S. B. JAIN INSTITUTE OF TECHNOLOGY, MANAGEMENT & RESEARCH, NAGPUR

Practical No. 05

Aim: Implement the concept of the Tower of Hanoi with 3 (A, B, and C) towers using 3 and 4 disks to reach the goal state (on the C tower) and show the production rules.

Name of Student: Shrutika Pradeep Bagdi

Roll No.: CS22130

Semester/Year: V/III

Academic Session: 2024-2025

Date of Performance:

Date of Submission:

AIM: Implement the concept of the Tower of Hanoi with 3 (A, B, and C) towers using 3 and 4 disks to reach the goal state (on the C tower) and show the production rules.

OBJECTIVE/EXPECTED LEARNING OUTCOME:

The objectives and expected learning outcome of this practical are:

- To be able to understand the concept of Tower of Hanoi
- To be able to acquire the problem-solving capability.

THEORY:

Tower of Hanoi is a mathematical puzzle where we have three rods (A, B, and C) and N disks. Initially, all the disks are stacked in decreasing value of diameter i.e., the smallest disk is placed on the top and they are on rod A. The objective of the puzzle is to move the entire stack to another rod (here considered C), obeying the following simple rules:

- Only one disk can be moved at a time.
- 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.
- No disk may be placed on top of a smaller disk.

Tower of Hanoi using Recursion:

The idea is to use the helper node to reach the destination using recursion. Below is the pattern for this problem:

- Shift 'N-1' disks from 'A' to 'B', using C.
- Shift last disk from 'A' to 'C'.
- Shift 'N-1' disks from 'B' to 'C', using A.

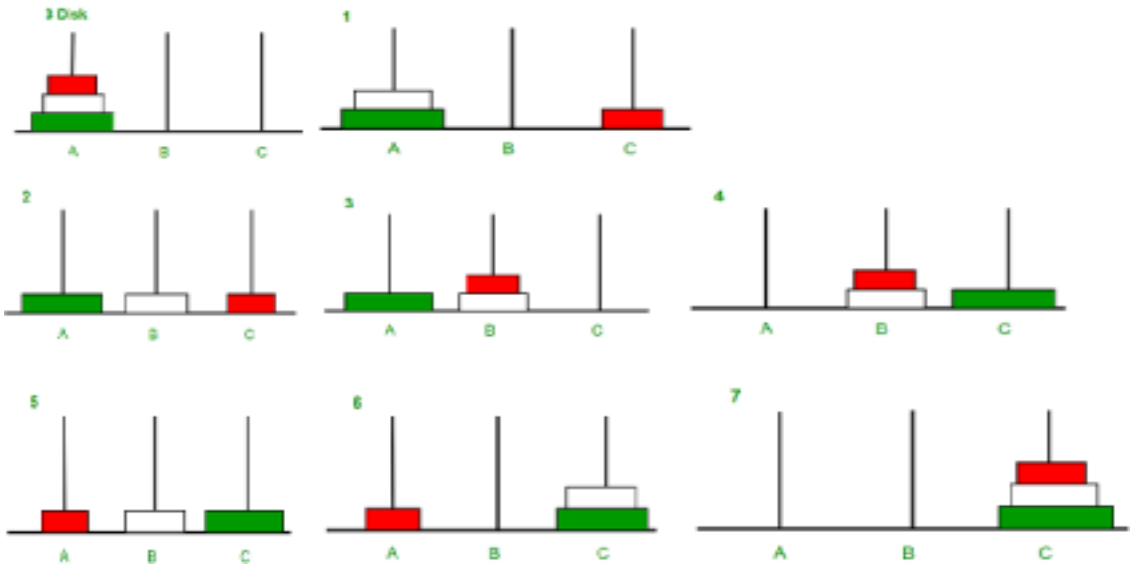

Image illustration for 3 disks

Follow the steps below to solve the problem:

- Create a function **towerOfHanoi** where pass the **N** (current number of disk), **from_rod**, **to_rod**, **aux_rod.**
- Make a function call for N – 1 th disk.
- Then print the current the disk along with **from_rod** and **to_rod**
  · Again make a function call for N – 1 th disk.

TOWER OF HANOI MATHS

Now, you might have identified that to move N disks from one peg to another, you need 2N−1.
So, the number of steps almost double every time you insert another disk in the stack.

Let us prove that the number of steps in 2N−1
The question is what the minimum number of moves (aN) is required to move all the N−disks to another peg.
Let's look at a recursive solution

One can already see that a1=1,a2=3,a3)=7 and so on.
For a given N number of disks, the way to accomplish the task in a minimum number of steps is:

1. Move the top N−1 disks to an intermediate peg.
2. Move the bottom disk to the destination peg.
3. Finally, move the N−1 disks from the intermediate peg to the destination peg.

Therefore, the recurrence relation for this puzzle would become:

$$a_1=1, a_2=3; a_N=2a_{N-1}+1; N\geq 2$$

$$\begin{aligned} a_N &= 2a_{N-1}+1 \\ &= 2(2a_{N-2}+1)+1 \\ &= 2^2 a_{N-2}+2+1 \\ &= 2^3 a_{N-3}+2^2+2+1 \\ &= 2^{N-1} a_{N-(N-1)}+2^{n-2}+\ldots+2^2+2+1 \\ &= 2^N-1 \end{aligned}$$

Now one can solve this relation in an iteration as follows-

Disks	Move
1	1
2	3
3	7
4	15
5	31
6	63
7	127
8	255
9	511
10	1023

APPLICATIONS OF TOWER OF HANOI:

Alzheimer's disease: Balachandar et al., 2015 evaluated the age-related cognitive decline using the Tower of Hanoi task. 24 patients with mild Alzheimer's were assessed for their planning capabilities. The task was carried out using a varying number of disks (2 to 5 disks). The patients showed extreme difficulty in completing the task and showed significantly poorer performance as the disk numbers were increased.

Post-Traumatic Stress Disorder: Yang et al., 2017 assessed the often-neglected neurocognitive functioning in cancer patients with PTSD. The executive function assessment was performed using 3 and 4 disks Tower of Hanoi. In comparison to healthy controls, a significant difference in performance was observed.

Emotional Intelligence: Arefnasab et al., 2012 compared the problem-solving capabilities of individuals with high emotional intelligence with those having low emotional intelligence. The study employed a four disks version of the task without the constraint "no larger disk should be placed on a smaller disk." The subjects were also not given any time constraints. A significant performance difference was found between the two groups, with the high emotional intelligence group making lesser errors and requiring lesser time for the task.

Schizophrenia: Goldberg et al., 1990 used 3 and 4 disks versions of the Tower of Hanoi to assess learning and problem-solving capabilities in schizophrenic patients. In comparison to the control group, schizophrenic patients performed significantly worse. However, performance on the 4 disks Tower of Hanoi task was slightly better than in the 3 disks version in schizophrenic patients suggesting the ability to acquire a procedure and greater preservation of basal ganglia function.

Cerebellar Atrophy: Grafman et al., 1992 evaluated the performance of 12 patients with cerebellar atrophy using the 5 disks Tower of Hanoi task. Their performance was compared to healthy controls. Analysis of the scoring revealed that patients with cerebral atrophy performed significantly worse in

the task. The patients solved fewer trials of the task and showed more tendency in making illegal moves.

STRENGTHS:

- The Tower of Hanoi is a simple mathematical puzzle that can easily assess the problem-solving capabilities of an individual.
- It is a widely used tool for the evaluation of planning and working memory abilities.
- The Tower of Hanoi puzzle is sensitive to prefrontal damage and dysfunction.
- The puzzle's difficulty level can easily be increased or decreased with additional disks or pegs respectively.

LIMITATIONS:

- The Tower of Hanoi cannot be used standalone to understand and assess executive functions of the brain.
- It can be difficult to employ individuals unable to cooperate with the process.
- Other factors such as sleep, anxiety, may also affect performance.

PSEUDO CODE:

START

Procedure Hanoi(disk, source, dest, aux)

IF disk == 1, THEN
move disk from source to dest
ELSE
Hanoi(disk - 1, source, aux, dest) // Step 1
move disk from source to dest // Step 2
Hanoi(disk - 1, aux, dest, source) // Step 3 to
n END IF

END Procedure

STOP

CODE:

```
def hanoi(n, start, helper, end):
    if n == 1:
        print(f"Move disk 1 from {start} to {end}")
        return
    hanoi(n - 1, start, end, helper)
    print(f"Move disk {n} from {start} to {end}")
    hanoi(n - 1, helper, start, end)

def main():
    num_disks = int(input("Enter the number of disks: "))
    print("Sequence of moves:")
    hanoi(num_disks, 'A', 'B', 'C')  # A, B, and C are the names of the pegs

if __name__ == "__main__":
    main()
```

OUTPUT:

CONCLUSION: Thus successfully Implement the code of the Tower of Hanoi.

DISCUSSION QUESTIONS: (write handwritten answers)

1. How you justify Tower of Hanoi used in AI domain?
2. Comment which data structure used for implementation.
3. Which AI technique/ algorithm is similar with Tower of Hanoi puzzle?

REFERENCES:

- https://conductscience.com/digital-health/tower-of hanoi/#:~:text=The%20Tower%20of%20Hanoi%20is,solve%20simple%20real%2Dworld%20problems.
- https://www.hackerearth.com/blog/developers/tower-hanoi-recursion-game-algorithm-explained/ https://www.geeksforgeeks.org/c-program-for-tower-of-hanoi/
- https://futurumcareers.com/solving-the-tower-of-hanoi