Hands-on Activity 4.1: The Tower of Hanoi Problem

Objectives:

Introduce students to the Tower of Hanoi Problem; Apply the fundamentals of logical and algorithmic thinking; Solve the Tower of Hanoi Problem using Python.

Discussion:

The Tower of Hanoi is a mathematical puzzle invented by the French mathematician Edouard LucasLinks to an external site. in 1883.

There are three pegs, source(A), Auxiliary (B) and Destination(C). Peg A contains a set of disks stacked to resemble a tower, with the largest disk at the bottom and the smallest disk at the top. figure 1 Illustrate the initial configuration of the pegs for 3 disks. The objective is to transfer the entire tower of disks in peg A to peg C maintaining the same order of the disks.

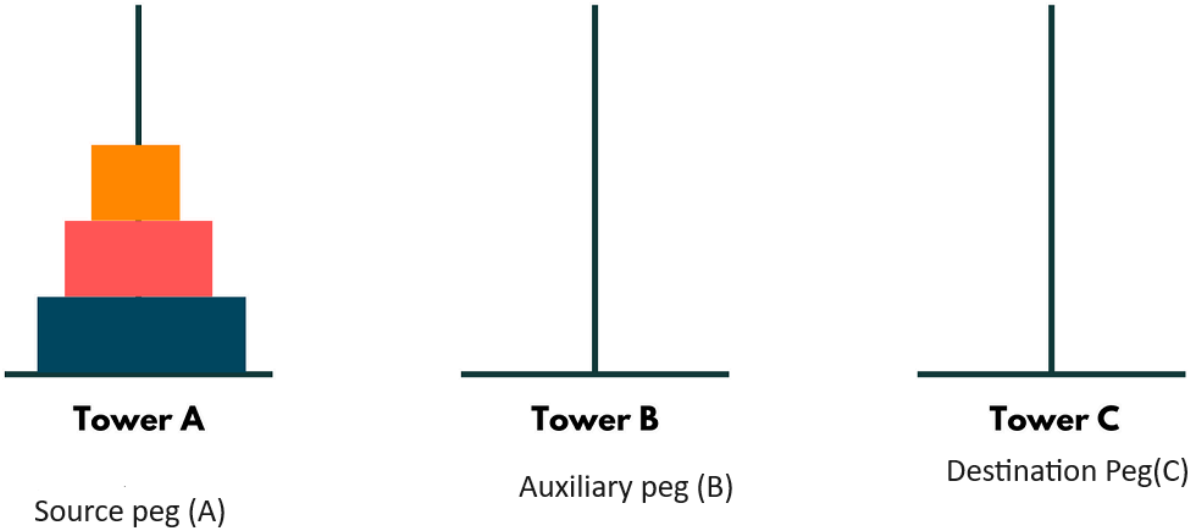

```
def read_moves(n, sp, dp, ap):
    # Base case: if there is only one disk, move it directly from source to destination
    if n == 1:
        print("Move disk 1 from source peg", sp, "to destination peg", dp)
        return
    # Move n-1 disks from source to auxiliary pole using destination pole as temporary
    read_moves(n - 1, sp, ap, dp)
    # Move nth disk from source to destination
    print("Move disk", n, "from source peg", sp, "to destination peg", dp)
    # Move n-1 disks from auxiliary pole to destination using source pole as temporary
    read_moves(n - 1, ap, dp, sp)

# Input: number of disks
number = int(input('Enter the number of disks: '))
# Call the function with appropriate parameters to solve the Tower of Hanoi problem
read_moves(number, 'A', 'B', 'C')
```

```
 Enter the number of disks: 4
 Move disk 1 from source peg A to destination peg C
 Move disk 2 from source peg A to destination peg B
 Move disk 1 from source peg C to destination peg B
 Move disk 3 from source peg A to destination peg C
 Move disk 1 from source peg B to destination peg A
 Move disk 2 from source peg B to destination peg C
 Move disk 1 from source peg A to destination peg C
 Move disk 4 from source peg A to destination peg B
 Move disk 1 from source peg C to destination peg B
 Move disk 2 from source peg C to destination peg A
 Move disk 1 from source peg B to destination peg A
 Move disk 3 from source peg C to destination peg B
 Move disk 1 from source peg A to destination peg C
 Move disk 2 from source peg A to destination peg B
 Move disk 1 from source peg C to destination peg B
```

Supplementary Activity

Explain the programming paradigms/techniques (like recursion or dynamic programming) that you used to solve the given problem. Provide screenshots of the techniques and provide a quick analysis.



This code defines a recursive function read_moves to solve the Tower of Hanoi problem. The function takes four parameters: n (the number of disks), sp (the source peg), dp (the destination peg), and ap (the auxiliary peg).

If there is only one disk, it prints the move to transfer the disk directly from the source peg to the destination peg. Otherwise, it recursively calls itself with n-1 disks, moving them from the source peg to the auxiliary peg, then prints the move for the nth disk from the source peg to the destination peg, and finally recursively calls itself to move the n-1 disks from the auxiliary peg to the destination peg.

At the end, it takes user input to determine the number of disks and calls the read_moves function with the appropriate parameters to solve the Tower of Hanoi problem.