

Tower of Hanoi

Agenda

- Tower of Hanoi introduction
- Tower of Hanoi for n disks
- Tower of Hanoi Algorithm for 3 disks
- Tower of Hanoi Algorithm for 4 disks
- Tower of Hanoi using recursion
- Tower of Hanoi in Java using Recursion – Codes
- Tower of Hanoi in Python
- Tower of Hanoi using Stack
- Tower of Hanoi in Java using Stack - Codes
- Complexity Analysis of Tower of Hanoi

Tower of Hanoi Introduction

Tower of Hanoi Introduction

What is Tower of Hanoi?

- IT is a mathematical puzzle that involves rods and discs. There are three rods and n discs of different sizes.
- History of Tower of Hanoi

Tower of Hanoi Introduction

Tower of Hanoi Game

- There are three rods and n (let $n=3$) discs of different sizes.
- Puzzles begin with the state that one rod has all the discs stacked one above the other in descending order from below.
- The game is about solving the puzzle such that all the discs are arranged in a different rod in same order.
- The rules to be followed are:-
 1. Only one disc can be moved at a time
 2. Discs should not be arranged at any given time such that a larger disk is stacked over the smaller disk.
 3. The disks should be placed on rods only

Tower of Hanoi for n disks

Tower of Hanoi for n Disks

How many moves are required when there are n disk ?

The tower of Hanoi formula is $2^n - 1$

Tower of Hanoi Algorithm for 3 disks

Tower of Hanoi Algorithm for 3 Disks

Tower of Hanoi solution



Tower of Hanoi Algorithm for 4 disks

Tower of Hanoi Algorithm for 4 Disks

Tower of Hanoi solution



Tower of Hanoi using Recursion

Tower of Hanoi using Recursion

Step 1:-

Shift the top $n-1$ disks to the helper rod (B) **//recursive call**

Step 2:-

Shift the last disk from the initial rod to the final rod

Step 3:-

Shift the $n-1$ disks from the helper disk to the destination rod **//recursive call**

Tower of Hanoi in Java using Recursion - Codes

Tower of Hanoi in Python - Codes

Tower of Hanoi using Stack

Tower of Hanoi using Stack

```
public static void shift(int n, int S, int H, int D)
{
    if (n > 0)
    {
        move(n-1, S, D, H);
        int x = tower[S].pop();
        tower[D].push(x);
        display();
        move(n-1, H, S, D);
    }
}
```

```
public static void stackelements(int n)
{
    for (int d = n; d > 0; d--)
        rods[1].push(d);
    display();
    move(n, 1, 2, 3);
}
```

Complexity Analysis of Tower of Hanoi

Complexity Analysis of Tower of Hanoi

- Solving this equation that represent Tower of Hanoi
$$T(n) = 2 T(n-1) + 1$$
- Time complexity of tower of Hanoi = $O(2^n)$

Summary

We discussed

- Concepts behind Tower of Hanoi
- Tower of Hanoi for 3 and 4 disks
- Implementation of Tower of Hanoi in Java and python using Recursion
- Tower of Hanoi time complexity analysis

Thank You