**Data Structures and Algorithms**

**Lab-08**

**Name:** Ahmad Amjad Mughal

**Reg No:** 121672

**Class:** BSCS-6C

Task:

//Name: Ahmad Amjad Mughal

//Class: BSCS-6C

//Reg No: 121672

#include <iostream> //input/output library

#include <stack> //import library stack

using namespace std;

//Function that solves Tower of hano Puzzle

void Tower(int disk, stack <int> start, stack <int> destination, stack <int> checkpoint) {

//

if (disk == 1)

{ //code for base case

cout << "Moving Disk # 1 " << &start << " --> " << &destination << endl;

}

else //for disks more than 1 then else statement executes and fucntion call itself

{

Tower(disk - 1, start, checkpoint, destination); //Recursion process

cout << "Moving Disk # " << disk << " " << &start << "-->" << &destination << endl;

Tower(disk - 1, checkpoint, destination, start);

}

}

int main(void) {

int disk;

//Creating three stack objects that are taken as the three towers

stack <int> start;

stack <int> destination;

stack <int> checkpoint;

cout << "\tTOWERS OF HANOI " << endl; //prompt

cout << "Enter the number of Disks : "; //Asking the user to enter No of disks

cin >> disk;

cout << endl;

//Printing the address of the three stacks

cout << "Tower 1:\t" << &start << endl;

cout << "Tower 2:\t" << &destination << endl;

cout << "Tower 3:\t" << &checkpoint << endl;

cout << endl;

//Pushing the disks onto the source stack

for (int i = disk; i>0; i--) {

start.push(i);

}

Tower(disk, start, destination, checkpoint); //function call that actually solves Tower of Hanoi Problem

return 0;

}

**Screenshot:**

