Tower of Hanoi Problem

Rahul Das Gupta

The **Tower of Hanoi** (also called the **Tower of Brahma** or **Lucas' Tower** and sometimes pluralized) is a mathematical game or puzzle. It consists of three rods and a number of disks of different sizes, which can slide onto any rod. The puzzle starts with the disks in a neat stack in ascending order of size on one rod, the smallest at the top, thus making a conical shape.

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 or on an empty rod.**
3. **No disk may be placed on top of a smaller disk.**

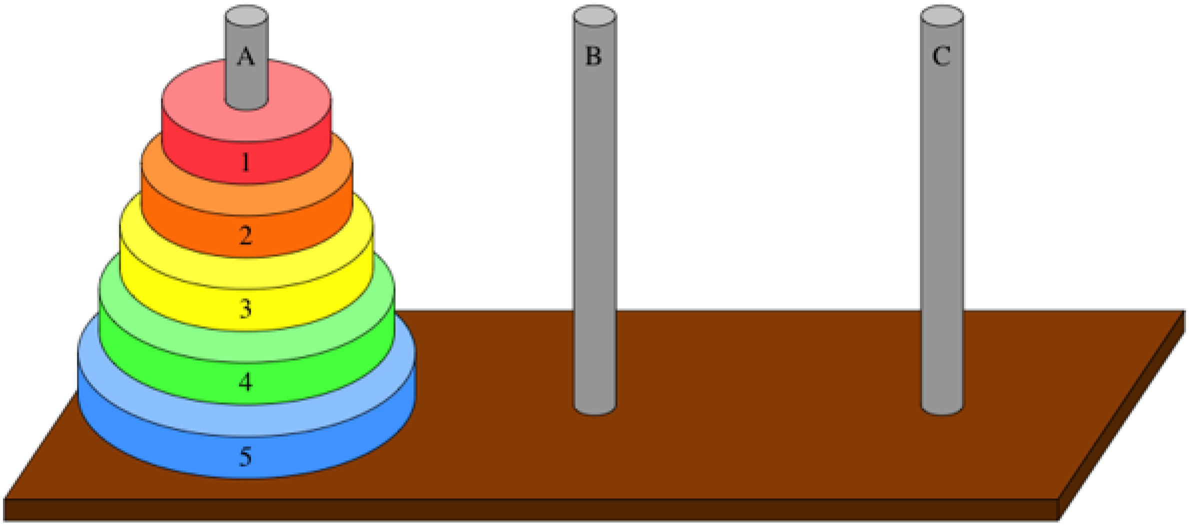

Fig.-1

Approach

Illustrated in the **Fig. 2**.

Step #1: **Recursively move top (N-1) disk** from **source** to **auxiliary** peg.

Step #2: Move the last disk (largest disk at the bottom) from **source** to **destination peg**.

Step #3: **Recursively move (N-1) disk** from **auxiliary** to **destination** peg.

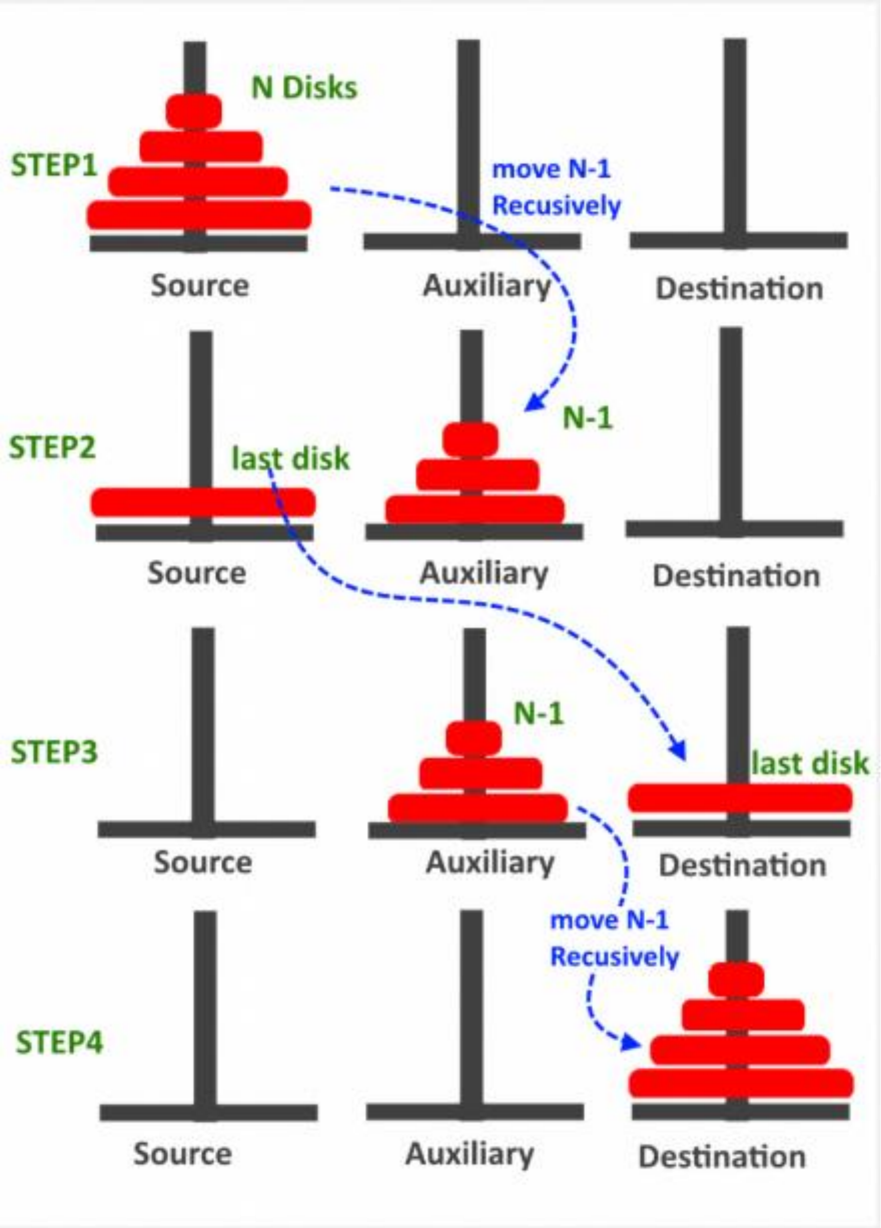

Fig.-2

Terminating Condition for Recursion:

If N=0 then No disk transfer is required.

Recursive C function to solve Tower of Hanoi Problem

```
# include <stdio.h>


viod initialisation (int *);
void Tower_of_Hanoi ( int *,  int , char, char, char);


viod initialisation (int * step_count)
{
    *step_count=0;
}



void Tower_of_Hanoi ( int *step_count,  int N, char Source, char
Aux, char Target)
{
    if(N>0)
    {
       Tower_of_Hanoi (step_count, (N-1), Source, Target, Aux);
       printf("\n Step No. %d : Transfer Disk %d from %c to
                  %c",   ++(*step_count), N, Source, Target);
       Tower_of_Hanoi (step_count, (N-1), Aux, Source, Target);
    }
}
```

```
void main( )
{
    int n, step_no;
    printf("\n Enter the no. of disks:");
    scanf("%d", &n);
    initialisation(&step_no);
   Tower_of_Hanoi ( &step_no,  n, 'S', 'A', 'T');
 }
```

No. of Steps Required to Transfer N no. Disks

Let, T(N) = No. of steps required to transfer N disks

$$
\begin{aligned}
&T(\mathrm{N}) = 2T(N-1)+1 \quad \textit{if } N > 0 \\
&T(0) = 0 \\
&T(\mathrm{N}) = 2T(N-1)+1 \\
&\qquad = 2[2T(N-2)+1]+1 \\
&\qquad = 2^2 T(\mathrm{N}-2)+(2+1) \\
&\qquad = 2^2 [2T(\mathrm{N}-3)+1]+(2+1) \\
&\qquad = 2^3 T(\mathrm{N}-3)+(2^2+2+1) \\
&\qquad\quad \ldots. \qquad \ldots. \qquad \ldots. \\
&\qquad = 2^r T(\mathrm{N}-r)+(2^{r-1}+\ldots+2^2+2+1) \\
&\qquad\qquad\qquad (\text{after r no. of steps}) \\
&\qquad = 2^N T(0)+(2^{N-1}+\ldots+2^2+2+1) \\
&\qquad\qquad\qquad (\text{after } N \text{ no. of steps}) \\
&\qquad = 0+(2^{N-1}+\ldots+2^2+2+1) \quad [\because T(0)=0] \\
&\qquad = 2^N - 1
\end{aligned}
$$