```
//Compute Factorial using Recursion
#include<iostream>
using namespace std;
long factorial (long a)
{
if (a > 1)
return (a * factorial (a-1));
else
return 1;
}
int main (){
long number;
cout<<"Enter Numer = ";
cin>>number;
cout<< number << "! = " << factorial (number);
return 0;
}
```

```
// Compute fibonacci using Recursion
#include <iostream>
using namespace std;

int fibonacci(int n) {
    if (n <= 1)
        return n;
    else
        return fibonacci(n - 1) + fibonacci(n - 2);
}

int main() {
    int n;
    cout << "Enter the position for Fibonacci sequence: ";
    cin >> n;
    cout << "Fibonacci number at position " << n << " is: "
<< fibonacci(n) << endl;
    return 0;
}
```

```
// Compute fibonacci using Iterative
#include <iostream>
using namespace std;
//void fibonacciIterative(int N) {
    int a = 0, b = 1, c;
    cout << "Fibonacci Series (Iterative): ";
    for (int i = 0; i < N; i++) {
        cout << a << " ";
        c = a + b;
        a = b;
        b = c;
    }
    cout << endl;
}
int main() {
    int N;
    cout << "Enter the number of terms: ";
    cin >> N;
    fibonacciIterative(N);
    return 0;
}
```

```
// Find the GCD of two numbers
#include <iostream>
using namespace std;

// Recursive function to find the GCD of two numbers
int gcd(int a, int b) {
    if (b == 0)
        return a;
    return gcd(b, a % b);
}

int main() {
    int num1, num2;
    cout << "Enter two numbers: ";
    cin >> num1 >> num2;

    cout << "GCD of " << num1 << " and " << num2 << "
is: " << gcd(num1, num2) << endl;
    return 0;
}
```