

Lab 5: Understanding the Recursion with Application

A. Explain algorithm of Fibonacci with its tree diagram for f(5).

Algorithm:

1. Without recursion

Step 1: Start

Step 2: Declare variables i, a, b, show

Step 3: Initialize the variables, $a = 0$, $b = 1$, and $show = 0$

Step 4: Enter the number of terms of Fibonacci series to be printed, as n

Step 5: Print the First two terms of series

Step 6: Use loop for the following steps

$show = a + b$

$a = b$

$b = show$

 Increase value of I each time by 1

 Print the value of show

Step 7: Stop

2. With recursion

Step 1: Start

Step 2: Declare a function Fibo with interger argument num

Step 3: If($num == 0$)

 Then return 0;

Step 4: Else if($num == 1$)

 Then return 1;

Step 5: Else

 Return $fibonacci(num-1) + fibonacci(num-2)$

Step 6: Stop

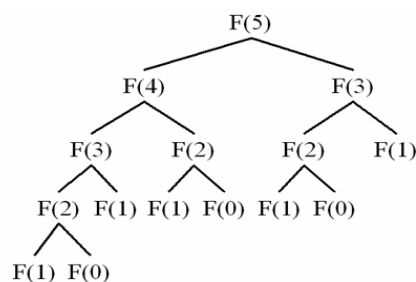


Fig: Tree diagram for f(5)

B. Explain algorithm of TOH with its tree diagram for 3 disks.

Algorithm:

To move n disks from A plate to plate using B plate

Step 1: Start

Step 2: move (n-1)th disks from A to B using C

Step 3: move nth bottom disks from A to C

Step 4: move (n-1)th disks from B to C using A

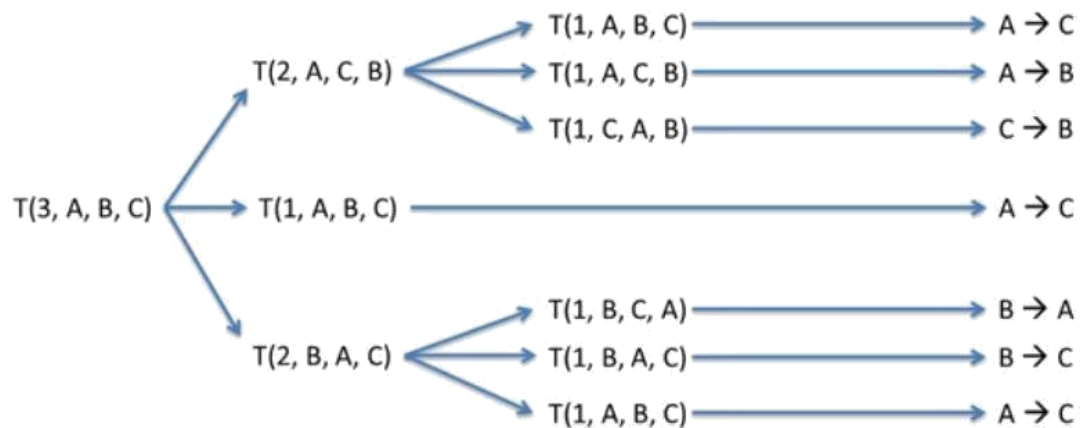


Fig: Recursion tree diagram for 3 disks

Implementation of TOH using Recursion in C++

```
#include<iostream>
using namespace std;

//tower of HANOI function implementation
int TOH(int n,char ini, char mid, char last)
{
    if(n==1)
    {
        cout<<"Move Disk "<<n<<" from "<<ini<<" to "<<last<<endl;
        return 0;
    }

    TOH(n-1,ini,last,mid);
    cout<<"Move Disk "<<n<<" from "<<ini<<" to "<<last<<endl;
    TOH(n-1,mid,ini,last);
    return 0;
}

//main program
int main()
{
    int n;
    cout<<"Enter no. of disks:";
    cin>>n;
    //calling the TOH
    TOH(n,'A','B','C');

    return 0;
}
```

Implementation of Fibonacci using Recursion in C++

```
#include<iostream>
using namespace std;

int fibonacci (int number)
{
    if (number == 0) return 0;
    else if (number==1) return 1;
    else return (fibonacci(number-1)+ fibonacci(number-2));
}

int main(){
    cout << fibonacci(5);
    return 0;
}
```