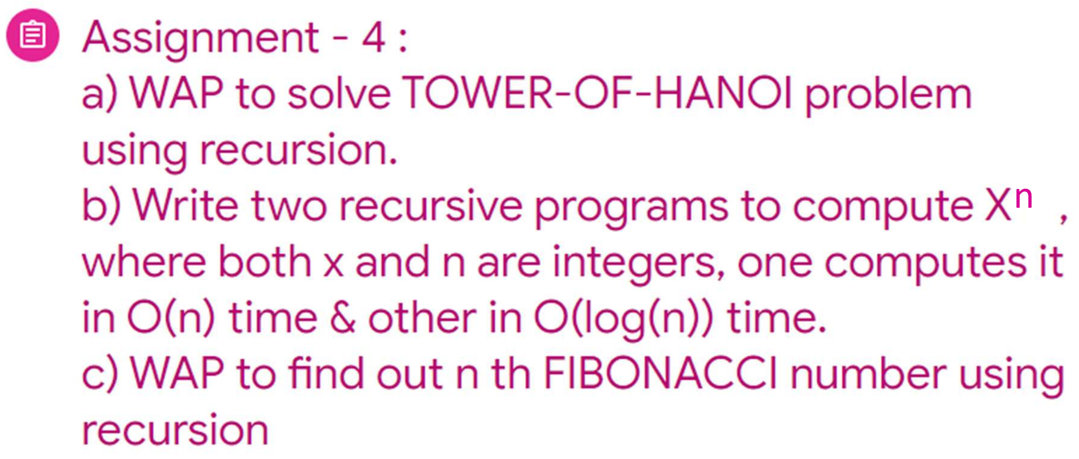


Assignment - 4 :
a) WAP to solve TOWER-OF-HANOI problem using recursion.
b) Write two recursive programs to compute X^n , where both x and n are integers, one computes it in O(n) time & other in O(log(n)) time.
c) WAP to find out n th FIBONACCI number using recursion

(a) WAP to solve TOWER OF HANOI problem using recursion –

<u>Algorithm</u> –

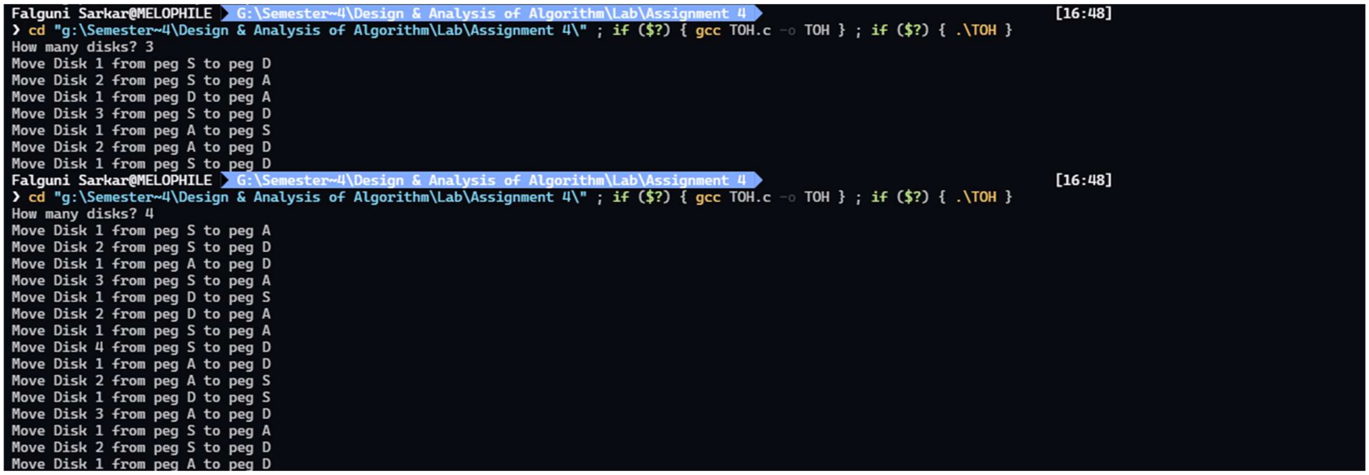
Procedure: tower(int n, char beg, char aux, char end)
Inputs: n – no. of disks.
beg – first (or starting) peg.
aux – second (or intermediate) peg.
end – last (or final) peg.
Output: Will return the steps of moving disk from first peg to final peg.

```
if n == 1 do
        print ("Move Disk n from peg beg to peg end")
        return.
end if.
call tower(n-1, beg, end, aux)
print ("Move Disk n from peg beg to peg end")
call tower(n-1, beg, end, aux)
return.
```

Program –

```
#include <stdio.h>

void tower(int n, char beg, char aux, char end)
{
        if(n==1)
        {
                printf("Move Disk %d from peg %c to peg %c\n", n, beg, end);
                return;
        }
        tower(n-1,beg,end,aux);
        printf("Move Disk %d from peg %c to peg %c\n", n, beg, end);
        tower(n-1,aux,beg,end);
        return;
}

int main()
{
        int n;
        void tower(int, char, char, char);
        printf("How many disks? ");
        scanf("%d",&n);
        if(n>0)
                tower(n,'S','A','D');
        else
                printf("Invalid! Press any key to exit.\n");
        return(0);
}
```

Output –



```
Falguni Sarkar@MELOPHILE  G:\Semester~4\Design & Analysis of Algorithm\Lab\Assignment 4                      [16:48]
❯ cd "g:\Semester~4\Design & Analysis of Algorithm\Lab\Assignment 4\" ; if ($?) { gcc TOH.c -o TOH } ; if ($?) { .\TOH }
How many disks? 3
Move Disk 1 from peg S to peg D
Move Disk 2 from peg S to peg A
Move Disk 1 from peg D to peg A
Move Disk 3 from peg S to peg D
Move Disk 1 from peg A to peg S
Move Disk 2 from peg A to peg D
Move Disk 1 from peg S to peg D
Falguni Sarkar@MELOPHILE  G:\Semester~4\Design & Analysis of Algorithm\Lab\Assignment 4                      [16:48]
❯ cd "g:\Semester~4\Design & Analysis of Algorithm\Lab\Assignment 4\" ; if ($?) { gcc TOH.c -o TOH } ; if ($?) { .\TOH }
How many disks? 4
Move Disk 1 from peg S to peg A
Move Disk 2 from peg S to peg D
Move Disk 1 from peg A to peg D
Move Disk 3 from peg S to peg A
Move Disk 1 from peg D to peg S
Move Disk 2 from peg D to peg A
Move Disk 1 from peg S to peg A
Move Disk 4 from peg S to peg D
Move Disk 1 from peg A to peg D
Move Disk 2 from peg A to peg S
Move Disk 1 from peg D to peg S
Move Disk 3 from peg A to peg D
Move Disk 1 from peg S to peg A
Move Disk 2 from peg S to peg D
Move Disk 1 from peg A to peg D
```

(b) Write two recursive programs to compute X^n, where both x and n are integers, one computes it in O(n) time & other in O(log(n)) time –

Time Complexity: [O(n)] –

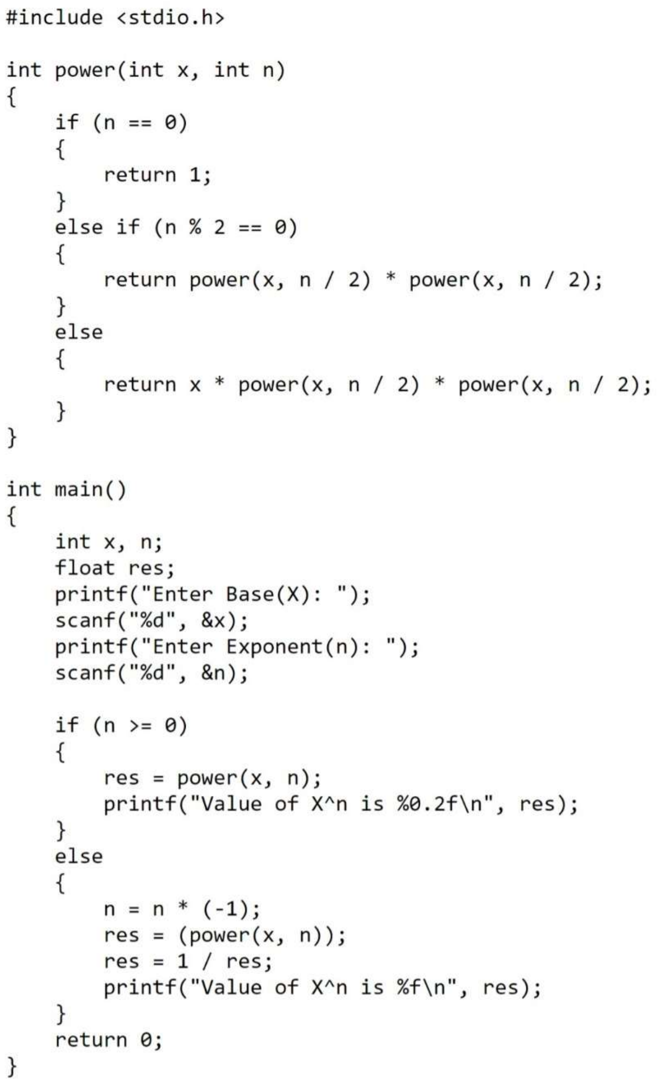
Algorithm–

Procedure: power(int x, int n)
Inputs: x – base.
n – exponent
Output: Will return the base raised to the power given.

[**Note**: For the negative powers, we will take the positive power and return the positive value and in the main function, the returned value will be divided by 1 to procure the final value.]

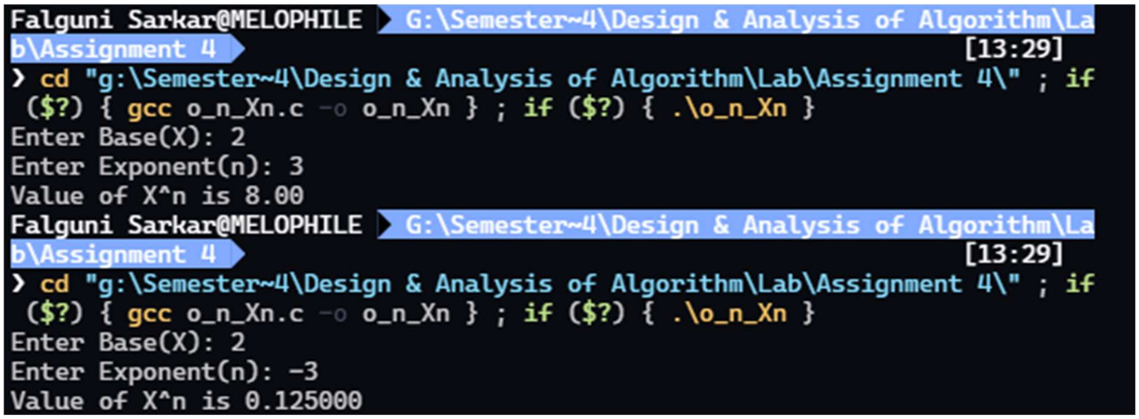
```
1. if n == 0 do
2.     return 1.
3. else if n%2 == 0 do
4.     return power(x, n / 2) * power(x, n / 2).
5. else do
6.     return x*power(x, n / 2) * power(x, n / 2).
7. end if.
```

Program –



```
#include <stdio.h>

int power(int x, int n)
{
    if (n == 0)
    {
        return 1;
    }
    else if (n % 2 == 0)
    {
        return power(x, n / 2) * power(x, n / 2);
    }
    else
    {
        return x * power(x, n / 2) * power(x, n / 2);
    }
}

int main()
{
    int x, n;
    float res;
    printf("Enter Base(X): ");
    scanf("%d", &x);
    printf("Enter Exponent(n): ");
    scanf("%d", &n);

    if (n >= 0)
    {
        res = power(x, n);
        printf("Value of X^n is %0.2f\n", res);
    }
    else
    {
        n = n * (-1);
        res = (power(x, n));
        res = 1 / res;
        printf("Value of X^n is %f\n", res);
    }
    return 0;
}
```

Output –



```
Falguni Sarkar@MELOPHILE  G:\Semester~4\Design & Analysis of Algorithm\La
b\Assignment 4                                                     [13:29]
❯ cd "g:\Semester~4\Design & Analysis of Algorithm\Lab\Assignment 4\" ; if
 ($?) { gcc o_n_Xn.c -o o_n_Xn } ; if ($?) { .\o_n_Xn }
Enter Base(X): 2
Enter Exponent(n): 3
Value of X^n is 8.00
Falguni Sarkar@MELOPHILE  G:\Semester~4\Design & Analysis of Algorithm\La
b\Assignment 4                                                     [13:29]
❯ cd "g:\Semester~4\Design & Analysis of Algorithm\Lab\Assignment 4\" ; if
 ($?) { gcc o_n_Xn.c -o o_n_Xn } ; if ($?) { .\o_n_Xn }
Enter Base(X): 2
Enter Exponent(n): -3
Value of X^n is 0.125000
```

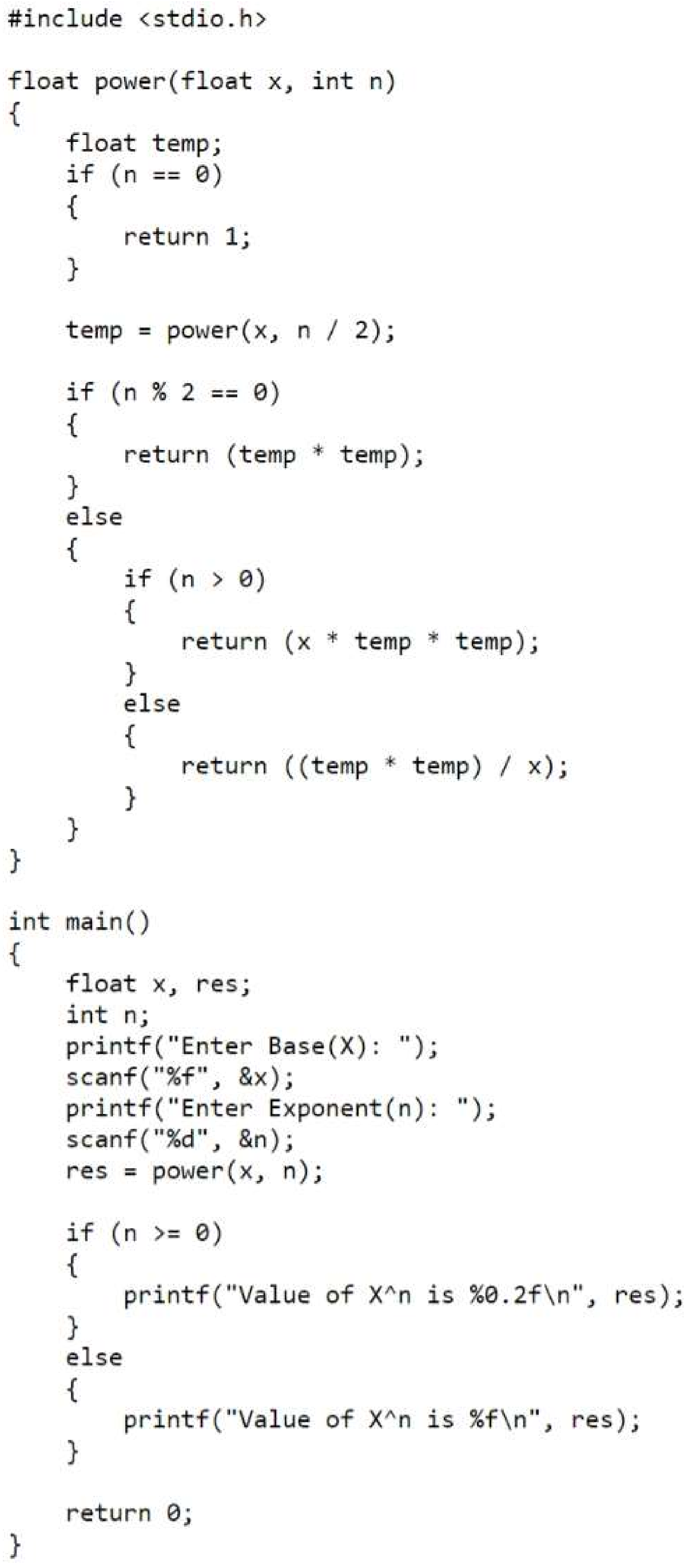
Time Complexity: [O(log n)] –

Algorithm –

Procedure: power(float x, int n)
Inputs: x – base.
n – exponent
Output: Will return the base raised to the power given.

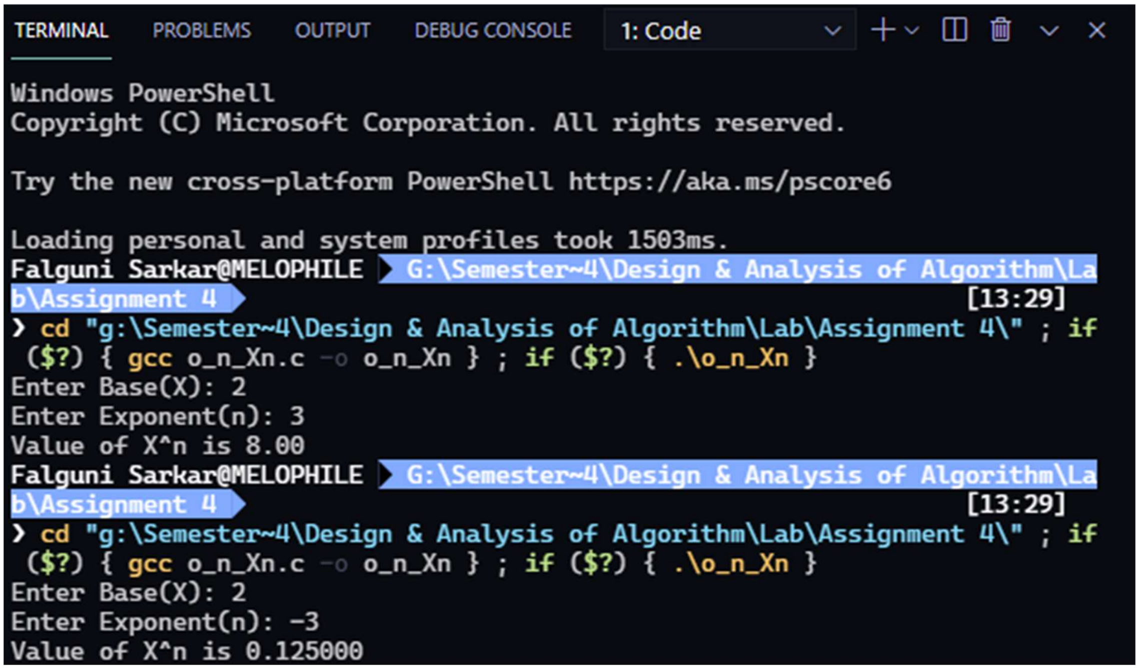
```
if n == 0 do
        return 1.
end if.
temp ← power(x, n/2).
if n%2 == 0 do
        return temp * temp.
else do
        if n>0 do
                return x* temp * temp.
        else do
                return (temp * temp) / x;
        end if.
end if.
```

Program –



```
#include <stdio.h>

float power(float x, int n)
{
    float temp;
    if (n == 0)
    {
        return 1;
    }

    temp = power(x, n / 2);

    if (n % 2 == 0)
    {
        return (temp * temp);
    }
    else
    {
        if (n > 0)
        {
            return (x * temp * temp);
        }
        else
        {
            return ((temp * temp) / x);
        }
    }
}

int main()
{
    float x, res;
    int n;
    printf("Enter Base(X): ");
    scanf("%f", &x);
    printf("Enter Exponent(n): ");
    scanf("%d", &n);
    res = power(x, n);

    if (n >= 0)
    {
        printf("Value of X^n is %0.2f\n", res);
    }
    else
    {
        printf("Value of X^n is %f\n", res);
    }

    return 0;
}
```

Output –



```
Windows PowerShell
Copyright (C) Microsoft Corporation. All rights reserved.

Try the new cross-platform PowerShell https://aka.ms/pscore6

Loading personal and system profiles took 1503ms.
Falguni Sarkar@MELOPHILE  G:\Semester~4\Design & Analysis of Algorithm\La
b\Assignment 4                                                      [13:29]
❯ cd "g:\Semester~4\Design & Analysis of Algorithm\Lab\Assignment 4\" ; if
 ($?) { gcc o_n_Xn.c -o o_n_Xn } ; if ($?) { .\o_n_Xn }
Enter Base(X): 2
Enter Exponent(n): 3
Value of X^n is 8.00
Falguni Sarkar@MELOPHILE  G:\Semester~4\Design & Analysis of Algorithm\La
b\Assignment 4                                                      [13:29]
❯ cd "g:\Semester~4\Design & Analysis of Algorithm\Lab\Assignment 4\" ; if
 ($?) { gcc o_n_Xn.c -o o_n_Xn } ; if ($?) { .\o_n_Xn }
Enter Base(X): 2
Enter Exponent(n): -3
Value of X^n is 0.125000
```

(c) WAP to find out n^th^ FIBONACCI number using recursion –

Algorithm –

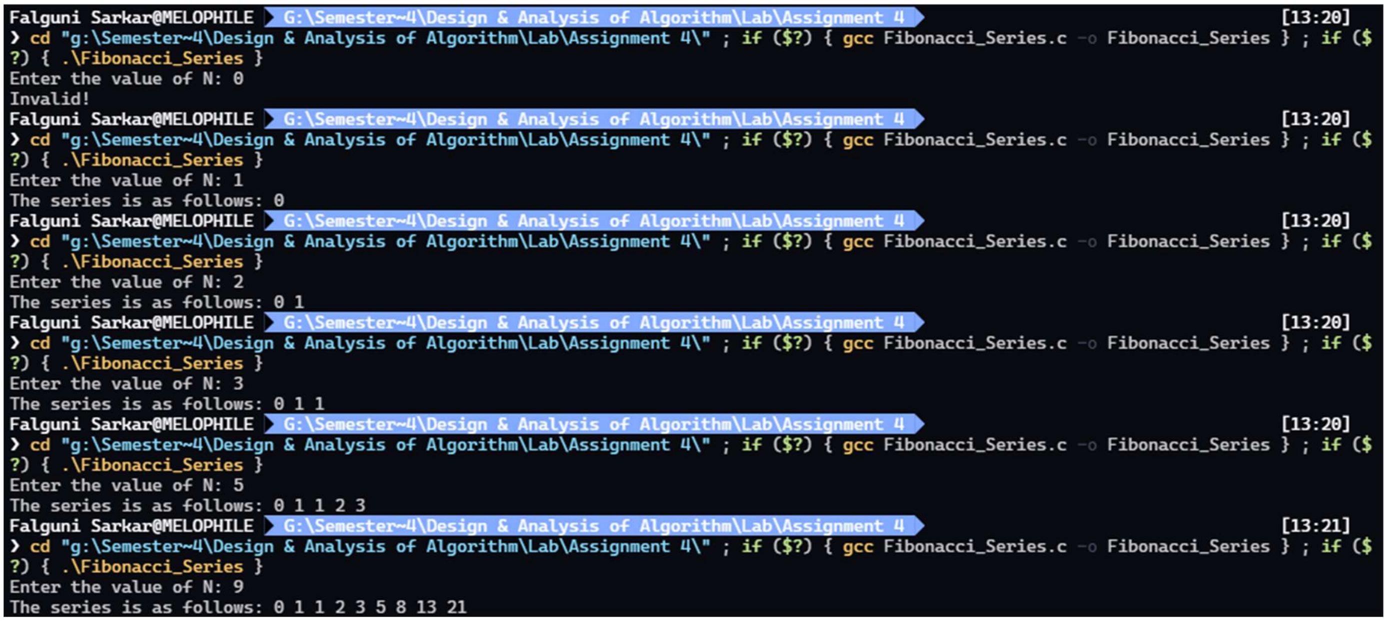
Procedure: fibonacci(int n)
Inputs: n – length of the sequence.
Output: Returns the Fibonacci series.

```
if n == 0 OR n==1 do
        return n.
else do
        return fibonacci(n-1) + fibonacci(n-2).
 end if.
```

Program –

```
#include<stdio.h>
int fibonacci(int n)
{
    if(n==0 || n==1)
    {
        return n;
    }
    else
    {
        return (fibonacci(n-1)+fibonacci(n-2));
    }
}
int main()
{
    int n,i,m=0;
    printf("Enter the value of N: ");
    scanf("%d",&n);
    if(n<=0)
    {
        printf("Invalid!\n");
    }
    else
    {
        printf("The series is as follows: ");
        for(i=1;i<=n;i++)
        {
            printf("%d ",fibonacci(m));
            m++;
        }
    }
    return 0;
}
```

Output –



```
Falguni Sarkar@MELOPHILE  G:\Semester-4\Design & Analysis of Algorithm\Lab\Assignment 4                    [13:20]
❯ cd "g:\Semester-4\Design & Analysis of Algorithm\Lab\Assignment 4\" ; if ($?) { gcc Fibonacci_Series.c -o Fibonacci_Series } ; if ($
?) { .\Fibonacci_Series }
Enter the value of N: 0
Invalid!
Falguni Sarkar@MELOPHILE  G:\Semester-4\Design & Analysis of Algorithm\Lab\Assignment 4                    [13:20]
❯ cd "g:\Semester-4\Design & Analysis of Algorithm\Lab\Assignment 4\" ; if ($?) { gcc Fibonacci_Series.c -o Fibonacci_Series } ; if ($
?) { .\Fibonacci_Series }
Enter the value of N: 1
The series is as follows: 0
Falguni Sarkar@MELOPHILE  G:\Semester-4\Design & Analysis of Algorithm\Lab\Assignment 4                    [13:20]
❯ cd "g:\Semester-4\Design & Analysis of Algorithm\Lab\Assignment 4\" ; if ($?) { gcc Fibonacci_Series.c -o Fibonacci_Series } ; if ($
?) { .\Fibonacci_Series }
Enter the value of N: 2
The series is as follows: 0 1
Falguni Sarkar@MELOPHILE  G:\Semester-4\Design & Analysis of Algorithm\Lab\Assignment 4                    [13:20]
❯ cd "g:\Semester-4\Design & Analysis of Algorithm\Lab\Assignment 4\" ; if ($?) { gcc Fibonacci_Series.c -o Fibonacci_Series } ; if ($
?) { .\Fibonacci_Series }
Enter the value of N: 3
The series is as follows: 0 1 1
Falguni Sarkar@MELOPHILE  G:\Semester-4\Design & Analysis of Algorithm\Lab\Assignment 4                    [13:20]
❯ cd "g:\Semester-4\Design & Analysis of Algorithm\Lab\Assignment 4\" ; if ($?) { gcc Fibonacci_Series.c -o Fibonacci_Series } ; if ($
?) { .\Fibonacci_Series }
Enter the value of N: 5
The series is as follows: 0 1 1 2 3
Falguni Sarkar@MELOPHILE  G:\Semester-4\Design & Analysis of Algorithm\Lab\Assignment 4                    [13:21]
❯ cd "g:\Semester-4\Design & Analysis of Algorithm\Lab\Assignment 4\" ; if ($?) { gcc Fibonacci_Series.c -o Fibonacci_Series } ; if ($
?) { .\Fibonacci_Series }
Enter the value of N: 9
The series is as follows: 0 1 1 2 3 5 8 13 21
```