

## **12.TOWERS OF HANOI USING RECURSION**

### **Aim:**

To write a c program for solving the towers of Hanoi problem using recursion.

### **Algorithm:**

Step 1:Start the program

Step 2:get the no of disks

Step 3:recursively move the no of disks from one peg to another

Step 4:print the steps of each disk movement from one peg to another.

Step 5:Stop the program.

### **Program coding:**

```
/*
```

Rules of Tower of Hanoi:

Only a single disc is allowed to be transferred at a time.

Each transfer or move should consist of taking the upper disk from one of the stacks and then placing it on the top of another stack i.e. only a topmost disk on the stack can be moved.

Larger disk cannot be placed over smaller disk; placing of disk should be in increasing order.

```
*/
```

```
#include <stdio.h>
```

```
#include <conio.h>
```

```
void towerofhanoi(int n, char from, char to, char aux)
```

```
{
```

```
if (n == 1)
```

```
{
```

```
printf("\n Move disk 1 from Peg %c to Peg %c", from, to);
```

```
return;
```

```
}
```

```
towerofhanoi(n-1, from, aux, to);
```

```
printf("\n Move disk %d from Peg %c to Peg %c", n, from, to);
```

```
towerofhanoi(n-1, aux, to, from);
```

```
}
```

```
int main()
```

```
{
```

```
    int n;
```

```
    clrscr();
```

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

```
    scanf("%d",&n); // Number of disks
```

```
    towerofhanoi(n, 'A', 'C', 'B'); // A, B and C are names of peg
```

```
    getch();
```

```
    return 0;
```

```
}
```

## **Output:**

```
Enter the number of disks : 3
```

```
Move disk 1 from Peg A to Peg C
```

```
Move disk 2 from Peg A to Peg B
```

```
Move disk 1 from Peg C to Peg B
```

```
Move disk 3 from Peg A to Peg C
```

```
Move disk 1 from Peg B to Peg A
```

```
Move disk 2 from Peg B to Peg C
```

```
Move disk 1 from Peg A to Peg C_
```