**Experiment Number: 3b**

**Title:** Fibonacci Sequence Using Recursion in C++.

**Problem Statement**: Write C++ program to generate Fibonacci sequence using recursion.

**Algorithm:**

1. Read the number of terms in the Fibonacci sequence from user.
2. Starting from 0 till the number of terms minus 1 call the function fib() which calculates the Fibonacci of the number and display the Fibonacci of that number.
3. In fib function,
4. If the number is 0 or 1 return the number as it is.
5. If number is other than 0 or 1 then call the fib() function recursively as per the equation

fibterm=fib(n-1)+fib(n-2)

1. Return the fibterm.

**Code:**

#include<iostream>

using namespace std;

int fib(int n);

int main()

{

int n,i=0;

cout<<"\nEnter the number of terms for Fibonacci series:";

cin>>n;

cout<<"\nFibonacci sequence is as follows::";

while(i<n)

{

cout<<" "<<fib(i);

i++;

}

return 0;

}

int fib(int n)

{

int fibterm;

if((n==1)||(n==0))

return n;

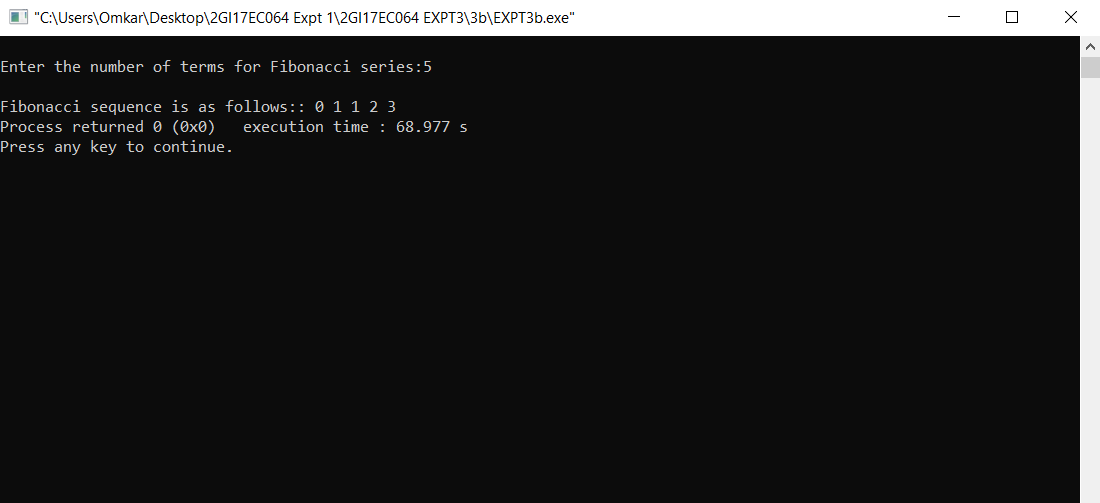
else

fibterm=fib(n-1)+fib(n-2);

return fibterm;

}

**Output:**



**Analysis/Limitations:**

There are following limitations of above implementation.

*Time Complexity:* We can observe that this implementation does a lot of repeated work. So this is a bad implementation for nth Fibonacci number