1. **Input**

```
Input number of rows: 5
```

Output

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

```
#include <stdio.h>

int main()
{
    int i, j, N;

    /* Input number of rows from user */
    printf("Enter number of rows: ");
    scanf("%d", &N);

    /* Iterate through N rows */
    for(i=1; i<=N; i++)
    {
        /* Iterate over columns */
        for(j=1; j<=N; j++)
        {
            /* Print star for each column */
            printf("*");
        }

        /* Move to the next line/row */
        printf("\n");
    }

    return 0;
}
```

2.

Input

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

Output

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

```c
#include <stdio.h>

int main()
{
    int i, j, N;

    /* Input number of rows from user */
    printf("Enter number of rows: ");
    scanf("%d", &N);

    /* Iterate over each row */
    for(i=1; i<=N; i++)
    {
        /* Iterate over each column */
        for(j=1; j<=N; j++)
        {
            if(i==1 || i==N || j==1 || j==N)
            {
                /* Print star for 1st, Nth row and column */
                printf("*");
            }
            else
            {
                printf(" ");
            }
        }

        /* Move to the next line/row */
        printf("\n");
    }

    return 0;
}
```

3.

Input

```
Input rows: 5
```

Output

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

```
#include <stdio.h>

int main()
{
    int i, j, rows;

    /* Input rows from user */
    printf("Enter number of rows: ");
    scanf("%d", &rows);

    for(i=1; i<=rows; i++)
    {
        for(j=1; j<=i; j++)
        {
            /*
             * Print star for first column(j==1),
             * last column(j==i) or last row(i==rows).
             */
            if(j==1 || j==i || i==rows)
            {
                printf("*");
            }
            else
            {
                printf(" ");
            }
        }

        printf("\n");
    }

    return 0;
}
```

4. Input

```
Input rows: 5
```

Output

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

```c
#include <stdio.h>

int main()
{
    int i, j, rows;
    int stars, spaces;

    printf("Enter rows to print : ");
    scanf("%d", &rows);


    stars = 1;
    spaces = rows - 1;

    /* Iterate through rows */
    for(i=1; i<rows*2; i++)
    {
        /* Print spaces */
        for(j=1; j<=spaces; j++)
            printf(" ");

        /* Print stars */
        for(j=1; j<stars*2; j++)
            printf("*");

        /* Move to next line */
        printf("\n");

        if(i<rows)
        {
            spaces--;
            stars++;
        }
        else
        {
            spaces++;
            stars--;
        }
    }
    return 0;
}
```

5.

Input

```
Input rows: 5
Input columns: 5
```

Output

```
1  2  3  4  5
6  7  8  9 10
11 12 13 14 15
16 17 18 19 20
21 22 23 24 25
```

```
#include <stdio.h>

int main()
{
    int rows, cols, i, j, k;

    /* Input rows and columns from user */
    printf("Enter number of rows: ");
    scanf("%d", &rows);
    printf("Enter number of columns: ");
    scanf("%d", &cols);

    k = 1;
    for(i=1; i<=rows; i++)
    {
        for(j=1; j<=cols; j++, k++)
        {
            printf("%-3d", k);
        }

        printf("\n");
    }

    return 0;
}
```

6.

Input

```
Input N: 5
```

Output

Pattern 1:   Pattern 2:

```
 1                    1
 12                  12
 123                123
 1234              1234
 12345            12345
```

Pattern1 :

```
#include <stdio.h>

int main()
{
    int i, j, N;

    printf("Enter N: ");
    scanf("%d", &N);

    for(i=1; i<=N; i++)
    {
        // Logic to print numbers
        for(j=1; j<=i; j++)
        {
            printf("%d", j);
        }

        printf("\n");
    }

    return 0;
}
```

Pattern 2:

```
#include <stdio.h>

int main()
{
    int i, j, N;

    printf("Enter N: ");
    scanf("%d", &N);

    for(i=1; i<=N; i++)
    {
        // Logic to print spaces
        for(j=1; j<=N-i; j++)
        {
            printf(" ");
        }

        // Logic to print numbers
        for(j=1; j<=i; j++)
        {
            printf("%d", j);
        }

        printf("\n");
    }

    return 0;
}
```

Input

```
Input N: 5
```

Output

```
 12345      12345
 1234        1234
 123          123
 12            12
 1              1
```

Series:

1+2+3+4+5+………+n
1+4+7+10+……….+n
1+5+9+13+….

```
#include <stdio.h>

int main()
{
        int terms; //to store total terms
        int i;
        int sum=0, temp=1;

        printf("Enter total number of terms: ");
        scanf("%d",&terms);


        for(i=0; i<terms; i++)
        {
                printf("%d ", temp);
                if(i<terms-1)
                        printf("+ ");
                sum += temp;
                temp +=4;
        }

        printf("\nSUM of the series is: %d\n",sum);

        return 0;
}
```

$1^2+2^2+3^2+\ldots\ldots\ldots\ldots+n^2$

$1+1/2^2+1/3^2+\ldots\ldots\ldots+1/n^2$

1-2+3-4+5-6+……..n

```
#include <stdio.h>

int main()
{
    int n, sum=0;
    printf("Series:1-2+3-4+5-6+7-8.....N\n");
    printf("Want some up to N terms?\nEnter the N term:");
    scanf("%d", &n);
        if (n % 2 == 0)
        sum =(-(n / 2));
    else
        sum = ((n + 1) / 2);

    printf("Sum is:%d", sum);

    return 0;
}
```

1+11+111+1111+...

```
#include <stdio.h>

int main()
{
        int terms; //to store total terms
        int i; //to run loo;
        long int sum; //to store sum of values
        long int temp; //to add the diff or initial term value

        //assign 0 to sum and assign 1 as initial term value
        sum =0;
        temp =1;
        //input total terms
        printf("Enter total number of terms: ");
        scanf("%d",&terms);

        //run loop to find sum of each value
        //and then increase it with the differences
        for(i=0; i<terms; i++)
        {
                //print the  value
                printf("%ld ", temp);
                //print '+' sign in the series
                if(i<terms-1)
                        printf("+ ");

                //add the value and store in sum
                sum += temp;
                //update/increase the value
                temp = (temp*10)+1;
        }

        //print the value of sum of the series
        printf("\nSUM of the series is: %ld\n",sum);

        return 0;
}
```

Array:

delete element from an array

```
#include <stdio.h>
#define MAX_SIZE 100

int main()
{
    int arr[MAX_SIZE];
    int i, size, pos;

    /* Input size and element in array */
    printf("Enter size of the array : ");
    scanf("%d", &size);
    printf("Enter elements in array : ");
    for(i=0; i<size; i++)
    {
        scanf("%d", &arr[i]);
    }

    /* Input element position to delete */
    printf("Enter the element position to delete : ");
    scanf("%d", &pos);
    /* Invalid delete position */
    if(pos < 0 || pos > size)
    {
        printf("Invalid position! Please enter position between 1 to %d",
size);
    }
    else
    {
        /* Copy next element value to current element */
        for(i=pos-1; i<size-1; i++)
        {
            arr[i] = arr[i + 1];
        }

        /* Decrement array size by 1 */
        size--;

        /* Print array after deletion */
        printf("\nElements of array after delete are : ");
        for(i=0; i<size; i++)
        {
            printf("%d\t", arr[i]);
        }
    }

    return 0;
}
```

Output

```
Enter size of the array : 5
Enter elements in array : 10 20 30 40 50
Enter the element position to delete : 2

Elements of array after delete are : 10      30      40      50
```

count frequency of each element of array

```
#include <stdio.h>

int main()
{
    int arr[100], freq[100];
    int size, i, j, count;

    /* Input size of array */
    printf("Enter size of array: ");
    scanf("%d", &size);

    /* Input elements in array */
    printf("Enter elements in array: ");
```

```c
    for(i=0; i<size; i++)
    {
        scanf("%d", &arr[i]);

        /* Initially initialize frequencies to -1 */
        freq[i] = -1;
    }


    for(i=0; i<size; i++)
    {
        count = 1;
        for(j=i+1; j<size; j++)
        {
            /* If duplicate element is found */
            if(arr[i]==arr[j])
            {
                count++;

                /* Make sure not to count frequency of same element again */
                freq[j] = 0;
            }
        }

        /* If frequency of current element is not counted */
        if(freq[i] != 0)
        {
            freq[i] = count;
        }
    }

    /*
     * Print frequency of each element
     */
    printf("\nFrequency of all elements of array : \n");
    for(i=0; i<size; i++)
    {
        if(freq[i] != 0)
        {
            printf("%d occurs %d times\n", arr[i], freq[i]);
        }
    }

    return 0;
}
```

Output

```
Enter size of array: 10
Enter elements in array: 5 10 2 5 50 5 10 1 2 2

Frequency of all elements of array :
5 occurs 3 times
10 occurs 2 times
2 occurs 3 times
50 occurs 1 times
1 occurs 1 times
```

Program to print unique elements in array

Program to find sum of rows and columns of matrix

```c
/**
 * C program to find sum of elements of rows and columns of matrix
 */

#include <stdio.h>

#define SIZE 3 // Matrix size

int main()
{
    int A[SIZE][SIZE];
    int row, col, sum = 0;

    /* Input elements in matrix from user */
    printf("Enter elements in matrix of size %dx%d: \n", SIZE, SIZE);
    for(row=0; row<SIZE; row++)
    {
        for(col=0; col<SIZE; col++)
        {
            scanf("%d", &A[row][col]);
        }
    }

    /* Calculate sum of elements of each row of matrix */
    for(row=0; row<SIZE; row++)
    {
        sum = 0;
        for(col=0; col<SIZE; col++)
        {
            sum += A[row][col];
        }

        printf("Sum of elements of Row %d = %d\n", row+1, sum);
    }

    /* Find sum of elements of each columns of matrix */
    for(row=0; row<SIZE; row++)
    {
        sum = 0;
        for(col=0; col<SIZE; col++)
        {
            sum += A[col][row];
        }

        printf("Sum of elements of Column %d = %d\n", row+1, sum);
    }

    return 0;
}
```

C program to find upper triangular matrix

```
#include <stdio.h>
#define MAX_ROWS 3
#define MAX_COLS 3

int main()
{
    int array[MAX_ROWS][MAX_COLS];
    int row, col, isUpper;

    /* Input elements in matrix from user */
    printf("Enter elements in matrix of size %dx%d: \n", MAX_ROWS, MAX_COLS);
    for(row=0; row<MAX_ROWS; row++)
    {
        for(col=0; col<MAX_COLS; col++)
        {
            scanf("%d", &array[row][col]);
        }
    }

    /* Check Upper triangular matrix condition */
    isUpper = 1;
    for(row=0; row<MAX_ROWS; row++)
    {
        for(col=0; col<MAX_COLS; col++)
        {
            /*
             * If elements below the main diagonal (col<row)
             * is not equal to zero then it is not upper triangular matrix
             */
            if(col<row && array[row][col]!=0)
            {
                isUpper = 0;
            }
        }
    }

    /* Print elements of upper triangular matrix  */
    if(isUpper == 1)
    {
        printf("\nThe matrix is Upper triangular matrix.\n");

        for(row=0; row<MAX_ROWS; row++)
        {
            for(col=0; col<MAX_COLS; col++)
            {
                printf("%d ", array[row][col]);
            }

            printf("\n");
        }
    }
    else
    {
        printf("\nThe matrix is not Upper triangular matrix.");
    }

    return 0;
}
```

Output

```
Enter elements in matrix of size 3x3:
1 2 3
0 5 6
0 0 9

The matrix is Upper triangular matrix.
1 2 3
0 5 6
0 0 9
```