Tower of Hanoi (Iterative Approach)

■ Problem Statement

The Tower of Hanoi is a mathematical puzzle where: 1. You have 3 rods (A, B, C). 2. You have n discs of different sizes stacked in ascending order on rod A (largest at the bottom, smallest at the top). 3. The objective is to move all the discs from rod A to rod C, following these rules: - Only one disc can be moved at a time. - A disc can only be placed on top of a larger disc (or on an empty rod). - You cannot place a larger disc on a smaller one.

■■ How the Code Works

- The total number of moves required = 2^n - 1. - If the number of discs (n) is odd: - The first move is from A → C. - If the number of discs (n) is even: - The roles of rods B and C are swapped. - Moves are performed in cycles: 1. Move between A and C 2. Move between A and B 3. Move between B and C This cycle repeats until all discs are moved correctly.

■ Example Output

```
Enter number of discs: 3
Move disc from A to C
Move disc from A to B
Move disc from B to C
Move disc from A to C
Move disc from B to A
Move disc from B to C
Move disc from A to C
```

■ How to Run

```
1. Save the code in a file named hanoi.c
2. Compile using:
   gcc hanoi.c -o hanoi -lm
   (-lm is required to link the math library for pow() function).
3. Run the program:
   ./hanoi
```

■ Concepts Covered

- Iterative solution for recursion-based problem. - Modulus operator for cyclic movement. - Use of math.h for power calculation. - Conditional logic for even/odd disc count.