Lecture 07

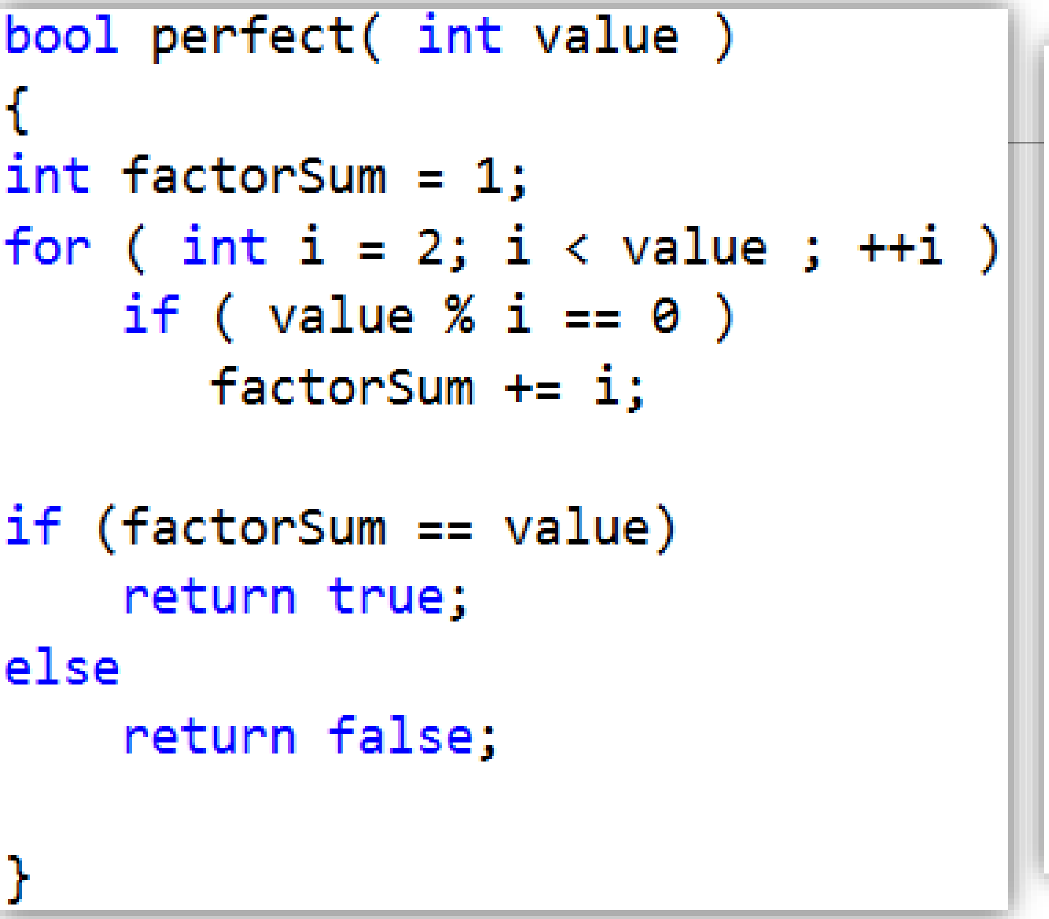
BUSHRA AYYASH

2017

Example:

Write a C++ function that receives one integer number as a parameter, to determine if this number is a perfect or not, then call this method in the Main() method.

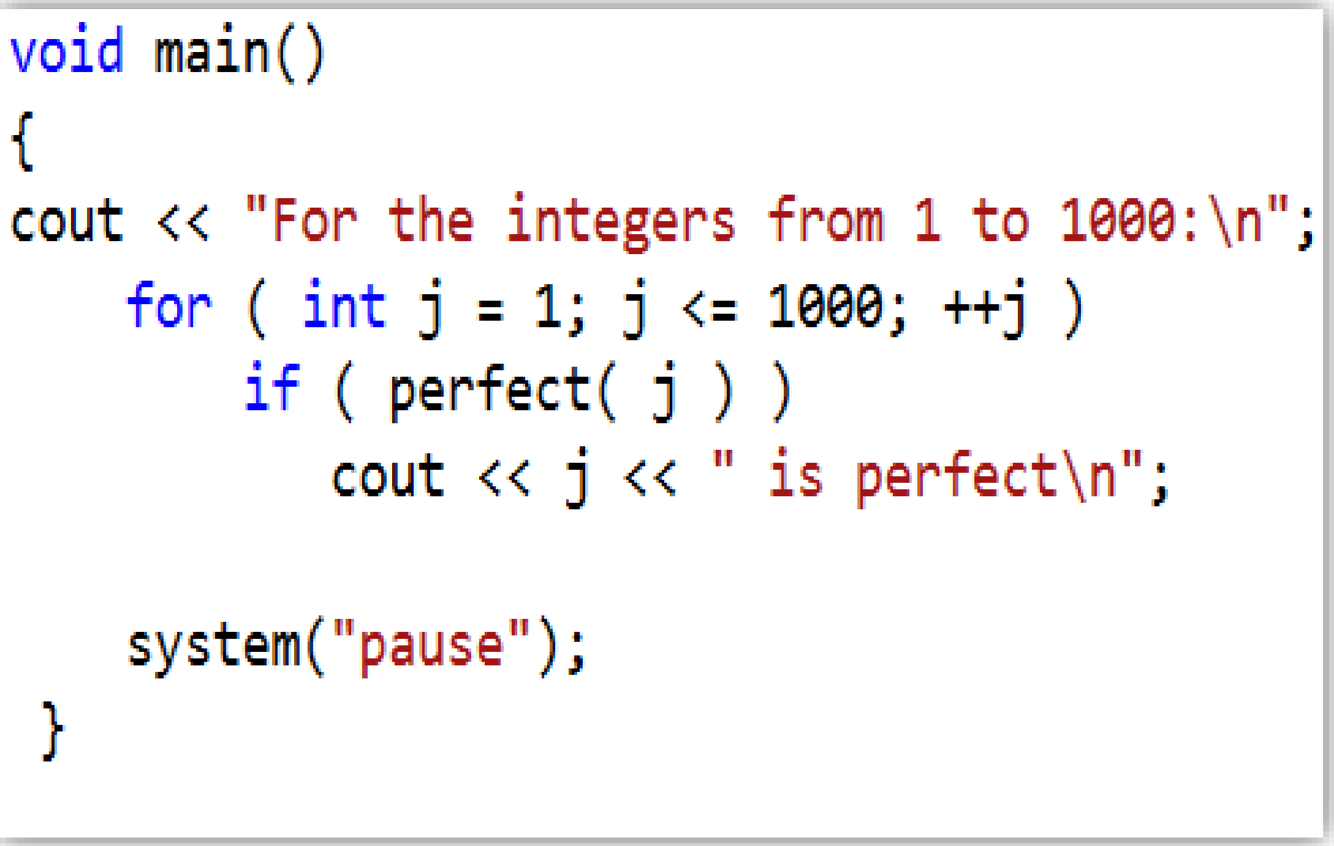
Hint: An integer number is said to be a perfect number if the sum of its factors, including 1 (but not the number itself), is equal to the number.



```
bool perfect( int value )
{
int factorSum = 1;
for ( int i = 2; i < value ; ++i )
    if ( value % i == 0 )
        factorSum += i;

if (factorSum == value)
    return true;
else
    return false;

}
```



```
void main()
{
cout << "For the integers from 1 to 1000:\n";
    for ( int j = 1; j <= 1000; ++j )
        if ( perfect( j ) )
            cout << j << " is perfect\n";

    system("pause");
 }
```

Recursive Functions

- The process in which a function calls itself is known as recursion.

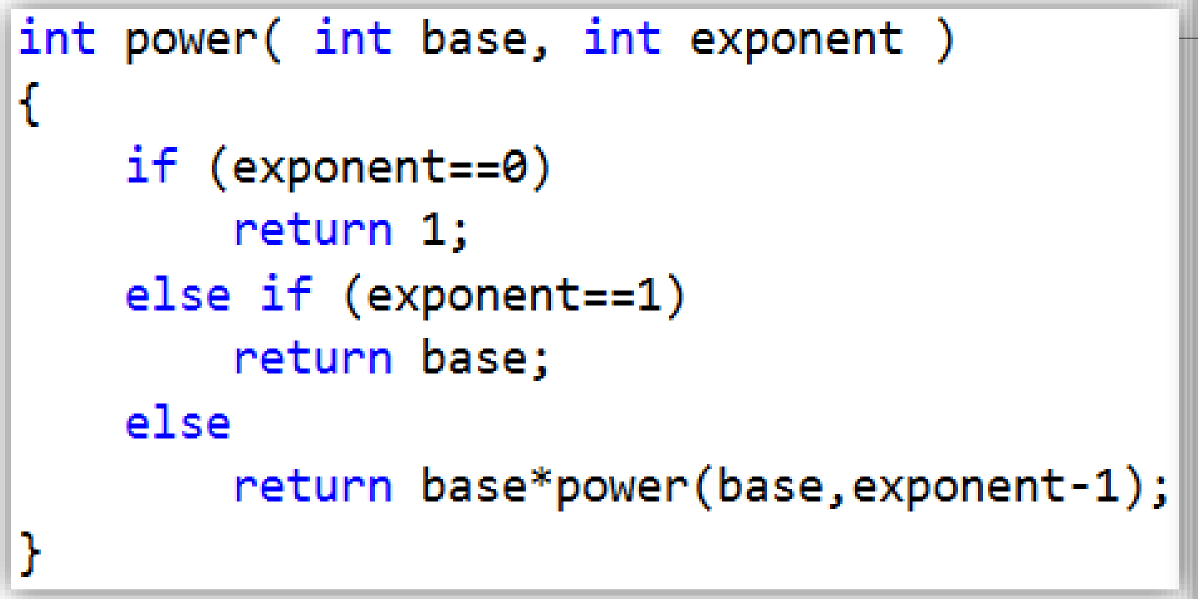
- and the corresponding function is called the **recursive function**.

- The purpose of recursion is to divide the problem into smaller problems till the base condition is reached.

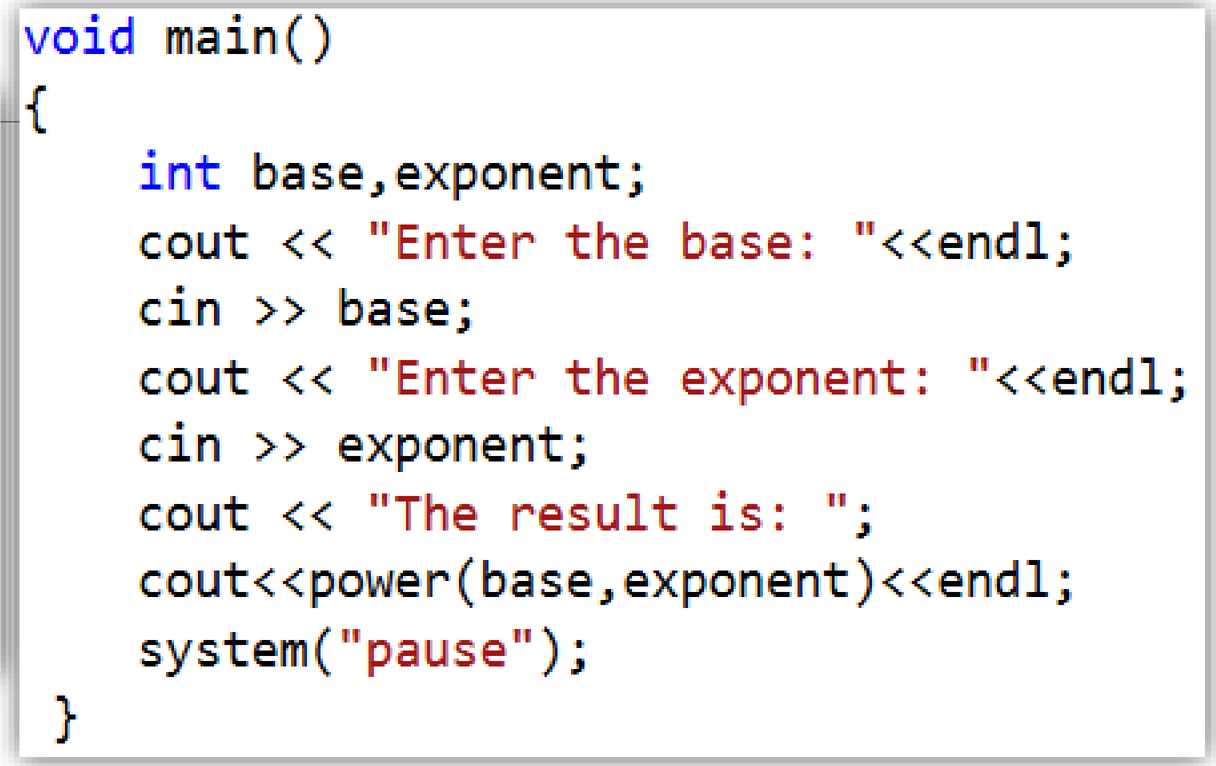
Example2:

Repeat the Power function that we implement in previous lectures by using recursion.

Note: exponent is considered here to be greater or equal 0



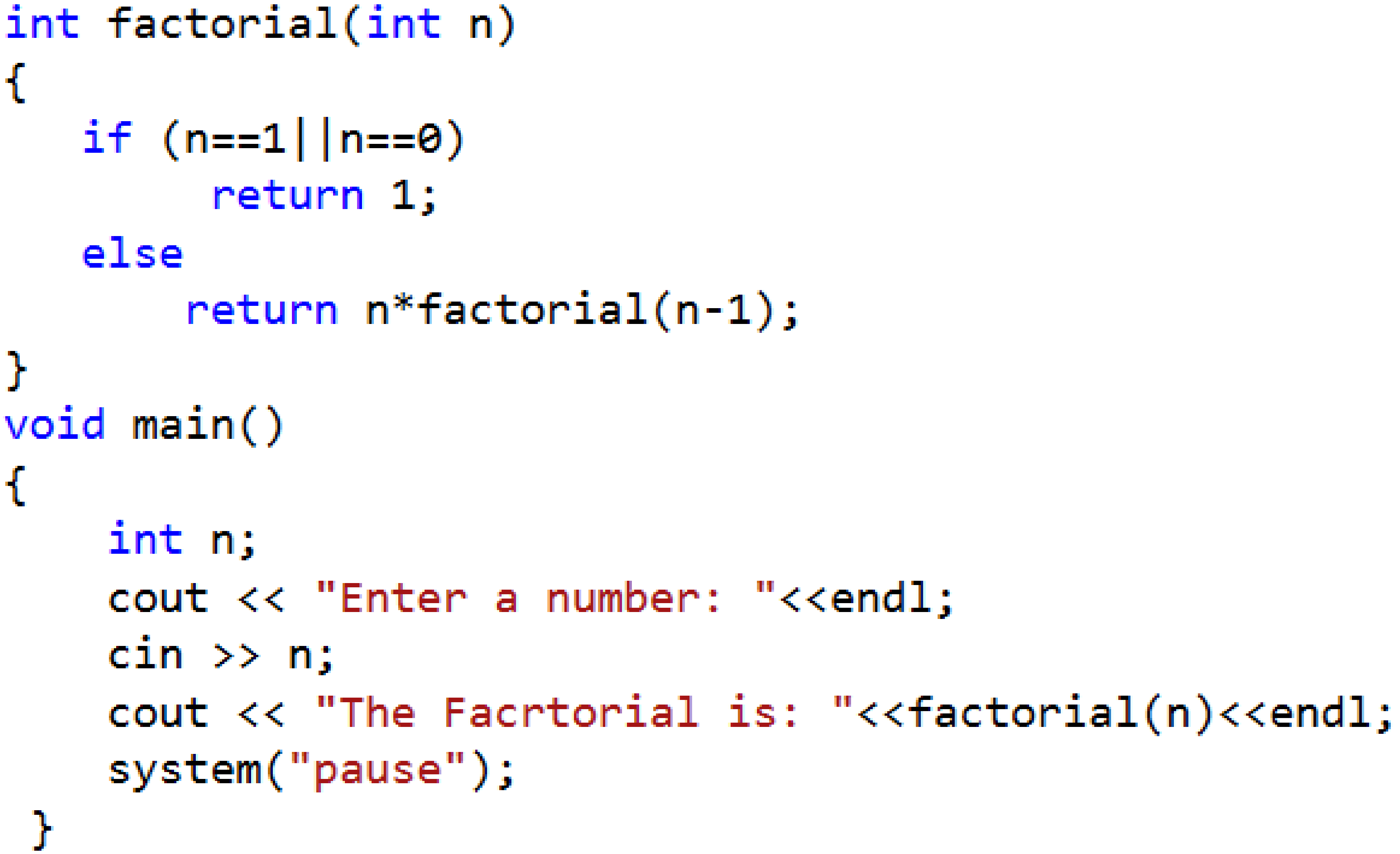
```cpp
int power( int base, int exponent )
{
    if (exponent==0)
        return 1;
    else if (exponent==1)
        return base;
    else
        return base*power(base,exponent-1);
}
```



```cpp
void main()
{
    int base,exponent;
    cout << "Enter the base: "<<endl;
    cin >> base;
    cout << "Enter the exponent: "<<endl;
    cin >> exponent;
    cout << "The result is: ";
    cout<<power(base,exponent)<<endl;
    system("pause");
 }
```

Example3:

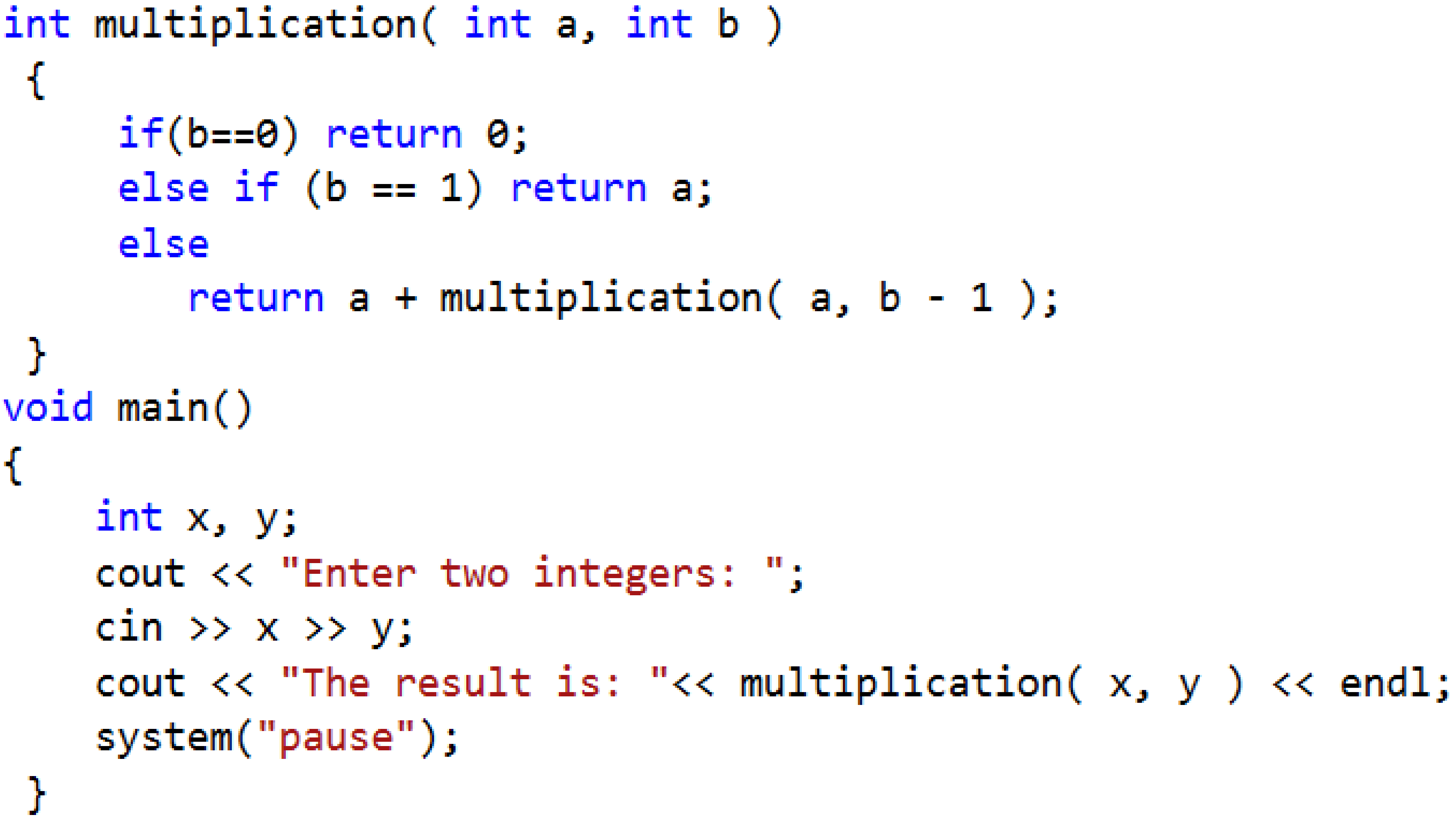
Repeat the Factorial function (for positive numbers) that we implement in previous lectures by using recursion.



```
int factorial(int n)
{
   if (n==1||n==0)
        return 1;
   else
       return n*factorial(n-1);
}
void main()
{
    int n;
    cout << "Enter a number: "<<endl;
    cin >> n;
    cout << "The Facrtorial is: "<<factorial(n)<<endl;
    system("pause");
 }
```

Example4:

Write a C++ program that multiply two positive numbers using recursion.



```
int multiplication( int a, int b )
 {
     if(b==0) return 0;
     else if (b == 1) return a;
     else
        return a + multiplication( a, b - 1 );
 }
void main()
{
    int x, y;
    cout << "Enter two integers: ";
    cin >> x >> y;
    cout << "The result is: "<< multiplication( x, y ) << endl;
    system("pause");
 }
```

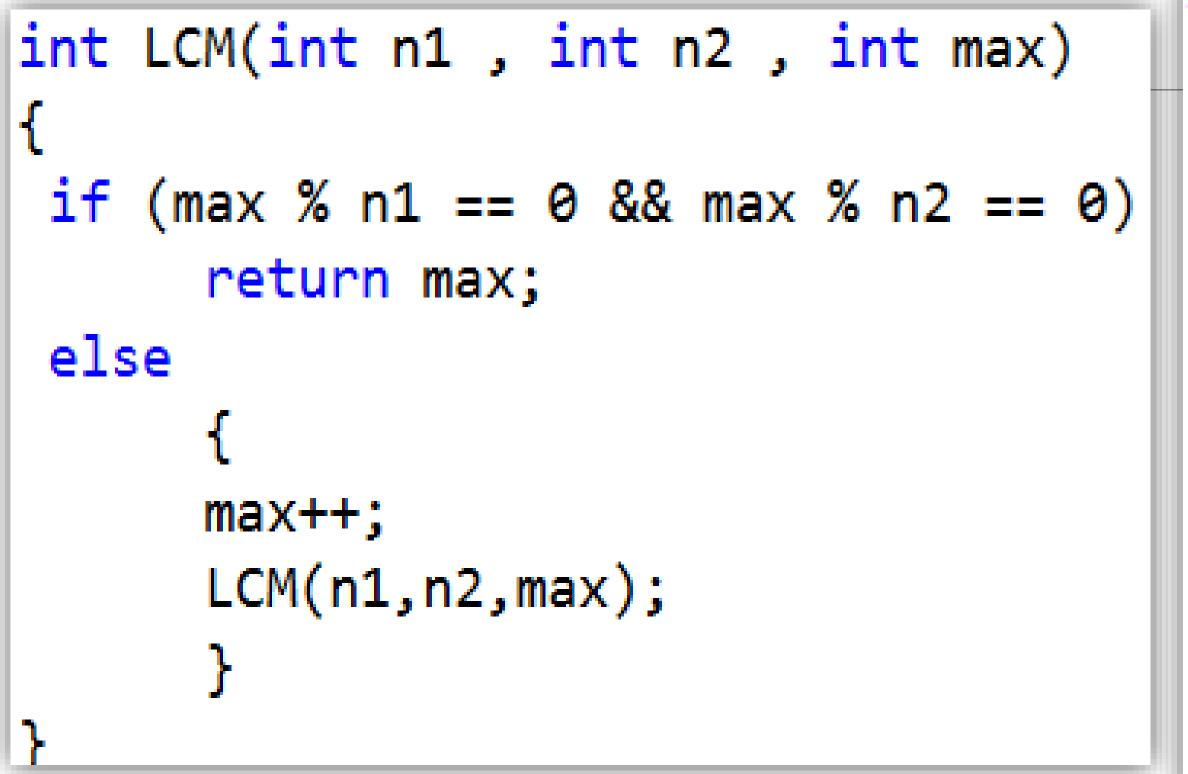
Example5:

Find the least common multiplier using recursion

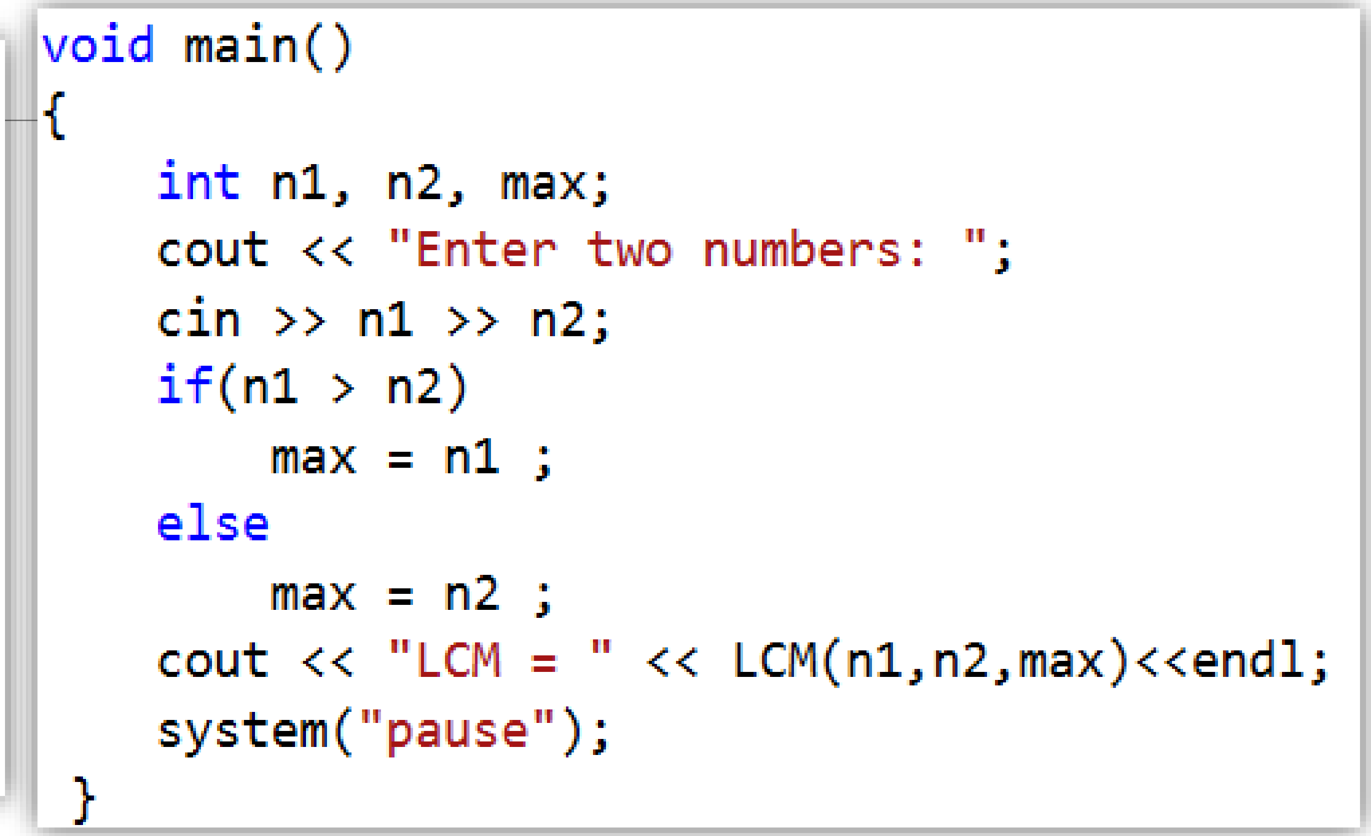
Hint: LCM of two integers a and b is the smallest positive integer that is divisible by both a and b.

Example: to find the LCM Of 4 & 6we say:

Multiples Of 4 Are: 4, 8, 12, 16, 20, 24,... and the multiples of 6 are: 6, 12, 18, 24, ... ,common multiples of 4 and 6 are simply the numbers that are in both lists: 12, 24, ... So, from this list of the first few common multiples of the numbers 4 and 6, their least common multiple is 12.



```
int LCM(int n1 , int n2 , int max)
{
 if (max % n1 == 0 && max % n2 == 0)
      return max;
 else
      {
      max++;
      LCM(n1,n2,max);
      }
}
```



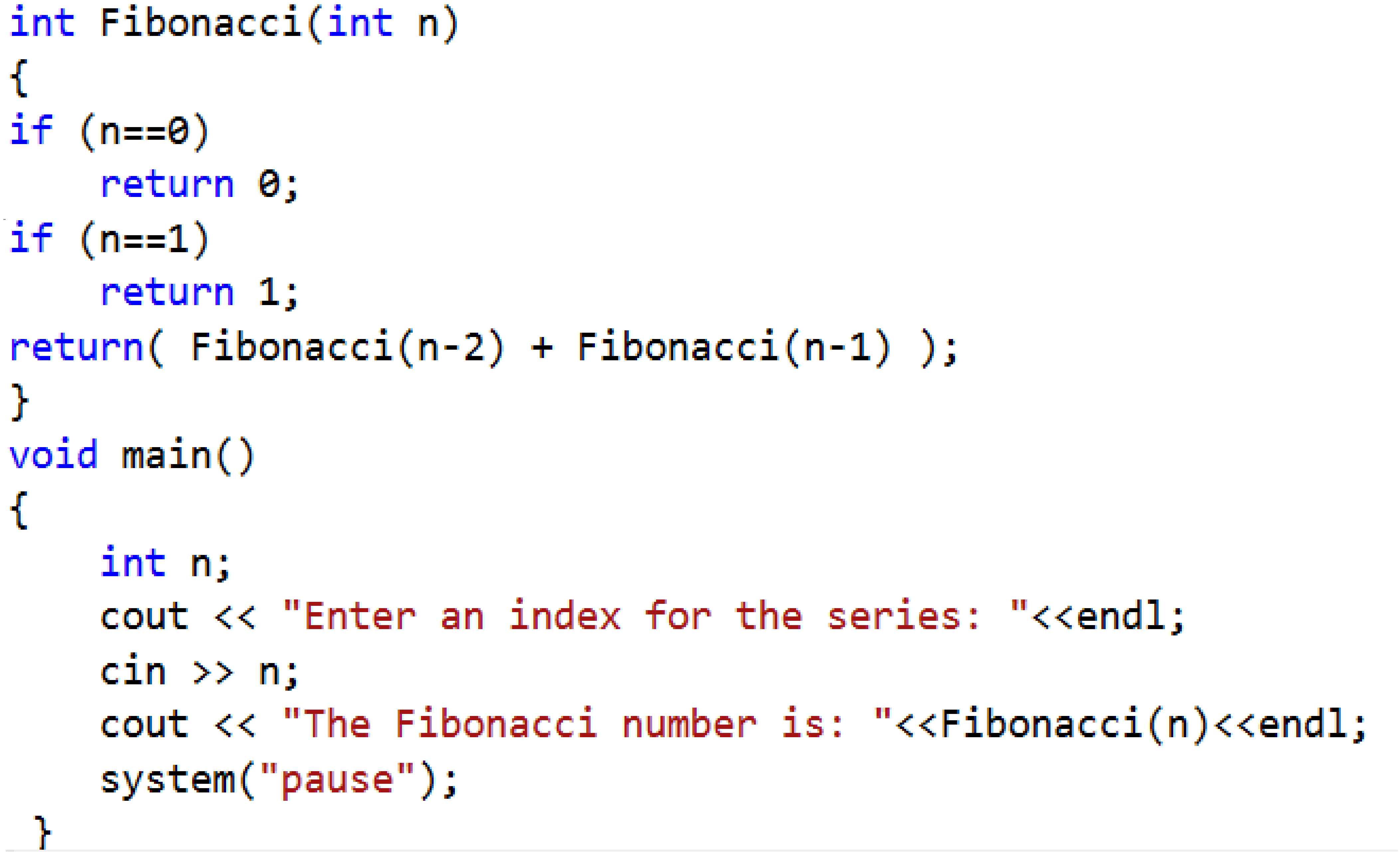
```
void main()
{
    int n1, n2, max;
    cout << "Enter two numbers: ";
    cin >> n1 >> n2;
    if(n1 > n2)
        max = n1 ;
    else
        max = n2 ;
    cout << "LCM = " << LCM(n1,n2,max)<<endl;
    system("pause");
 }
```

Example6:

Write a C++ program that get the Fibonacci number corresponding a given index in the series using recursion.

Note: Fibonacci series is an infinite series, which every number in it except the first two, is the sum of the two preceding ones.

0,1,1,2,3,5,8,13,21,etc..



```
int Fibonacci(int n)
{
if (n==0)
    return 0;
if (n==1)
    return 1;
return( Fibonacci(n-2) + Fibonacci(n-1) );
}
void main()
{
    int n;
    cout << "Enter an index for the series: "<<endl;
    cin >> n;
    cout << "The Fibonacci number is: "<<Fibonacci(n)<<endl;
    system("pause");
 }
```

Good Luck ^_^

BUSHRA AYYASH

2017