

DAY 19 :

ASSIGNMENT 1:

Task 1: Tower of Hanoi Solver

Create a program that solves the Tower of Hanoi puzzle for n disks.

The solution should use recursion to move disks between three pegs (source, auxiliary, and destination) according to the game's rules.

The program should print out each move required to solve the puzzle.

ANSWER:

```
public class TowerOfHanoi {

    // Method to print the steps
    public static void move(int n, char fromPeg, char toPeg) {
        System.out.println("Move disk " + n + " from " + fromPeg + " to " + toPeg);
    }

    // Recursive method to solve Tower of Hanoi problem
    public static void solve(int n, char source, char auxiliary, char destination) {
        if (n == 0) {
            return;
        }

        // Move n-1 disks from source to auxiliary, so they are out of the way
        solve(n - 1, source, destination, auxiliary);

        // Move the nth disk from source to destination
        move(n, source, destination);

        // Move the n-1 disks from auxiliary to destination
        solve(n - 1, auxiliary, source, destination);
    }
}
```

```
public static void main(String[] args) {  
    int n = 3; // Number of disks  
    solve(n, 'A', 'B', 'C'); // A, B, and C are names of rods  
}  
}
```

Explanation:

1. **move method**: This method prints the action of moving a disk from one peg to another.
2. **solve method**: This is the recursive method to solve the Tower of Hanoi problem. It takes the number of disks n , and the names of the source, auxiliary, and destination pegs.
3. **Base case**: When n is 0, the method returns without making any moves.
4. **Recursive case**:
 - The method first moves $n-1$ disks from the source peg to the auxiliary peg using the destination peg as a temporary holding area.
 - Then, it moves the n th disk from the source peg to the destination peg.
 - Finally, it moves the $n-1$ disks from the auxiliary peg to the destination peg using the source peg as a temporary holding area.
5. **main method**: This method initializes the number of disks and calls the solve method to start the recursive process.

You can change the value of n in the main method to solve the Tower of Hanoi puzzle for a different number of disks.