Course Code	**:**	**BCSL-159**
Course Title	**:**	**Introduction to Algorithm Design Lab**
Assignment Number	**:**	**BCA_NEW/BCAOL(V)L-159/Assign/2026-27**
Maximum Marks	**:**	**100**
Last Date of Submission	**:**	**31st October, 2026 (For July 2026 Session)**
	:	**30th April, 2027 (For January 2027 Session)**

There are 8 questions of 10 marks each in this assignment, carrying a total of 80 marks. The remaining 20 marks are for viva voce. Please go through the guidelines regarding assignments given in the Programme Guide for the format of the presentation. Write all the programs in the C' language.

Q1: Write a C program to generate the Fibonacci sequence using recursion. **(10 Marks)**

Q2: Write a recursive C program to solve the Tower of Hanoi problem. Also, analyse its time complexity. **(10 Marks)**

Q3: Write a C program to evaluate a polynomial efficiently using Horner's method. **(10 Marks)**

Q4: Write a C program to multiply two n×n matrices using nested loops. **(10 Marks)**

Q5: Write a C program to solve the Fractional Knapsack problem and maximise the total profit using a greedy approach. **(10 Marks)**

Q6: Write a C program to construct a Huffman tree and generate the corresponding Huffman codes. **(10 Marks)**

Q7: Write a recursive C program to perform large integer multiplication using the Karatsuba multiplication algorithm. **(10 Marks)**

Q8: Write a C program to detect a cycle in an undirected graph. **(10 Marks)**

Q1: Write a C program to generate the Fibonacci sequence using recursion. (10 Marks)

Answer

Fibonacci numbers begin with 0 and 1. Every later term is the sum of the previous two: F(n) = F(n - 1) + F(n - 2). In recursion, the function calls itself with two smaller arguments until it reaches the base cases F(0) = 0 and F(1) = 1.

The following C program prints the first n terms. The input is limited to 40 terms because the direct recursive method repeats many calculations. This makes the basic recursion easy to understand and keeps the example responsive.

C program

```
#include <stdio.h>

/* Fibonacci numbers: F(0)=0, F(1)=1. */
unsigned long long fib(unsigned int n)
{
    if (n <= 1) return n;
    return fib(n - 1) + fib(n - 2);
}

int main(void)
{
    unsigned int n, i;
    printf("Number of terms (1 to 40): ");
    if (scanf("%u", &n) != 1 || n < 1 || n > 40) {
        puts("Invalid number of terms.");
        return 1;
    }
    printf("Fibonacci sequence: ");
    for (i = 0; i < n; ++i) {
        printf("%llu", fib(i));
        if (i + 1 < n) printf(" ");
    }
    putchar('\n');
    return 0;
}
```

Sample execution

```
Number of terms (1 to 40): 10
Fibonacci sequence: 0 1 1 2 3 5 8 13 21 34
```