**For loop**

program in C to display a pattern like a right angle triangle #include <stdio.h>  // Include the standard input/output header file.void main() {    int i, j, rows;  // Declare variables 'i' and 'j' for loop counters, 'rows' for user input.    printf("Input number of rows : ");  // Print a message to prompt user input.    scanf("%d", &rows);  // Read the value of 'rows' from the user.    for (i = 1; i <= rows; i++) {  // Start a loop to generate rows of asterisks.        for (j = 1; j <= i; j++)  // Nested loop to print asterisks based on the current row.            printf("\*");  // Print an asterisk.        printf("\n");  // Move to the next line for the next row.    }}display a pattern like a right angle triangle with a number.#include <stdio.h>  // Include the standard input/output header file.void main() {    int i, j, rows;  // Declare variables 'i' and 'j' for loop counters, 'rows' for user input.    printf("Input number of rows : ");  // Print a message to prompt user input.    scanf("%d", &rows);  // Read the value of 'rows' from the user.    for (i = 1; i <= rows; i++) {  // Start a loop to generate rows.        for (j = 1; j <= i; j++)  // Nested loop to print numbers based on the current row.            printf("%d", j);  // Print the value of 'j'.        printf("\n");  // Move to the next line for the next row.    }}make such a pattern like a right angle triangle with the number increased by 1.#include <stdio.h>  // Include the standard input/output header file.void main() {    int i, j, rows, k = 1;  // Declare variables 'i' and 'j' for loop counters, 'rows' for user input, 'k' for incrementing numbers.    printf("Input number of rows : ");  // Print a message to prompt user input.    scanf("%d", &rows);  // Read the value of 'rows' from the user.    for (i = 1; i <= rows; i++) {  // Start a loop to generate rows.        for (j = 1; j <= i; j++)  // Nested loop to print numbers based on the current row.            printf("%d ", k++);  // Print the value of 'k' and increment it.        printf("\n");  // Move to the next line for the next row.    }}pattern as a pyramid with an asterisk.#include <stdio.h>  // Include the standard input/output header file.void main() {   int i, j, spc, rows, k;  // Declare variables 'i' and 'j' for loop counters, 'spc' for spaces, 'rows' for user input, 'k' for loop counter.   printf("Input number of rows : ");  // Print a message to prompt user input.   scanf("%d", &rows);  // Read the value of 'rows' from the user.   spc = rows + 4 - 1;  // Calculate the initial number of spaces.   for (i = 1; i <= rows; i++) {  // Start a loop to generate rows.      for (k = spc; k >= 1; k--) {  // Loop to print spaces before the asterisks.         printf(" ");      }      for (j = 1; j <= i; j++) {  // Loop to print asterisks based on the current row.         printf("\* ");  // Print an asterisk followed by a space.      }      printf("\n");  // Move to the next line for the next row.      spc--;  // Decrement the number of spaces for the next row.   }}Pascal triangle #include <stdio.h> // Include the standard input/output header file.void main(){    int no\_row,c=1,blk,i,j; // Declare variables for row count, pattern counter, and loop control.    printf("Input number of rows: "); // Prompt the user to input the number of rows.    scanf("%d",&no\_row); // Read the input from the user.    for(i=0;i<no\_row;i++) // Outer loop for iterating over rows.    {        for(blk=1;blk<=no\_row-i;blk++) // Inner loop for printing spaces.            printf("  ");        for(j=0;j<=i;j++) // Inner loop for generating and printing pattern.        {            if (j==0||i==0) // If it's the first column or first row, set c to 1.                c=1;            else               c=c\*(i-j+1)/j; // Calculate the next pattern value.            printf("% 4d",c); // Print the pattern value.        }        printf("\n"); // Move to the next row.    }}Print prymis using while #include <stdio.h>int main() {    int rows, i = 1, j, k;    // Number of rows for the pyramid    printf("Enter the number of rows: ");    scanf("%d", &rows);    while(i <= rows) {        // Print spaces for alignment        j = rows;        while(j > i) {            printf(" ");            j--;        }                // Print asterisks for the pyramid        k = 1;        while(k <= (2\*i - 1)) {            printf("\*");            k++;        }                // Move to the next line after each row        printf("\n");        i++;    }    return 0;}print Floyd's Triangle.The Floyd's triangle is#include <stdio.h> // Include the standard input/output header file.void main(){    int i, j, n, p, q; // Declare variables to store input and control loop indices.    printf("Input number of rows : "); // Prompt the user for input.    scanf("%d", &n); // Read the value of 'n' from the user.    for (i = 1; i <= n; i++) // Loop for the number of rows.    {        if (i % 2 == 0) // Check if 'i' is even.        {            p = 1;            q = 0;        }        else // If 'i' is odd.        {            p = 0;            q = 1;        }        for (j = 1; j <= i; j++) // Loop for each element in the row.        {            if (j % 2 == 0) // Check if 'j' is even.                printf("%d", p); // Print 'p' if 'j' is even.            else                printf("%d", q); // Print 'q' if 'j' is odd.        }        printf("\n"); // Move to the next line after printing a row.    }}Diamond pattern#include <stdio.h> // Include the standard input/output header file.void main(){   int i, j, r; // Declare variables for loop control and the number of rows.   printf("Input number of rows (half of the diamond) :"); // Prompt the user to input the number of rows.   scanf("%d", &r); // Read the number of rows from the user.   for(i = 0; i <= r; i++) // Start a loop to print the upper half of the diamond.   {     for(j = 1; j <= r - i; j++) // Loop to print spaces before the asterisks.       printf(" ");     for(j = 1; j <= 2 \* i - 1; j++) // Loop to print asterisks.       printf("\*");     printf("\n"); // Move to the next line after completing a row.   }   for(i = r - 1; i >= 1; i--) // Start a loop to print the lower half of the diamond.   {     for(j = 1; j <= r - i; j++) // Loop to print spaces before the asterisks.       printf(" ");     for(j = 1; j <= 2 \* i - 1; j++) // Loop to print asterisks.       printf("\*");     printf("\n"); // Move to the next line after completing a row.   }}