

# Some Important C++ Program Using Recursive Function

1. CALCULATE  $X^N$  USING RECURSION ----- 2 -
2. SUM OF N NATURAL NUMBERS USING RECURSION ----- 3 -
3. GCD OF TWO NUMBERS USING RECURSION ----- 4 -
4. FACTORIAL OF A NUMBER USING RECURSION ----- 6 -
5. FIBONACCI SERIES USING RECURSION AND NORMAL ----- 7 -
6. PRINT 1 TO N USING RECURSION ----- 9 -
7. PRINT N TO 1 USING RECURSION ----- 10 -



## 1. Calculate $x^n$ using Recursion

```
#include<iostream>

using namespace std;

int main()
{
    int x,n;

    long int power(int,int),result;

    cout<<"Enter Base : ";

    cin>>x;

    cout<<"Enter Power : ";

    cin>>n;

    result = power(x,n);

    cout<<"\n The "<<x<<"^"<<n<<" is : "<<result<<endl;
}

long int power(int x,int n){
    if(n==0)
        return 1;

    else
        return (x*power(x,n-1));
}
```

```
D:\program's\SECOND_SEM\C++(For_Project)\x^n_(Recursion).exe
Enter Base : 10
Enter Power : 5

The 10^5 is : 100000

-----
Process exited after 11.8 seconds with return value 0
Press any key to continue . . .
```

## 2. Sum of N Natural Numbers using Recursion

```
#include<iostream>
```

```
using namespace std;
```

```
int main(){
```

```
    int n,sum(int);
```

```
    cout<<"Enter N: ";
```

```
    cin>>n;
```

```
    cout<<"\n The Sum of Natural Number up to "<<n<<" is
    :"<<sum(n)<<endl;
```

```
}
```

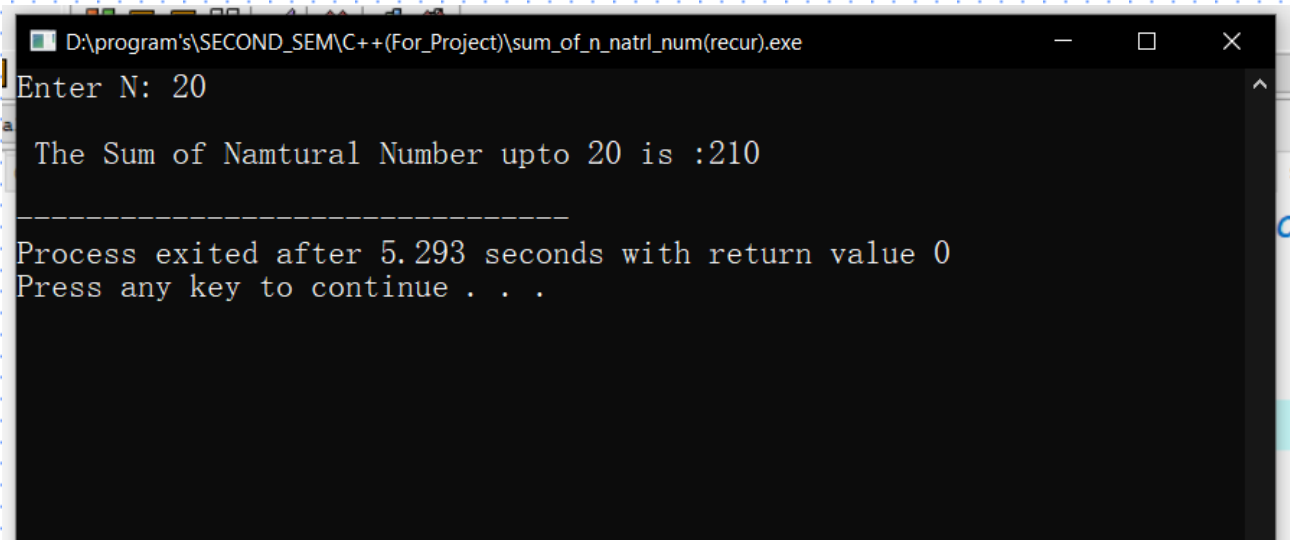
```
int sum(int n){
```

```
    if(n==1)
```

```
        return 1;
```

```
    else
```

```
    return (n+sum(n-1));  
}
```



```
D:\program's\SECOND_SEM\C++(For_Project)\sum_of_n_natr1_num(recur).exe  
Enter N: 20  
  
The Sum of Namtural Number upto 20 is :210  
-----  
Process exited after 5.293 seconds with return value 0  
Press any key to continue . . .
```

### 3. GCD of Two Numbers using Recursion

```
#include<iostream>  
  
using namespace std;  
  
int main(){  
    int a,b;  
    int GCD(int,int),result;  
    cout<<"Enter A : ";  
  
    cin>>a;  
  
    cout<<"Enter B : ";  
  
    cin>>b;
```

```
    result = GCD(a,b);

    cout<<"\n GCD of "<<a<<" and "<<b<<" is :"<<result<<endl;
}

int GCD(int a, int b){

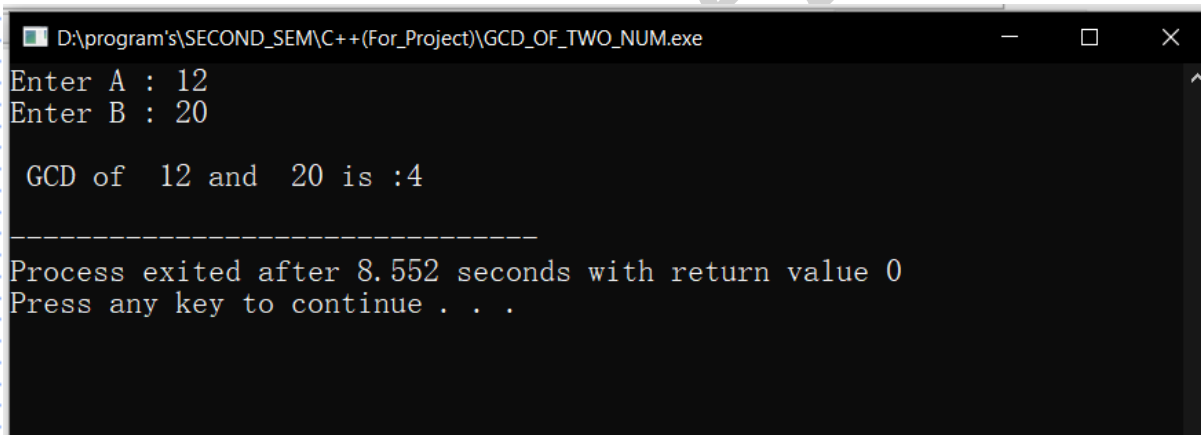
    if(b==0)

        return a;

    else

        return (GCD(b,a%b));

}
```



```
D:\program's\SECOND_SEM\C++(For_Project)\GCD_OF_TWO_NUM.exe
Enter A : 12
Enter B : 20

GCD of 12 and 20 is :4

-----
Process exited after 8.552 seconds with return value 0
Press any key to continue . . .
```

## 4. Factorial of a Number using Recursion

```
#include<iostream>
```

```
using namespace std;
```

```
int main(){
```

```
    int n;
```

```
    long int fact(int);
```

```
    cout<<" Enter N : ";
```

```
    cin>>n;
```

```
    cout<<" \n The factorial of "<<n<<" is :"<<fact(n)<<endl;
```

```
}
```

```
long int fact(int n)
```

```
{
```

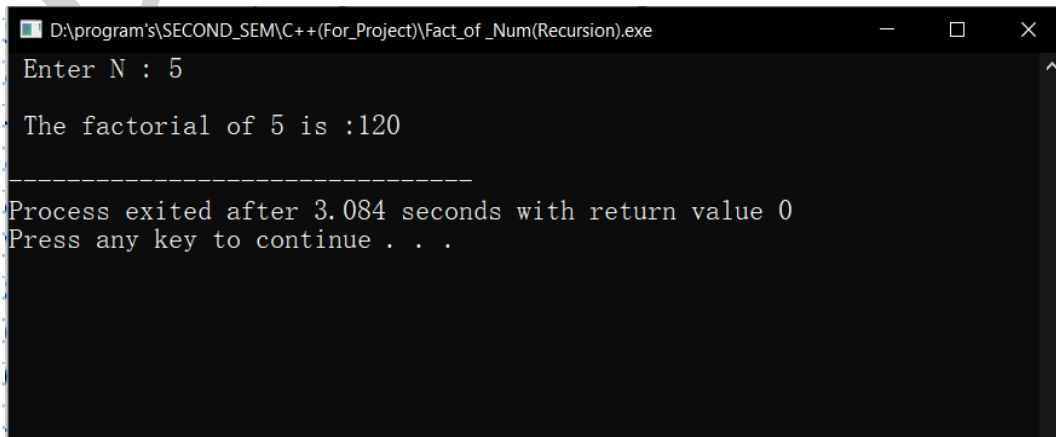
```
    if(n==0)
```

```
        return 1;
```

```
    else
```

```
        return (n*fact(n-1));
```

```
}
```



```
D:\program's\SECOND_SEM\C++(For_Project)\Fact_of_Num(Recursion).exe
Enter N : 5

The factorial of 5 is :120

-----
Process exited after 3.084 seconds with return value 0
Press any key to continue . . .
```

## 5. Fibonacci series using recursion and normal

```
#include<iostream>
```

```
using namespace std;
```

```
//Using Recursion up to given term -->fibonacci(n) = fibonacci(n-1)+fibonacci(n-2) | tail Point-->n=0,o-->n=1,1
```

```
int main(){
```

```
    int n,i=0,c=0;
```

```
    int fibo(int);
```

```
    system("cls");
```

```
    cout<<"N =";cin>>n;
```

```
    cout<<"Fibonacci Series :"<<endl;
```

```
    for(i=1;i<=n;i++){
```

```
        cout<<fibo(c++)<<" ";
```

```
    }
```

```
    return 0;
```

```
}
```

```
int fibo(int n){
```

```
    if(n==0)
```

```
        return 0;
```

```
    else if (n==1)
```

```
        return 1;
```

```

        else

            return (fibo(n-1)+fibo(n-2));

    }

    /*
    //-----
    //Normal up to n

    int main(){

        int a=0,b=1,c=0,n;

        cout<<" N= ";cin>>n;

        cout<<"Fibonacci Series up to N "<<n<<"are as follows :
"<<endl;

        while(c<=n){

            cout<<c<<" ";

            a = b;
            b = c;
            c = a+b;

        }

    }

    */

```



```
C:\Users\DELL\Desktop\fibonacci_usng_recursion.exe
N =10
Fibonacci Serise :
0 1 1 2 3 5 8 13 21 34
-----
Process exited after 3.241 seconds with return value 0
Press any key to continue . . .
```

## 6. Print 1 to N using Recursion

```
#include<iostream>
```

```
using namespace std;
```

```
int main(){
```

```
    int n;
```

```
    void disp(int);
```

```
    system("cls");
```

```
    cout<<"N = ";cin>>n;
```

```
    disp(n);
```

```
    return 0;
```

```
}
```

```
void disp(int n){
```

```
    if(n==0){
```

```
        return;
```

```
    }
```

```

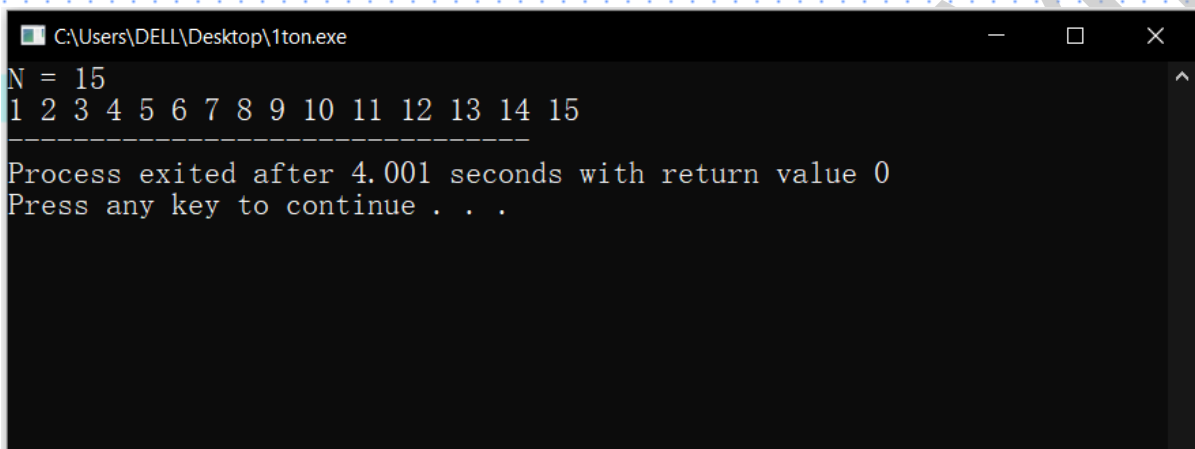
else

    disp(n-1);

cout<<n<<" ";

}

```



```

C:\Users\DELL\Desktop\1ton.exe
N = 15
1 2 3 4 5 6 7 8 9 10 11 12 13 14 15
-----
Process exited after 4.001 seconds with return value 0
Press any key to continue . . .

```

## 7. Print n to 1 using Recursion

```

#include<iostream>

using namespace std;

int main(){

    int n;

    void disp(int);

    system("cls");

    cout<<"N = ";cin>>n;

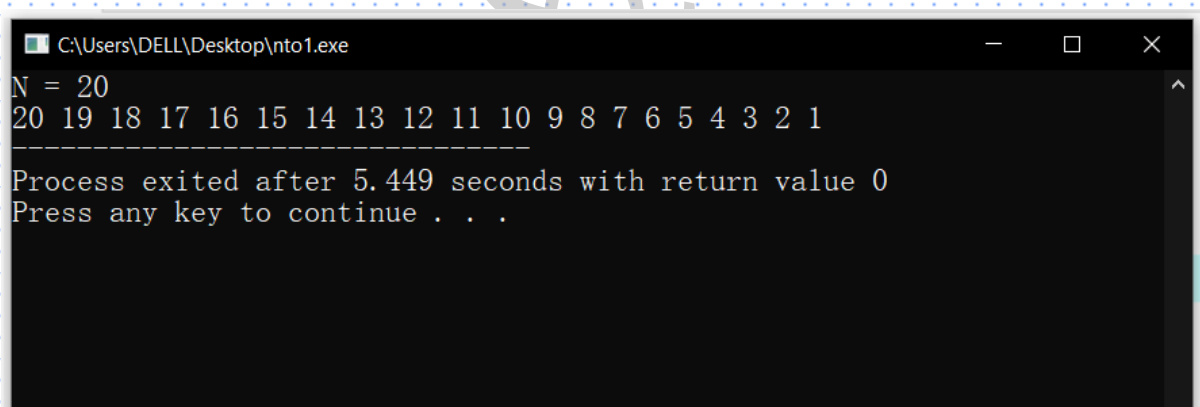
    disp(n);

```

```
        return 0;
    }

    void disp(int n){
        if(n==0){
            return;
        }
        else
            cout<<n<<" ";

        disp(n-1);
    }
}
```



```
C:\Users\DELL\Desktop\nto1.exe
N = 20
20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1
-----
Process exited after 5.449 seconds with return value 0
Press any key to continue . . .
```