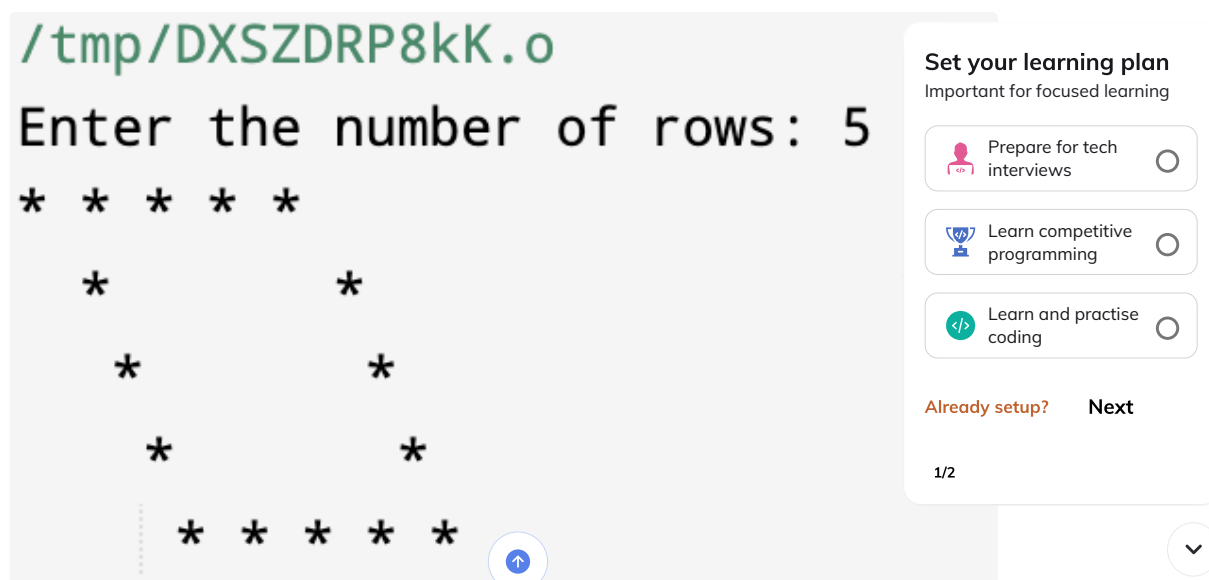
```
#include <stdio.h>

int main() {
    int n;
    printf("Enter the number of rows: ");
    scanf("%d", &n);
    for (int i = 1; i <= n; i++) {
        for (int space = 1; space <= i; space++) {
            printf(" ");
        }
        for (int j = 1; j <= n; j++) {
            if (i == 1 || i == n || j == 1 || j == n) {
                printf("* ");
            } else {
                printf(" ");
            }
        }
        printf("\n");
    }
    return 0;
}
```

 You can also try this code with [Online C Compiler](#)

Run Code

Output



Mirrored Rhombus Star Pattern

A mirror diamond star pattern is created by entering 'n' for the number of lines in the pattern.

A pattern is created using two nested loops. The outer loop goes from 1 to 'n', representing each row of the diamond.

For each line, the inner loop "n - i" prints a space to indent the line properly. After printing a space, the inner loop prints the '*' character n times to create a series of asterisks.

When the inner loop for a given row is completed, a newline character is printed and the process repeats until 'n' rows of mirrored diamond patterns are produced.

Code

C

```
#include <stdio.h>
int main() {
    int n;
    printf("Enter the number of rows: ");
    scanf("%d", &n);
    for (int i = 1; i <= n; i++) {
        for (int space = 1; space < i; space++) {
            printf(" ");
        }
        for (int j = 1; j <= n; j++) {
            printf("* ");
        }
        printf("\n");
    }
    return 0;
}
```

</> You can also try this code with [Online C Compiler](#)


[Run Code](#)

Output


```
/tmp/DXSZDRP8kK.o
Enter the number of rows: 5
* * * * *
 * * * * *
  * * * * *
   * * * * *
    * * * * *
```

Set your learning plan


Important for focused learning

 Prepare for tech interviews

☐

 Learn competitive programming



☐

 Learn and practise coding

☐

Already setup? [Next](#)

1/2



Hollow Mirrored Rhombus Star Pattern

The hollow mirrored rhombus star pattern also takes an input `n` for the number of rows. Using two nested loops, the outer loop iterates from 1 to "n" to represent each row.


It prints "n - i" spaces to indent lines. For the first and last row (**that is, 'i == 1' or 'i == n'**), the inner loop outputs 'n' times to create a diamond-shaped fixed border.

For other rows, the inner loop prints " " only in the first and last columns, creating a null effect. Remaining characters between the first and last columns are replaced with spaces. This process is repeated until 'n' rows of hollow mirror diamond patterns are formed.

Code

C

```
#include <stdio.h>
int main() {
    int n;
    printf("Enter the number of rows: ");
    scanf("%d", &n);
    for (int i = 1; i <= n; i++) {
        for (int space = 1; space < i; space++) {
            printf(" ");
        }
        for (int j = 1; j <= n; j++) {
            if (i == 1 || i == n || j == 1 || j == n) {
                printf("* ");
            } else {
                printf(" ");
            }
        }
        printf("\n");
    }
    return 0;
}
```

 You can also try this code with [Online C Compiler](#)

[Run Code](#)

Output

Set your learning plan

Important for focused learning



Prepare for tech interviews



Learn competitive programming



Learn and practise coding



[Already setup?](#)

[Next](#)

1/2




```
/tmp/DXSZDRP8kK.o
Enter the number of rows: 5
* * * * *
  *       *
    *     *
      *   *
        * *
          * * * * *
```

Right Triangle Star Pattern

The right triangle star pattern uses an input 'n' for the number of rows. It uses two nested loops to print each row.

The outer loop runs from 1 to 'n', representing each row of the triangle. For each row, the inner loop prints '*' 'i' times, where 'i' is the current row number.

This results in 'i' stars in each row, starting from 1 star in the first row and increasing by one for each subsequent row. This process is repeated until 'n' rows of the right triangle pattern are generated.

Code

C

```
#include <stdio.h>
int main() {
    int n;
    printf("Enter the number of rows: ");
    scanf("%d", &n);
    for (int i = 1; i <= n; i++) {
        for (int j = 1; j <= i; j++) {
            printf("* ");
        }
        printf("\n");
    }
    return 0;
}
```

</> You can also try this code with [Online C Compiler](#)

Output

Set your learning plan

Important for focused learning



Prepare for tech interviews



Learn competitive programming



Learn and practise coding



Already setup?

Next

1/2



```
/tmp/cDYsHKDtZB.o
```

```
Enter the number of rows: 5
```

```
*
```

```
* *
```

```
* * *
```

```
* * * *
```

```
* * * * *
```

Hollow Right Triangle Star Pattern

Entering 'n' for the number of rows creates a hollow star pattern of right triangles. Print each line using two nested loops. The outer loop runs from 1 to 'n', representing each row of the right triangle.

For the first and last row (that is, 'i == 1' or 'i == n'), the inner loop outputs 'i' times to create a series of stars that form the edges of the triangle. For the other rows, the inner loop prints only "" in the first and last columns, replacing the characters in between with spaces to create a hollow effect in the triangle.

This process is repeated until a hollow right triangle pattern with n rows is created.

Code

C

```
#include <stdio.h>
int main() {
    int n;
    printf("Enter the number of rows: ");
    scanf("%d", &n);
    for (int i = 1; i <= n; i++) {
        for (int j = 1; j <= i; j++) {
            if (j == 1 || j == i || i == n) {
                printf("* ");
            } else {
                printf(" ");
            }
        }
        printf("\n");
    }
    return 0;
}
```

</> You can also try this code with [Online C Compiler](#)

Set your learning plan

Important for focused learning



Prepare for tech interviews



Learn competitive programming



Learn and practise coding



Already setup?

Next

Run

1/2

Output



```
/tmp/DXSZDRP8kK.o
Enter the number of rows: 5
*
* *
*   *
*     *
* * * * *
```

Mirrored Right Triangle Star Pattern

The mirrored right triangle star pattern takes an input 'n' for the number of rows. It uses two nested loops, with the outer loop running from 1 to 'n', representing each row. For each row, the inner loop prints 'n - i' spaces to indent the row accordingly.

Then, the inner loop prints '*' 'i' times to create the right-angled triangle in each row. This process is repeated until 'n' rows of the mirrored right triangle pattern are generated.

Code

C

```
#include <stdio.h>
int main() {
    int n;
    printf("Enter the number of rows: ");
    scanf("%d", &n);

    for (int i = 1; i <= n; i++) {
        for (int space = i; space <= n; space++) {
            printf(" ");
        }
        for (int j = 1; j <= i; j++) {
            printf("* ");
        }
        printf("\n");
    }
    return 0;
}
```

</> You can also try this code with [Online C Compiler](#)

Output

Set your learning plan
Important for focused learning



Prepare for tech
interviews



Learn competitive
programming



Learn and practise
coding



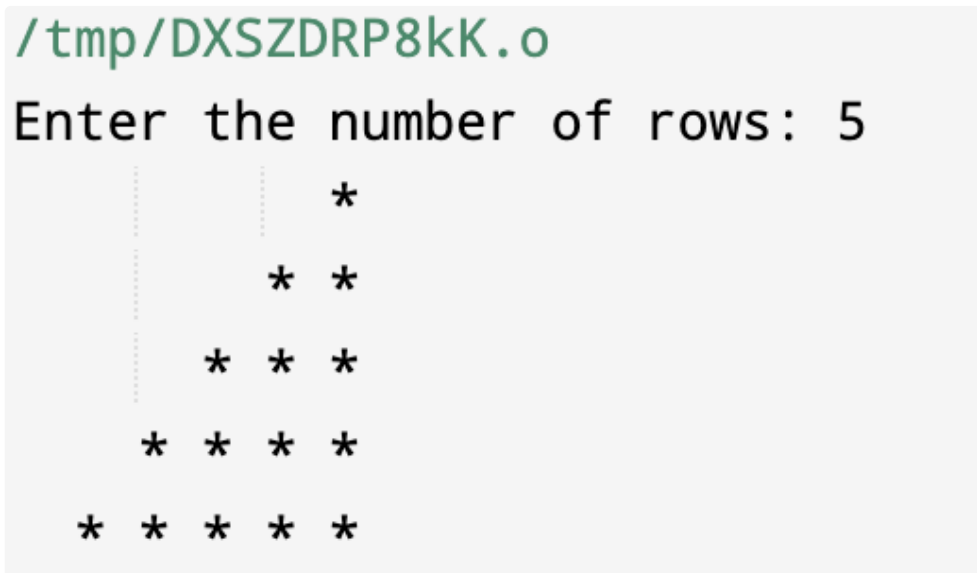
Run

Already setup?

Next

1/2





Hollow Mirrored Right Triangle Star Pattern

Hollow mirrored right triangle star patterns also require 'n' to be entered as the number of rows. Using two nested loops, the outer loop iterates from 1 to "n" to represent each row.

For the first and last row (that is, 'i == 1' or 'i == n'), the inner loop outputs 'i' times to create a series of stars that form the edges of the triangle. For the other rows, the inner loop prints " " only in the first and last columns, with a space in between, creating the effect of a hollow right triangle.

This process is repeated until 'n' rows of hollow mirrored right triangle patterns are formed.

Code

C

```

#include <stdio.h>
int main() {
    int n;
    printf("Enter the number of rows: ");
    scanf("%d", &n);

    for (int i = 1; i <= n; i++) {
        for (int space = 1; space <= n - i; space++) {
            printf(" ");
        }
        for (int j = 1; j <= i; j++) {
            if (j == 1 || j == i || i == n) {
                printf("* ");
            } else {
                printf(" ");
            }
        }

        printf("\n");
    }
    return 0;
}

```

Set your learning plan

Important for focused learning



Prepare for tech interviews



Learn competitive programming



Learn and practise coding



Already setup? **Next**

1/2



</> You can also try this code with [Online C Compiler](#)

Run Code



Output

```
/tmp/DXSZDRP8kK.o
Enter the number of rows: 5

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

Inverted Right Triangle Star Pattern

An inverted right triangle star pattern is formed by entering 'n' for the number of rows.

It draws each row of an inverted right triangle using a single loop that iterates from 'n' to 1. For each row, the loop prints '*' 'i' times. 'i' is the current line number.

The result is that the first row will have "n" stars, the second row will have "n-1" stars, and so on, with a maximum of 1 star in the last row.

This process is repeated until n rows of inverted right triangle patterns are generated.

Code

C

```
#include <stdio.h>
int main() {
    int n;
    printf("Enter the number of rows: ");
    scanf("%d", &n);

    for (int i = n; i >= 1; i--) {
        for (int j = 1; j <= i; j++) {
            printf("* ");
        }
        printf("\n");
    }
    return 0;
}
```

</> You can also try this code with [Online C Compiler](#)

Set your learning plan

Important for focused learning



Prepare for tech interviews



Learn competitive programming



Learn and practise coding



Already setup?

Next

Run

1/2

Output



```
/tmp/DXSZDRP8kK.o
```

```
Enter the number of rows: 5
```

```
* * * * *
```

```
* * * *
```

```
* * *
```

```
* *
```

```
*
```

Hollow Inverted Right Triangle Star Pattern

For the hollow inverted right triangle star pattern, you must enter 'n' as the number of rows. It draws each row of an inverted right triangle using a single loop that iterates from 'n' to 1.

For the first and last row, the loop outputs 'i' times to create triangle edges.

For the other lines, the loop prints " " only in the first and last columns, replacing the characters in between with spaces to create the effect of whitespace.

This process is repeated until a hollow inverted right triangle pattern with n rows is created.

Code

C

```
#include <stdio.h>
int main() {
    int n;
    printf("Enter the number of rows: ");
    scanf("%d", &n);
    for (int i = n; i >= 1; i--) {
        for (int j = 1; j <= i; j++) {
            if (j == 1 || j == i || i == n) {
                printf("* ");
            } else {
                printf(" ");
            }
        }
        printf("\n");
    }
    return 0;
}
```

</> You can also try this code with [Online C Compiler](#)

Set your learning plan

Important for focused learning



Prepare for tech interviews



Learn competitive programming



Learn and practise coding



Already setup?

Next

Run 1/2

Output



```
/tmp/tIWJN4odCy.o
Enter the number of rows: 5
* * * * *
*       *
*     *
*  *
*
```

Inverted Mirrored Right Triangle Star Pattern

You must also enter 'n' as the number of rows for the reverse mirrored right triangle star pattern. It draws each row of an inverted right triangle using a single loop that iterates from 'n' to 1.

For each line, the "n - i" loop prints a space to properly indent the line. Then print '*' 'i' times to create a right triangle on each line.

This process is repeated until 'n' rows of reverse mirrored right triangle patterns are generated.

Code

C

```
#include <stdio.h>
int main() {
    int n;
    printf("Enter the number of rows: ");
    scanf("%d", &n);
    for (int i = n; i >= 1; i--) {
        for (int space = 1; space <= n - i; space++) {
            printf(" ");
        }
        for (int j = 1; j <= i; j++) {
            printf("* ");
        }
        printf("\n");
    }
    return 0;
}
```

<> You can also try this code with [Online C Compiler](#)

Output

Set your learning plan

Important for focused learning



Prepare for tech interviews



Learn competitive programming



Learn and practise coding

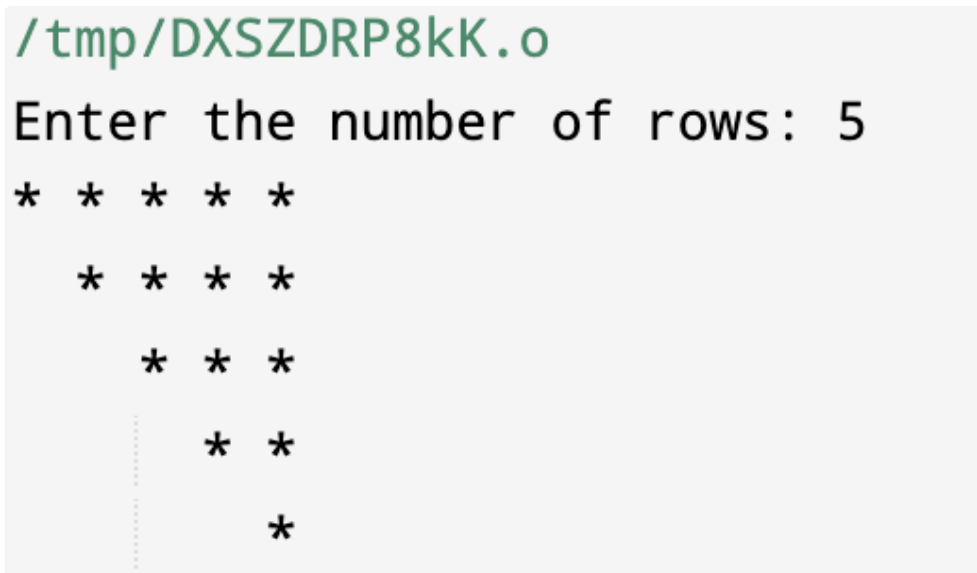


Run

Already setup?

Next

1/2



Hollow Inverted Mirrored Right Triangle Star Pattern

Entering 'n' for the number of rows creates a star pattern of hollow mirrored right triangles. It draws each row of an inverted right triangle using a single loop that iterates from 'n' to 1.

For the first and last lines, the "n - i" loop prints spaces to indent the lines. Then the 'i' is printed over and over to create the triangle edges.

For the other lines, the loop prints "" only in the first and last columns, replacing the characters in between with spaces to create the effect of whitespace. This process is repeated until 'n' rows of hollow inverted mirrored right triangle patterns are created.

Code

C

```
#include <stdio.h>
int main() {
    int n;
    printf("Enter the number of rows: ");
    scanf("%d", &n);
    for (int i = n; i >= 1; i--) {
        for (int space = 1; space <= n - i; space++) {
            printf(" ");
        }
        for (int j = 1; j <= i; j++) {
            if (j == 1 || j == i || i == n) {
                printf("* ");
            } else {
                printf(" ");
            }
        }
        printf("\n");
    }
    return 0;
}
```

</> You can also try this code with [Online C Compiler](#)



Run Code



Set your learning plan

Important for focused learning



Prepare for tech interviews



Learn competitive programming



Learn and practise coding



Already setup?

Next

1/2

Output

```
/tmp/DXSZDRP8kK.o
Enter the number of rows: 5
* * * * *
  *       *
    *     *
      *   *
        * *
          *
```

Pyramid Star Pattern

Entering 'n' for the number of columns will form a pyramid star pattern. It prints each line using two nested loops. The outer loop runs from 1 to 'n', representing each row of the pyramid.

For each line, the inner loop "n - i" prints a space in the middle of the line. Then "*" is printed "2 * i - 1" times, with a series of odd asterisks on each line. This creates a pyramid-like structure with the highest number of stars in the central column.

This process is repeated until 'n' rows of the pyramid pattern are created.

Code

C

```
#include <stdio.h>
int main() {
    int n;
    printf("Enter the number of rows: ");
    scanf("%d", &n);
    for (int i = 1; i <= n; i++) {
        for (int space = 1; space <= n - i; space++) {
            printf(" ");
        }
        for (int j = 1; j <= 2 * i - 1; j++) {
            printf("* ");
        }
        printf("\n");
    }
    return 0;
}
```

</> You can also try this code with [Online C Compiler](#)

Output

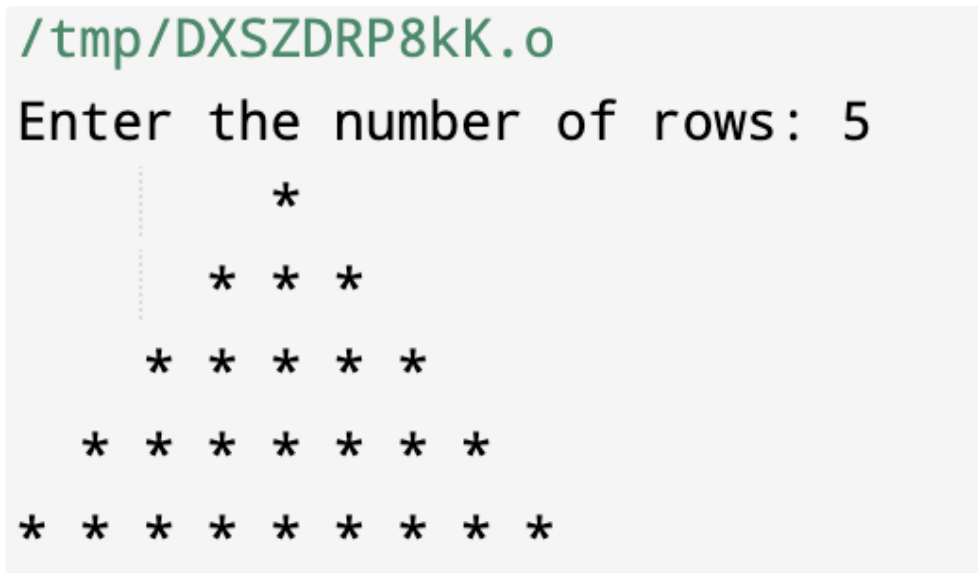
Set your learning plan

Important for focused learning

Prepare for tech
interviewsLearn competitive
programmingLearn and practise
coding**Already setup? Next**

Run

1/2



Hollow Pyramid Star Pattern

The Hollow Pyramid Star pattern requires you to enter 'n' as the number of rows. It prints each line using two nested loops.

The outer loop runs from 1 to 'n', representing each row of the pyramid. For the first and last row, the inner loop outputs "2 * i - 1" times to create the solid edge of the pyramid.

For the other rows, the inner loop prints "" only in the first and last columns, with spaces in between, creating a hollow effect. The number of spaces to print is 'n - i' before the '*' at the beginning and end of each line.

This process is repeated until 'n' columns of hollow pyramid patterns are created.

Code

C

```

#include <stdio.h>
int main() {
    int n;
    printf("Enter the number of rows: ");
    scanf("%d", &n);
    for (int i = 1; i <= n; i++) {
        for (int space = 1; space <= n - i; space++) {
            printf(" ");
        }
        for (int j = 1; j <= 2 * i - 1; j++) {
            if (j == 1 || j == 2 * i - 1 || i == n) {
                printf("* ");
            } else {
                printf(" ");
            }
        }
        printf("\n");
    }
    return 0;
}

```

Set your learning plan

Important for focused learning



Prepare for tech interviews



Learn competitive programming



Learn and practise coding



Already setup? **Next**

1/2

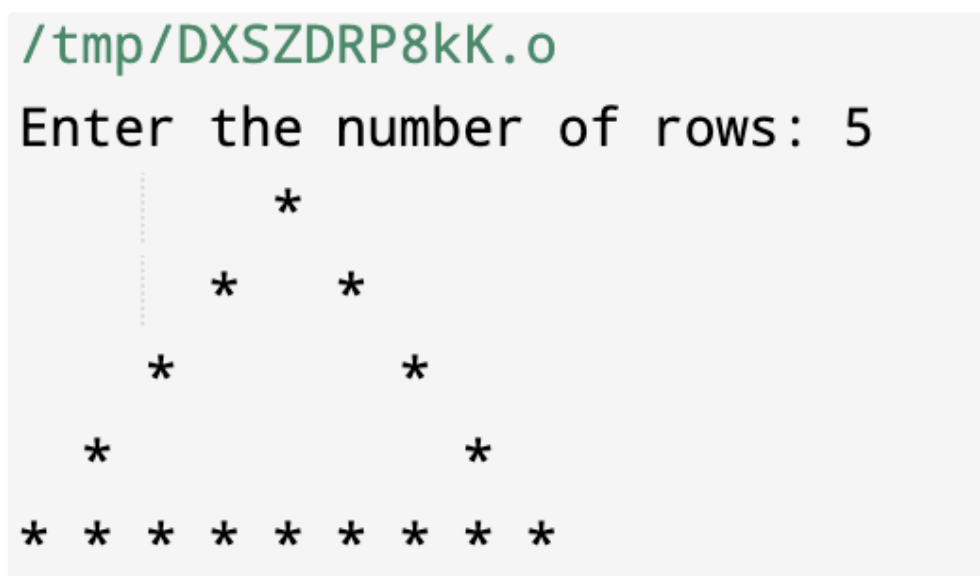
You can also try this code with [Online C Compiler](#)



[Run Code](#)



Output



Inverted Pyramid Star Pattern

An inverted pyramid star pattern is formed by entering 'n' for the number of rows. Plot each row of the inverted pyramid using a single loop that iterates from 'n' to 1.

For each line, the "n - i" loop prints a space to properly indent the line. Then "*" is printed "2 * i - 1" times, with a series of odd asterisks on each line.

This creates a structure like an inverted pyramid with the highest number of stars in the top row and one star in the last row. This process is repeated until an inverted pyramid pattern with 'n' rows is created.

Code

C

```
#include <stdio.h>
int main() {
    int n;
    printf("Enter the number of rows: ");
    scanf("%d", &n);
    for (int i = n; i >= 1; i--) {
        for (int space = 1; space <= n - i; space++) {
            printf(" ");
        }
        for (int j = 1; j <= 2 * i - 1; j++) {
            printf("* ");
        }
        printf("\n");
    }
    return 0;
}
```

</> You can also try this code with [Online C Compiler](#)

Set your learning plan

Important for focused learning



Prepare for tech
interviews



Learn competitive
programming



Learn and practise
coding



Already setup?

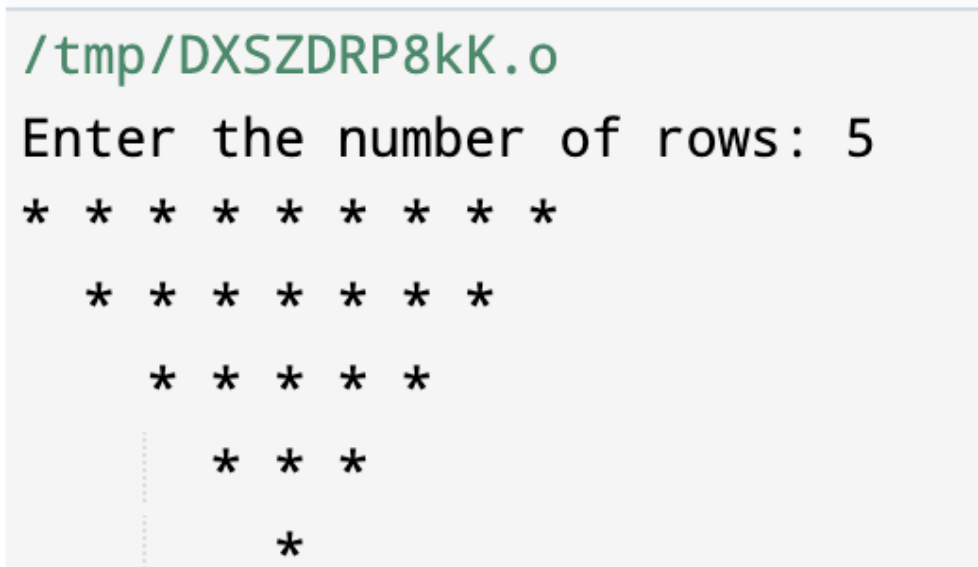
Next

Run

1/2

Output





Half Diamond Star Pattern

Entering 'n' for the number of rows will form a half-diamond star pattern. Print each line using two nested loops.

The outer loop goes from 1 to 'n', representing each row of the half-diamond. For each row, the inner loop '*' outputs 'i' times, creating a sequence of increasing asterisks for each row.

This creates a semi-rhombic structure with the highest number of stars in the last row. This process is repeated until 'n' rows of half-diamond patterns are created.

Code

C

```
#include <stdio.h>
int main() {
    int n;
    printf("Enter the number of rows: ");
    scanf("%d", &n);

    for (int i = 1; i <= n; i++) {
        for (int j = 1; j <= i; j++) {
            printf("* ");
        }
        printf("\n");
    }
    for (int i = n - 1; i >= 1; i--) {
        for (int j = 1; j <= i; j++) {
            printf("* ");
        }
        printf("\n");
    }
    return 0;
}
```

</> You can also try this code with [Online C Compiler](#)

Set your learning plan

Important for focused learning



Prepare for tech interviews



Learn competitive programming



Learn and practise coding



Already setup?

Next

Run 1/2

Output



```
/tmp/DXSZDRP8kK.o
```

```
Enter the number of rows: 5
```

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

Diamond Star Pattern

A diamond star pattern is formed by entering 'n' for the number of rows. Print each line using two nested loops. The outer loop goes from 1 to 'n', representing each row in the upper half of the diamond.

For each line, the inner loop "n - i" prints a space in the middle of the line. Then "*" is printed "2 * i - 1" times, with a series of odd asterisks on each line. This creates a diamond-like structure with the maximum number of stars placed in the center row of the diamond.

The bottom half of the diamond is printed with a similar approach, but the outer loop runs from 'n - 1' to 1. This process is repeated until 'n' rows of diamond patterns are created.

Code

C

```
#include <stdio.h>
int main() {
    int n;
    printf("Enter the number of rows: ");
    scanf("%d", &n);
    // Upper half of the diamond
    for (int i = 1; i <= n; i++) {
        for (int space = 1; space <= n - i; space++) {
            printf(" ");
        }
        for (int j = 1; j <= 2 * i - 1; j++) {
            printf("*");
        }
        printf("\n");
    }
```

Set your learning plan

Important for focused learning



Prepare for tech interviews



Learn competitive programming



Learn and practise coding



Already setup?

Next

1/2



```
}  
// Lower half of the diamond  
for (int i = n - 1; i >= 1; i--) {  
    for (int space = 1; space <= n - i; space++) {  
        printf(" ");  
    }  
    for (int j = 1; j <= 2 * i - 1; j++) {  
        printf("*");  
    }  
    printf("\n");  
}  
return 0;  
}
```

</> You can also try this code with [Online C Compiler](#)

[Run Code](#)

Output

```
Enter the number of rows: 5  
  
    *  
  ***  
 *****  
*****  
*****  
  *****  
    *****  
      ***  
        *
```

Right Arrow Star Pattern

A right arrow star pattern is formed by entering 'n' for the number of rows. Print each row using two nested loops.

The outer loop runs from 1 to 'n', representing each row of right arrows. For each line, the inner loop 'n - i' prints a space to indent the line properly. Then the 'i' is printed over and over, creating a sequence of increasing asterisks on each line. This creates a structure that looks like an arrow pointing to the right, with the highest number of stars in the last row.

After printing 'n' rows of right arrows, the pattern is completed by printing 'n-1' rows in reverse order to create the ends of the arrow. This process is repeated until "2n-1" rows of right arrow patterns are generated.

Set your learning plan

Important for focused learning



Prepare for tech interviews



Learn competitive programming



Learn and practise coding



Already setup?

[Next](#)

1/2



Code

C

```
#include <stdio.h>
int main() {
    int n;
    printf("Enter the number of rows: ");
    scanf("%d", &n);

    // Print the upper part of the arrow
    for (int i = 1; i < n; i++) {
        for (int j = 1; j <= (2 * i - 2); j++) {
            printf(" ");
        }

        // Print inverted right triangle star pattern
        for (int j = i; j <= n; j++) {
            printf("*");
        }
        printf("\n");
    }

    // Print lower part of the arrow
    for (int i = 1; i <= n; i++) {
        for (int j = 1; j <= (2 * n - 2 * i); j++) {
            printf(" ");
        }

        // Print simple right triangle star pattern
        for (int j = 1; j <= i; j++) {
            printf("*");
        }
        printf("\n");
    }
    return 0;
}
```

[You can also try this code with Online C Compiler](#)[Run Code](#)

Output

Set your learning plan

Important for focused learning



Prepare for tech interviews



Learn competitive programming



Learn and practise coding

[Already setup?](#) **Next**

1/2



```

/tmp/tIWJN4odCy.o
Enter the number of rows: 5
*****
 
 ****
 
  ***
 
   **
 
    *
 
   **
 
  ***
 
 ****
 
*****

```

Left Arrow Star Pattern

The left arrow star pattern is similar to the right arrow pattern, but printed in reverse order. Also, you must enter 'n' as the number of rows. Print each line using two nested loops.

The outer loop runs from 'n' to 1, representing each row of left arrows. For each line, the inner loop 'n - i' prints a space to indent the line properly.

Then the 'i' is printed over and over, creating a sequence of increasing asterisks on each line. This creates a left arrow-like structure with the maximum number of stars in the first row.

After printing 'n' rows of left arrows, the pattern is completed by printing 'n-1' rows reverse order to create the end of the arrow.

This process is repeated until the "2n-1" row left arrow pattern is generated.

Code

C


```


#include <stdio.h>
int main() {
    int n;
    printf("Enter the number of rows: ");
    scanf("%d", &n);
    // Print upper part of the arrow
    for (int i = 1; i < n; i++) {
        // Print trailing (n-rownumber) spaces
        for (int j = 1; j <= (n - i); j++) {
            printf(" ");


```

Set your learning plan

Important for focused learning

 Prepare for tech interviews ☐

 Learn competitive programming ☐

 Learn and practise coding ☐

Already setup? **Next**

1/2


```
}  
// Print inverted right triangle  
for (int j = i; j <= n; j++) {  
    printf("*");  
}  
printf("\n");  
}  
// Print bottom part of the arrow  
for (int i = 1; i <= n; i++) {  
    for (int j = 1; j < i; j++) {  
        printf(" ");  
    }  
    // Print the right triangle  
    for (int j = 1; j <= i; j++) {  
        printf("*");  
    }  
    printf("\n");  
}  
return 0;  
}
```

</> You can also try this code with [Online C Compiler](#)

[Run Code](#)

Output

```
/tmp/t1WJN4odCy.o  
Enter the number of rows: 5  
  
*****  
  ****  
   ***  
    **  
     *  
    **  
   ***  
  ****  
 *****  
*****
```

Plus Star Pattern

Set your learning plan

Important for focused learning



Prepare for tech
interviews



Learn competitive
programming



Learn and practise
coding



[Already setup?](#)

[Next](#)

1/2



Entering 'n' for the number of rows will form a plus star pattern. Print each line using two nested loops. The outer loop runs from 1 to 'n', representing each row of the plus pattern.

For each row, the inner loop outputs 'n' times to create a horizontal line of stars. After printing 'n' rows of horizontal lines, the pattern is completed by printing 'n-1' rows forming a vertical line of plus signs around the ' '.

This process is repeated until a "2*n-1" row plus-star pattern is generated.

Code

C

```
#include <stdio.h>
int main() {
    int n;
    printf("Enter the size of plus pattern (odd number): ");
    scanf("%d", &n);
    // Upper part of the plus pattern
    for (int i = 1; i <= n; i++) {
        if (i == (n + 1) / 2) {
            for (int j = 1; j <= n; j++) {
                printf("* ");
            }
        } else {
            for (int j = 1; j <= n; j++) {
                if (j == (n + 1) / 2) {
                    printf("* ");
                } else {
                    printf(" ");
                }
            }
        }
        printf("\n");
    }
    return 0;
}
```

</> You can also try this code with [Online C Compiler](#)




[Run Code](#)

Output

```
/tmp/tIWJN4odCy.o
Enter the size of plus pattern (odd number): 7



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

Set your learning plan
Important for focused learning

-  Prepare for tech interviews ☐
-  Learn competitive programming ☐
-  Learn and practise coding ☐

[Already setup?](#) [Next](#)

1/2



X Star Pattern

An X star pattern is formed by entering 'n' for the number of rows. Print each line using two nested loops. The outer loop runs from 1 to 'n', representing each row of the X pattern.

For each row, the inner loop outputs a '*' or a space (" ") based on the position calculated by the expression 'i == j || j == n-i+1'. This creates an X-like structure with the stars in the appropriate positions.

This process is repeated until 'n' rows of X-star patterns are generated.

Code

C

```
#include <stdio.h>
int main() {
    int n;
    printf("Enter the size of X pattern (odd number and minimum 3): ");
    scanf("%d", &n);
    // X pattern
    for (int i = 1; i <= n; i++) {
        for (int j = 1; j <= n; j++) {
            if (j == i || j == n - i + 1) {
                printf("* ");
            } else {
                printf(" ");
            }
        }
        printf("\n");
    }
    return 0;
}
```

</> You can also try this code with [Online C Compiler](#)

Run Code

Output

```
/tmp/t1WJN4odCy.o
Enter the size of X pattern (odd number and minimum 3): 7
*           *
 *         *
  *       *
   *     *
    *   *
   * * 
  *   *
 *     *
*       *
```

Set your learning plan

Important for focused learning



Prepare for tech interviews



Learn competitive programming



Learn and practise coding



Already setup?

Next

1/2

Frequently Asked Questions

What is star pattern in C?

A star pattern in C involves printing a specific arrangement of asterisks (*) to form various

visual designs. It's a common exercise to practice nested loops and control structures.

How to write a star pattern in C?

To create a star pattern in C, you use nested loops. The outer loop controls the number of rows, while the inner loop manages the number of asterisks in each row. By adjusting loop logic, you can create different patterns.

How to print star sequence in C?

Printing a star sequence in C typically involves using loops to generate a specific pattern, such as triangles, rectangles, or other geometrical shapes, using asterisks.

How to print 1 2 3 4 5 in C?

To print numbers 1 to 5 in C, you can use a simple loop, like a for loop, and increment the variable to display the desired sequence. For example, `for(int i = 1; i <= 5; i++) printf("%d ", i);` will output 1 2 3 4 5.

Conclusion

Star patterns program in C are an essential practice for understanding loop structures. They provide a hands-on approach to mastering nested loops and their applications, making them a valuable learning tool for budding C programmers.

Refer to the below topics for more understanding

- [Getch in C](#)
- [Problems in C](#)
- [Introduction to C language](#)
- [Ternary Operator in C](#)

Happy Learning!

Interested in Coding Ninjas Flagship Courses?


◀ Previous article

Binary Search in C
Using Recursion

Click Here

Next article

Implementation of Tic
Tac Toe Game



CODING NINJAS

Related articles

Download our app:

GET IT ON Google Play

About Us

Press

Privacy

Terms & Conditions

Floyd's Triangle in C

Floyd's triangle in C

Bounty

Hire from

Coding Ninjas

Studio

Problem of the day

Interview Problems

Interview

Experiences

Interview

Bundle

Guided

Paths

Library

Test Series

Contest

Coding Ninjas

Studio

Blog


Events


Campus


Ninjas

Set your learning plan

Important for focused learning

 Prepare for tech interviews ☐

 Learn competitive programming ☐

 Learn and practise coding ☐

Already setup?

Next

1/2

B *I*

Write your thoughts...

Cancel

Comment

Important Links

- Job Bootcamp:** [Web Development Bootcamp \(MERN & Springboot\)](#)
- Coding courses beginners:** [C++ Foundation with Data Structures](#) | [JAVA Foundation with Data Structures](#) | [Interview Preparation for Product Companies](#) | [Competitive Programming Course](#) | [Python Foundation with Data Structures](#) | [Aptitude Preparation Course](#)
- Advanced coding course:** [Advanced Front-End Web Development with React](#) | [Machine Learning and Deep Learning Course](#) | [Full Stack Web Development Course](#) | [Data Science & Machine Learning Complete](#) | [Operating Systems](#) | [System Design](#) | [Database Management System](#) | [MERN Stack Course](#) | [Data Analytics Course](#)
- Free trial:** [C++ Foundation with Data Structures](#) | [Java Foundation with Data Structures](#) | [Competitive Course](#) | [Aptitude Preparation Course](#) | [Python Foundation with Data Structures](#) | [Advanced Front-End Web Development with React](#) | [Data Science & Machine Learning Complete](#) | [Interview Preparation for Product Companies](#) | [Machine Learning Course](#)
- Free contents:** [C++ Foundation with Data Structures](#) | [Java Foundation with Data Structures](#) | [Competitive Programming Course](#) | [Aptitude Preparation Course](#) | [Python Foundation with Data Structures](#) | [Advanced Front-End Web Development with React](#) | [Data Science & Machine Learning Complete](#) | [Interview Preparation for Product Companies](#)

No comments yet

Be the first to share what you think

We accept payments using:



No Cost EMI



100% safe & secure payment

Set your learning plan

Important for focused learning



Prepare for tech interviews



Learn competitive programming



Learn and practise coding



Already setup?

Next

1/2

