|  |  |
| --- | --- |
| **AssignmentNo.** | **1** |
| **Title** | To Implement non-recursive and recursive program to calculate Fibonacci numbers and analyze their time and space complexity. |
| **PROBLEM STATEMENT/DEFINITION** | Write a program non-recursive and recursive program to calculate Fibonacci numbers and analyze their time and space complexity. |
| **Objectives** | Understand and implement non-recursive and recursive programming and their time and space complexity. |
| **Software packages and hardware apparatus used** | Technology : Java/Python  Ubuntu /Linux  Latest Version of 64 bit Operating Systems, Open Source Fedora-GHz. 8 G.B.  RAM, 500 G.B. HDD, 15"Color Monitor, Keyboard, Mouse |
| **References** | 1. <https://www.geeksforgeeks.org/program-for-nth-fibonacci-number/> 2. <https://www.scaler.com/topics/fibonacci-series-in-c/> |
|  |  |

**Concepts Related Theory**

A **Fibonacci sequence**is the sequence of integer in which each element in the sequence is the sum of the two previous elements.

Fibonacci series starts from two numbers − **F0 & F1**. The initial values of F0 & F1 can be taken 0, 1 or 1, 1 respectively.

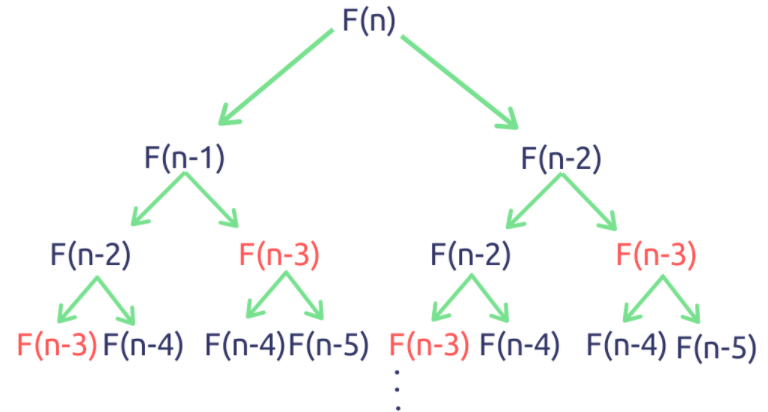
***Fn = Fn-1 + Fn-2***

E.g.

F8 = 0,  1,  1,  2,  3,   8,  13  or, F8 = 1,  1,  2,  3,  5,  8,  13,  21

|  |
| --- |
| A **recursion tree**is a tree that is generated by tracing the execution of a recursive algorithm. A recursion tree shows the relationship between calls to the algorithm. Each item in the tree represents a call to the algorithm. |

**Fibonacci Recursion Tree**



## Without Recursion:

We discussed how to implement the Fibonacci series using recursion. The Fibonacci series in C can be implemented without using recursion also. Let us look at the different ways of implementing the Fibonacci series without using recursion.

****Using dynamic programming****

In dynamic programming, We will store all the previously calculated values of the Fibonacci numbers in ****an array****. We know that the array is zero-indexed. Therefore the Fibonacci numbers are stored in the n-1 array index. For example, the 2nd Fibonacci number is stored in the 1st index of the array.

**Algorithm:**

**Recursive algorithm to get Fibonacci sequence:**

1. START

2. Input the non-negative integer ‘n’

3. If (n==o || n==1)

        return n;

    else

        return fib(n-1)+fib(n-2);

4. Print, nth Fibonacci number

5. END

**Non-Recursive algorithm to get Fibonacci sequence:**

* Create 2 variables that will store the value of last and second last number.
* Initialize both the variables with 0.
* Now start a loop till N and for every i-th index,
  + Print Sum of last and second last i.e SUM = LAST + SECOND\_LAST
  + Assign last to second last i.e. SECOND\_LAST = LAST
  + Assign Sum to last i.e. LAST = SUM

**Pseudocode:**

**Non-recursive Fibonacci Series:**

procedure fibonacci : fib\_num

IF fib\_num less than 1

DISPLAY 0

IF fib\_num equals to 1

DISPLAY 1

IF fib\_num equals to 2

DISPLAY 1, 1

IF fib\_num greater than 2

Pre = 1,

Post = 1,

DISPLAY Pre, Post

FOR 0 to fib\_num-2

Fib = Pre + Post

DISPLAY Fib

Pre = Post

Post = Fib

END FOR

END IF

end procedure

**Recursive Fibonacci Series:**

Fibo(n)

Begin

if n <= 1 then

Return n;

else

Return Call Fibo(n-1) + Call Fibo(n-2);

endif

End

**Time & Space complexity*:***

For Recursive,

****Time Complexity:**** *O*(*n* 2*n*)

****Space Complexity:**** *O*(1)

For Non- Recursive,

****Time Complexity:**** O(n)

****Space Complexity:**** O(n)

## Conclusion

* The Fibonacci number can be found out by taking the sum of the previous two Fibonacci terms. The first and second digit of the series is fixed to 0 and 1, respectively.
* The series 0, 1, 1, 2, 3, 5, 8, 13, 21, 34,......... is known as the Fibonacci series.