**Aim:** To implement Tower of Hanoi Problem in Prolog.

**Theory:**

The Tower of Hanoi (also called the Tower of Brahma or Lucas' Tower, and sometimes pluralized) is a mathematical game or puzzle. It consists of three rods, and a number of disks of different sizes which can slide onto any rod. The puzzle starts with the disks in a neat stack in ascending order of size on one rod, the smallest at the top, thus making a conical shape.

The objective of the puzzle is to move the entire stack to another rod, 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.

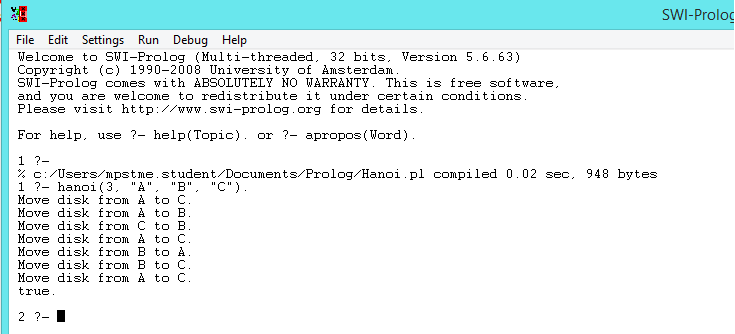
With three disks, the puzzle can be solved in seven moves. The minimum number of moves required to solve a Tower of Hanoi puzzle is 2n - 1, where n is the number of disks.

**Code:**

hanoi(0, \_, \_, \_):- !.

hanoi(N, A, B, C):- M is N-1, hanoi(M, A, C, B), writef("Move disk from %s to %s.\n", [A, C]), hanoi(M, B, A, C).

**Output:**



**Conclusion:**

Hence we have implemented Fibonacci Series problem in prolog.