

## Towers of Hanoi puzzle

### A legend of uncertain origin

- $n = 64$  discs of differing size; 3 posts; discs on one of the posts from largest to smallest.
- An ancient prophecy has commanded monks to move the discs to another post.
- When the task is completed, *the world will end*.

### Rules

- Move discs one at a time.
- Never put a larger disc on a smaller disc.

Q. Generate list of instruction for monks ?

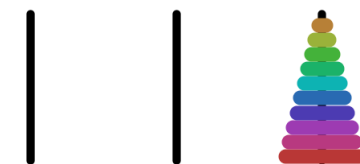
Q. When might the world end ?

$n = 10$

before



after



## Towers of Hanoi

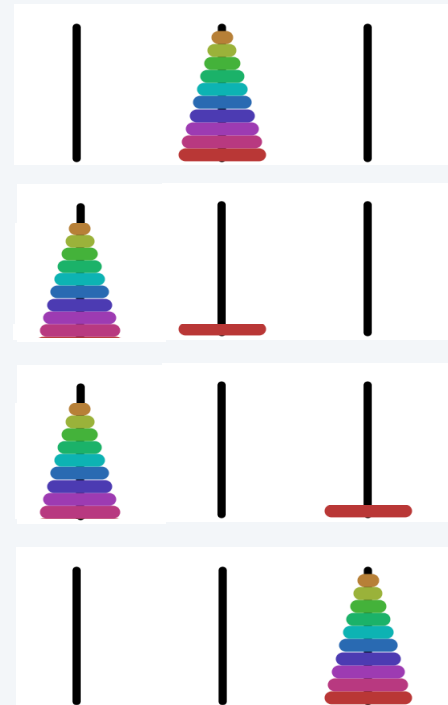
For simple instructions, use cyclic wraparound

- Move *right* means 1 to 2, 2 to 3, or 3 to 1.
- Move *left* means 1 to 3, 3 to 2, or 2 to 1.

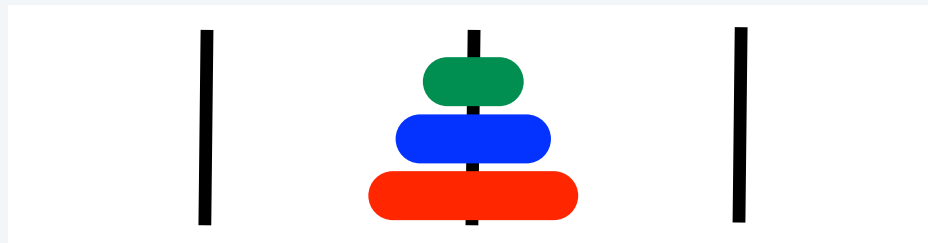


A recursive solution

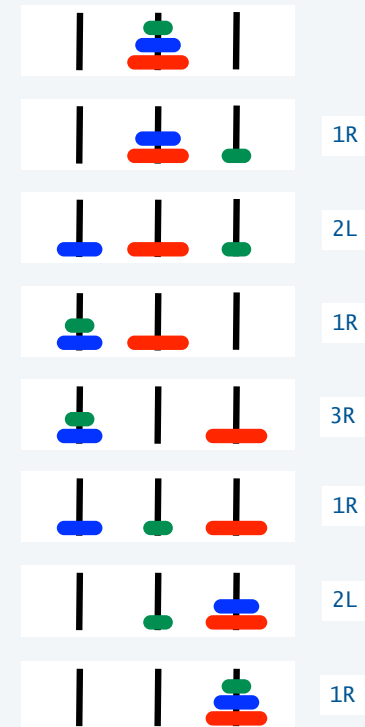
- Move  $n - 1$  discs to the left (recursively).
- Move largest disc to the *right*.
- Move  $n - 1$  discs to the left (recursively).



## Towers of Hanoi solution (n = 3)



1R 2L 1R 3R 1R 2L 1R



## Towers of Hanoi: recursive solution

---

hanoi( $n$ ): Print moves for  $n$  discs.

- Return one space for  $n = 0$ .
- Otherwise, set `move` to the specified move for disc  $n$ .
- Then sandwich `move` between two copies of `hanoi( $n-1$ )`.

```
public class HanoiR
{
    public static String hanoi(int n, boolean left)
    {
        if (n == 0) return " ";
        String move;
        if (left) move = n + "L";
        else     move = n + "R";
        return hanoi(n-1, !left) + move + hanoi(n-1, !left);
    }
    public static void main(String[] args)
    {
        int n = Integer.parseInt(args[0]);
        StdOut.println(hanoi(n, false));
    }
}
```

```
% java HanoiR 3
1R 2L 1R 3R 1R 2L 1R
```