**Institute of Computer Technology**

**B. Tech. Computer Science and Engineering**

**Sub: DS**

**Course Code: 2CSE302**

**Practical – 7**

**Name: Jaymin Gondaliya**

**Enrollment No: 23162171007**

**Sem - 3**

**Branch: CS**

**Class: A**

**Batch: 32**

**Problem Definition-1:** Tower of Hanoi is a mathematical puzzle where we have three rods and n disks. The objective of the puzzle is to move the entire stack to another rod, obeying the following simple rules:

1. Only one disk can be moved at a time.

2. Each move consists of taking the upper disk from one of the stacks and placing it on top of another stack i.e. a disk can only be moved if it is the uppermost disk on a stack.

3. No disk may be placed on top of a smaller disk. Design a program for Tower of Hanoi using recursion.

**Code:**

#include <stdio.h>

*// Function to solve the Tower of Hanoi problem*

void tower\_of\_hanoi(int n, char source, char destination, char auxiliary) {

*// Base case: If there is only one disk, move it from source to destination*

    if (n == 1) {

        printf("Top Disk moved from %c to %c\n", source, destination);

        return;

    }

*// Move n-1 disks from source to auxiliary using destination as the auxiliary rod*

    tower\_of\_hanoi(n - 1, source, auxiliary, destination);

*// Move the top disk from source to destination*

    printf("Top Disk moved from %c to %c\n", source, destination);

*// Move the n-1 disks from auxiliary to destination using source as the auxiliary rod*

    tower\_of\_hanoi(n - 1, auxiliary, destination, source);

}

int main() {

    int n;

    printf("Enter the number of disks: ");

    scanf("%d", &n);

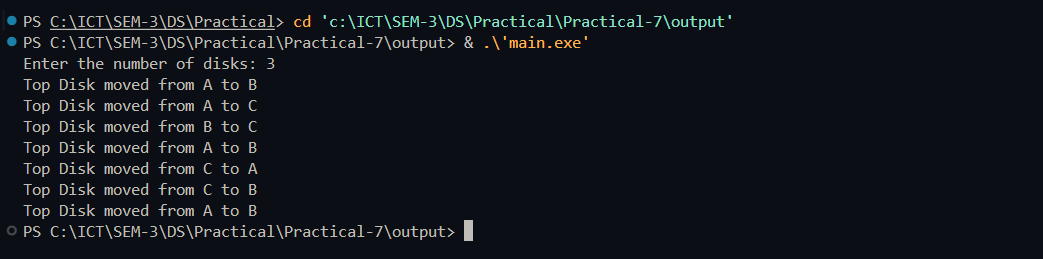
*// Rods are represented as 'A', 'B', 'C'*

    tower\_of\_hanoi(n, 'A', 'B', 'C');

    return 0;

}

**Output:**

****