TOWER OF HANOI USING C PROGRAMMING

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.

// objective of the puzzle is to move the stack of disks from rod A to rod C using auxiallary rod B

#include<stdio.h>

*void* towerOfHanoi(*int* *n*, *char* *from\_rod*, *char* *to\_rod*, *char* *aux\_rod*)

{

    if(*n*==1)

    {

        printf("\n Move 1 disk from rod %c to rod %c", *from\_rod*, *to\_rod*);

        return;

    }

    towerOfHanoi(*n*-1, *from\_rod*, *aux\_rod*, *to\_rod*);

    printf("\n Move disk %d from rod %c to rod %c ", *n*, *from\_rod*, *to\_rod*);

    towerOfHanoi(*n*-1, *aux\_rod*, *to\_rod*, *from\_rod*);

}

*int* main()

{

*int* n = 1; // Number of disks

    towerOfHanoi(n, 'A', 'C', 'B');  // A, B and C are names of rods

    return 0;

}

// For n disks , total 2^n-1 moves are required and 2^(n-1) function calls are made

