

Q1 Write a C++ program to print factorial of a number.

```
#include<iostream>

using namespace std;

int main()
{
    int num,factorial=1;

    cout<<"Enter Number To Find Its Factorial ";

    cin>>num;

    for (int a=1;a<=num;a++) {

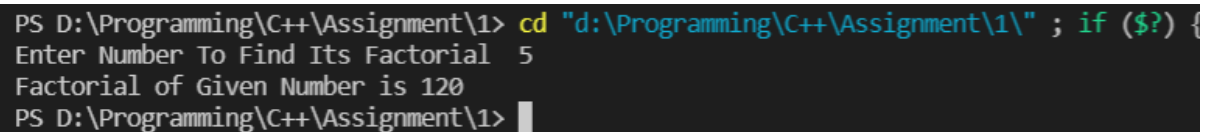
        factorial=factorial*a;

    }

    cout<<"Factorial of Given Number is "<<factorial<<endl;

    return 0;

}
```



```
PS D:\Programming\C++\Assignment\1> cd "d:\Programming\C++\Assignment\1\" ; if ($?) {
Enter Number To Find Its Factorial 5
Factorial of Given Number is 120
PS D:\Programming\C++\Assignment\1> █
```

Q2 Write a C++ program to print factorial of a number through recursion.

```
#include<iostream>

using namespace std;

int factorial(int n);

int main()
{
    int n;

    cout << "Enter a positive integer: ";
```

```

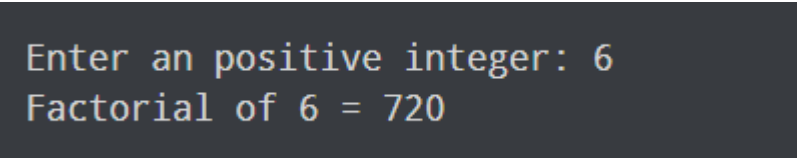
    cin >> n;

    cout << "Factorial of " << n << " = " << factorial(n);

    return 0;
}

int factorial(int n)
{
    if(n > 1)
        return n * factorial(n - 1);
    else
        return 1;
}

```



```

Enter an positive integer: 6
Factorial of 6 = 720

```

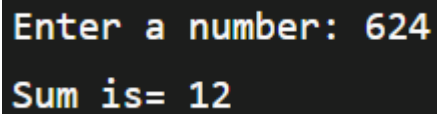
Q3 Write a C++ program to print sum of digits.

```

#include <iostream>
using namespace std;
int main()
{
    int n,sum=0,m;
    cout<<"Enter a number: ";
    cin>>n;
    while(n>0)
    {
        m=n%10;
        sum=sum+m;
    }
}

```

```
n=n/10;
}
cout<<"Sum is= "<<sum<<endl;
return 0;
}
```

A terminal window with a black background and yellow text. The first line reads "Enter a number: 624" and the second line reads "Sum is= 12".

```
Enter a number: 624
Sum is= 12
```

Q4 Write a C++ program to reverse given number.

```
#include <iostream>
using namespace std;

int main() {
    int n, reversedNumber = 0, remainder;

    cout << "Enter an integer: ";
    cin >> n;

    while(n != 0) {
        remainder = n%10;
        reversedNumber = reversedNumber*10 + remainder;
        n /= 10;
    }

    cout << "Reversed Number = " << reversedNumber;

    return 0;
}
```

```
Enter an integer: 12345
Reversed number = 54321
```

Q5 Write a C++ program to print multiplication of 2 matrices.

```
#include<iostream>

using namespace std;

int main()
{
    int i, j, N=4;

    int res[N][N]; // To store result

    int mat1[N][N] = { { 1, 1, 1, 1 },
                        { 2, 2, 2, 2 },
                        { 3, 3, 3, 3 },
                        { 4, 4, 4, 4 } };

    int mat2[N][N] = { { 1, 1, 1, 1 },
                        { 2, 2, 2, 2 },
                        { 3, 3, 3, 3 },
                        { 4, 4, 4, 4 } };

    int k;

    for (i = 0; i < N; i++) {
        for (j = 0; j < N; j++) {
            res[i][j] = 0;

            for (k = 0; k < N; k++)
                res[i][j] += mat1[i][k] * mat2[k][j];
        }
    }

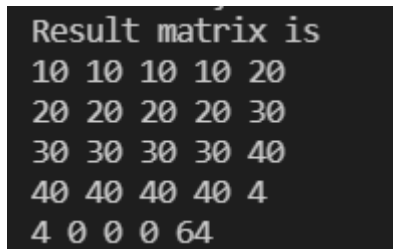
    cout << "Result matrix is \n";

    for(i=0; i<=N; i++){
        for(j=0; j<=N; j++)
            cout<<res[i][j]<<" ";
    }
}
```

```

        cout<<endl;
    }
    return 0;
}

```



```

Result matrix is
10 10 10 10 20
20 20 20 20 30
30 30 30 30 40
40 40 40 40 4
4 0 0 0 64

```

Q6 Write a C++ program to convert decimal number to binary.

```

#include <iostream>
#include <cmath>

```

```

using namespace std;

```

```

// function prototype

```

```

int convert(long long);

```

```

int main() {

```

```

    long long n;

```

```

    cout << "Enter a binary number: ";

```

```

    cin >> n;

```

```

    cout << n << " in binary = " << convert(n) << " in decimal";

```

```

    return 0;

```

```

}

```

```

// function definition

```

```

int convert(long long n) {

```

```

    int dec = 0, i = 0, rem;

```

```

while (n!=0) {
    rem = n % 10;
    n /= 10;
    dec += rem * pow(2, i);
    ++i;
}

```

```

Enter a binary number: 1101
1101 in binary = 13 in decimal

```

Q7 Write a C++ program to convert number in characters.

```

// C++ program to convert number in characters
#include<bits/stdc++.h>
using namespace std;
void NumbertoCharacter(int n)
{
    int rev = 0, r = 0;

    // To calculate the reverse of the number
    while (n > 0) {

        // The remainder will give the last digit of the number
        r = n % 10;
        rev = rev * 10 + r;
        n = n / 10;
    }

    while (rev > 0) {
        // Extract the first digit of the reversed number
        r = rev % 10;

```

```
// Match it with switch case
```

```
switch (r) {
```

```
case 1:
```

```
    cout << "one ";
```

```
    break;
```

```
case 2:
```

```
    cout << "two ";
```

```
    break;
```

```
case 3:
```

```
    cout << "three ";
```

```
    break;
```

```
case 4:
```

```
    cout << "four ";
```

```
    break;
```

```
case 5:
```

```
    cout << "five ";
```

```
    break;
```

```
case 6:
```

```
    cout << "six ";
```

```
    break;
```

```
case 7:
```

```
    cout << "seven ";
```

```
    break;
```

```
case 8:
```

```
    cout << "eight ";
```

```
    break;
```

```
case 9:
```

```
    cout << "nine ";
```

```
    break;
```

```
case 0:
```

```

        cout << "zero ";

        break;

    default:

        cout << "UnValid ";

        break;

    }

    // Divide the number by 10 to get the next number
    rev = rev / 10;

}

// Driver code
#include <iostream>

int main()
{
    int n = 12345;

    NumbertoCharacter(n);

    return 0;
}

```

```
one two three four five
```

Q8 Write a C++ program to generate all the prime numbers between 1 and n, where n is a value supplied by the user.

```
#include <iostream>
```

```
using namespace std;
```

```
int main() {
```

```
    int num, i, upto;
```



```

// Take input from user
cout << "Find prime numbers upto : ";
cin >> upto;

cout << endl << "All prime numbers upto " << upto << " are : " << endl;

for(num = 2; num <= upto; num++) {

    for(i = 2; i <= (num / 2); i++) {

        if(num % i == 0) {
            i = num;
            break;
        }
    }

    // If the number is prime then print it.
    if(i != num) {
        cout << num << " ";
    }
}

return 0;
}

```

```
Find prime numbers upto : 100
```

```
All prime numbers upto 100 are :
```

```
3 5 7 11 13 17 19 23 29 31 37 41 43 47 53 59 61 67 71 73 79 83 89 97
```

Q9 Write a C++ program to find both the largest and smallest number in a list of integers.

// Online C++ compiler to run C++ program online

```
#include<iostream>

using namespace std;

int main ()
{
    int arr[10], n, i, max, min;

    cout << "Enter the size of the array ";

    cin >> n;

    cout << "Enter the elements of the array ";

    for (i = 0; i < n; i++)

        cin >> arr[i];

    max = arr[0];

    for (i = 0; i < n; i++)
    {
        if (max < arr[i])

            max = arr[i];
    }

    min = arr[0];

    for (i = 0; i < n; i++)
    {
        if (min > arr[i])

            min = arr[i];
    }

    cout << "Largest element " << max<<endl;

    cout << "Smallest element " << min;

    return 0;
}
```

```
Enter the size of the array 5
Enter the elements of the array 1
2
3
4
5
Largest element 5
Smallest element 1
```

Q10 Write a C++ program to sort a list of numbers in ascending order.

```
#include <iostream>
```

```
using namespace std;
```

```
int main()
```

```
{
```

```
    int arr[100];
```

```
    int size, i, j, temp;
```

```
    // Reading the size of the array
```

```
    cout<<"Enter size of array: ";
```

```
    cin>>size;
```

```
    //Reading elements of array
```

```
    cout<<"Enter elements in array: ";
```

```
    for(i=0; i<size; i++)
```

```
    {
```

```
        cin>>arr[i];
```

```
    }
```

```
    //Sorting an array in ascending order
```

```
    for(i=0; i<size; i++)
```

```
    {
```

```
    for(j=i+1; j<size; j++)
    {
        //If there is a smaller element found on right of the array then swap it.
        if(arr[j] < arr[i])
        {
            temp = arr[i];
            arr[i] = arr[j];
            arr[j] = temp;
        }
    }
}

//Printing the sorted array in ascending order
cout<<"Elements of array in sorted ascending order:"<<endl;
for(i=0; i<size; i++)
{
    cout<<arr[i]<<endl;
}

return 0;
}
```

Enter size of array: 5

Enter elements in array: 1

5

2

4

3

9

Elements of array in sorted ascending order:

1

2

4

5

9