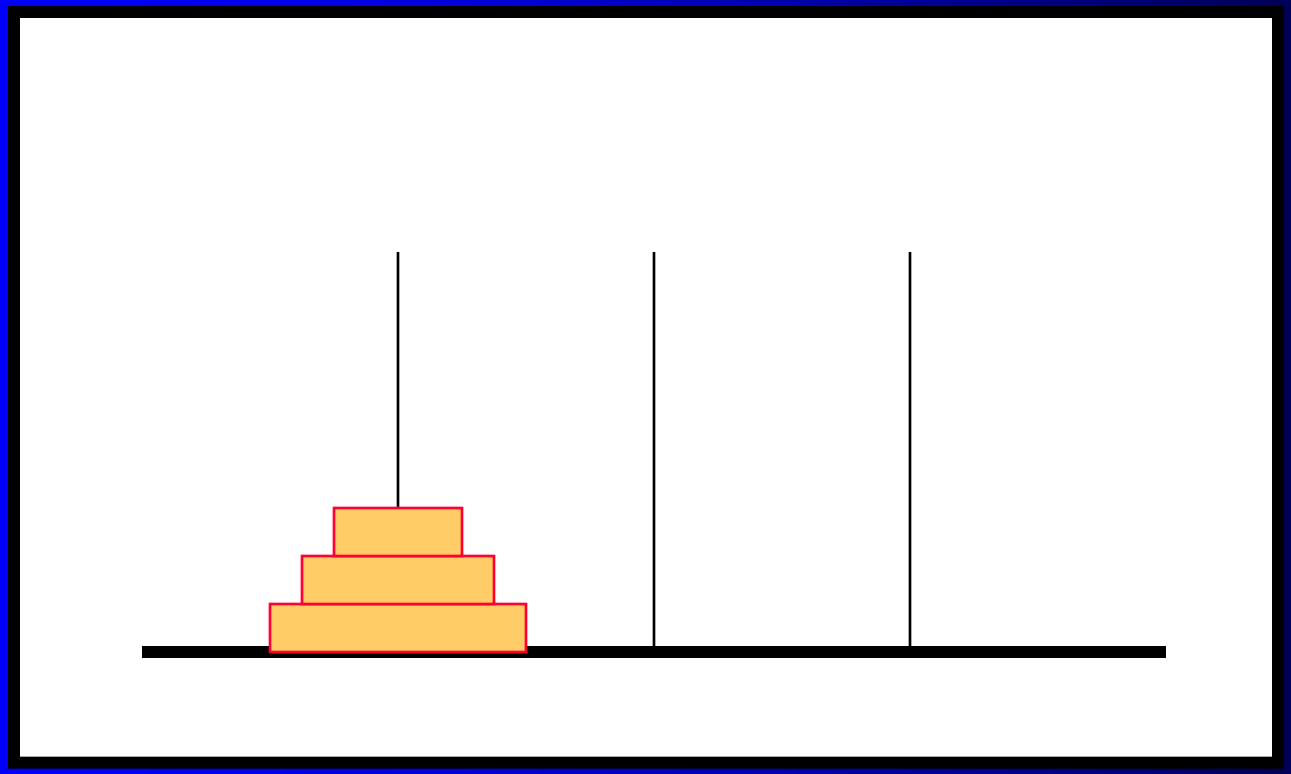
Recursion

- Reading: Savitch, Chapter 11

Recursion – Case Study

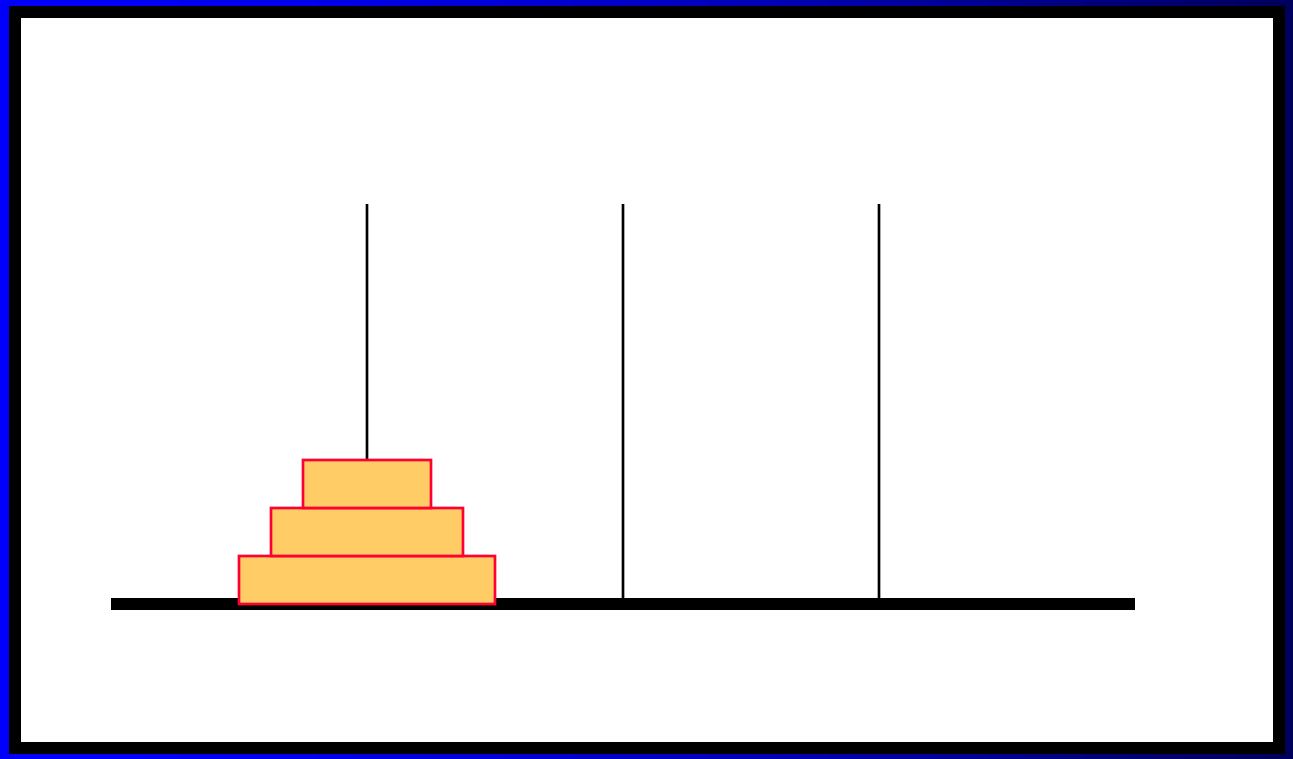
- The Towers of Hanoi



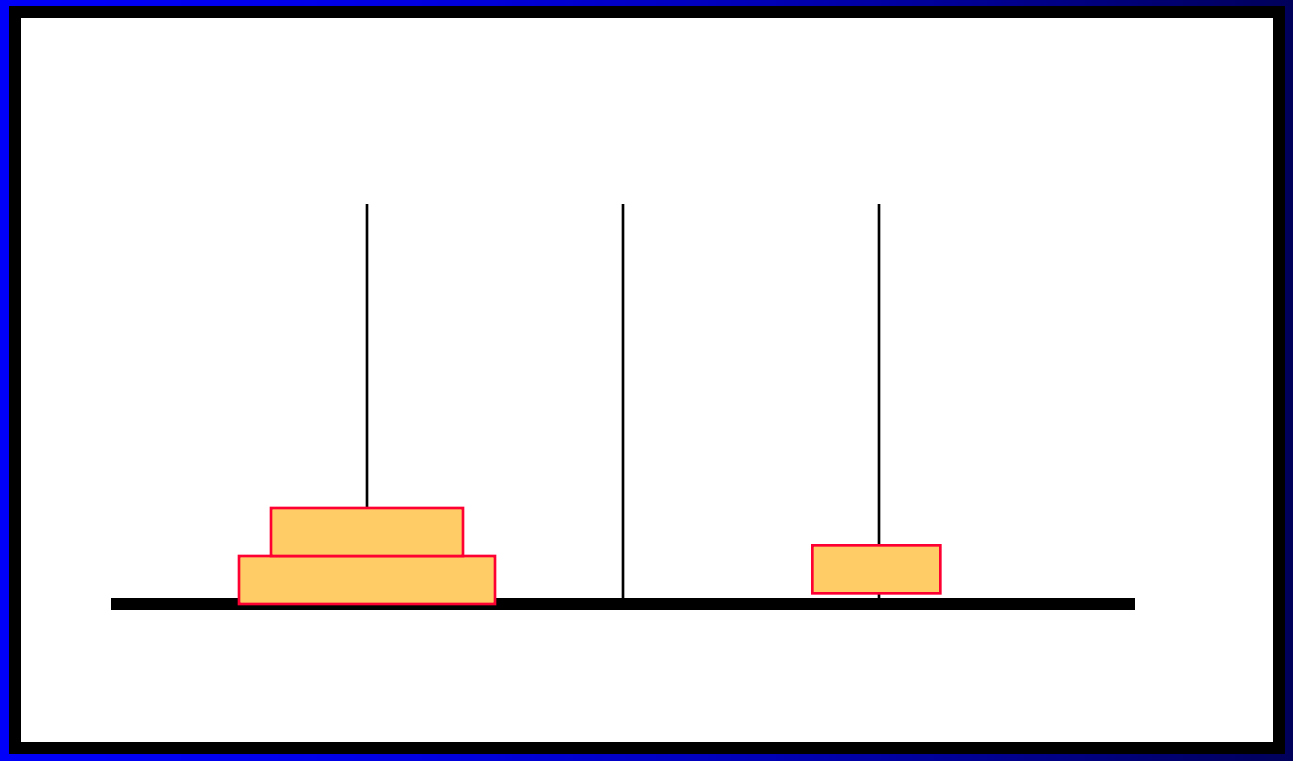
- The goal: to move all of the disks from their original (first) peg to the destination (third) peg.

- The rules:
  - can move one disk at a time.
  - cannot put a larger disk on top of a smaller disk.

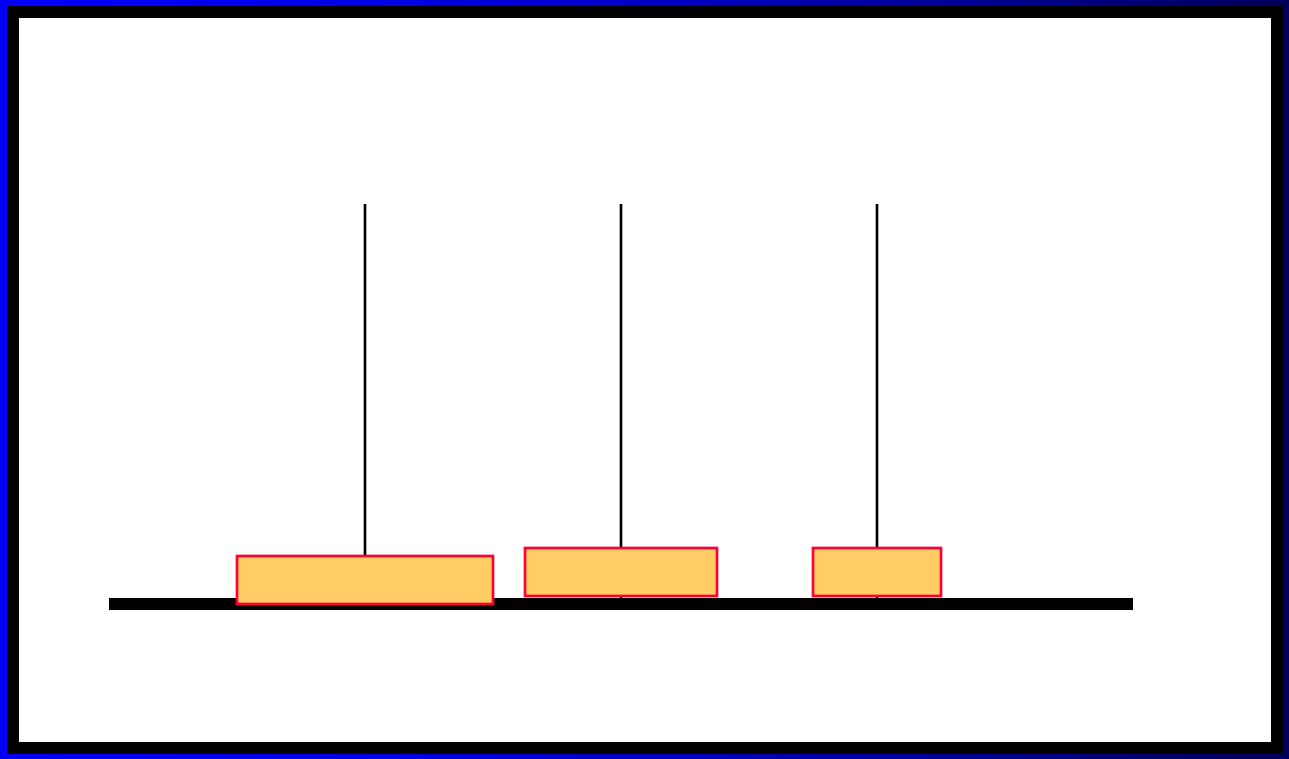
The Towers of Hanoi



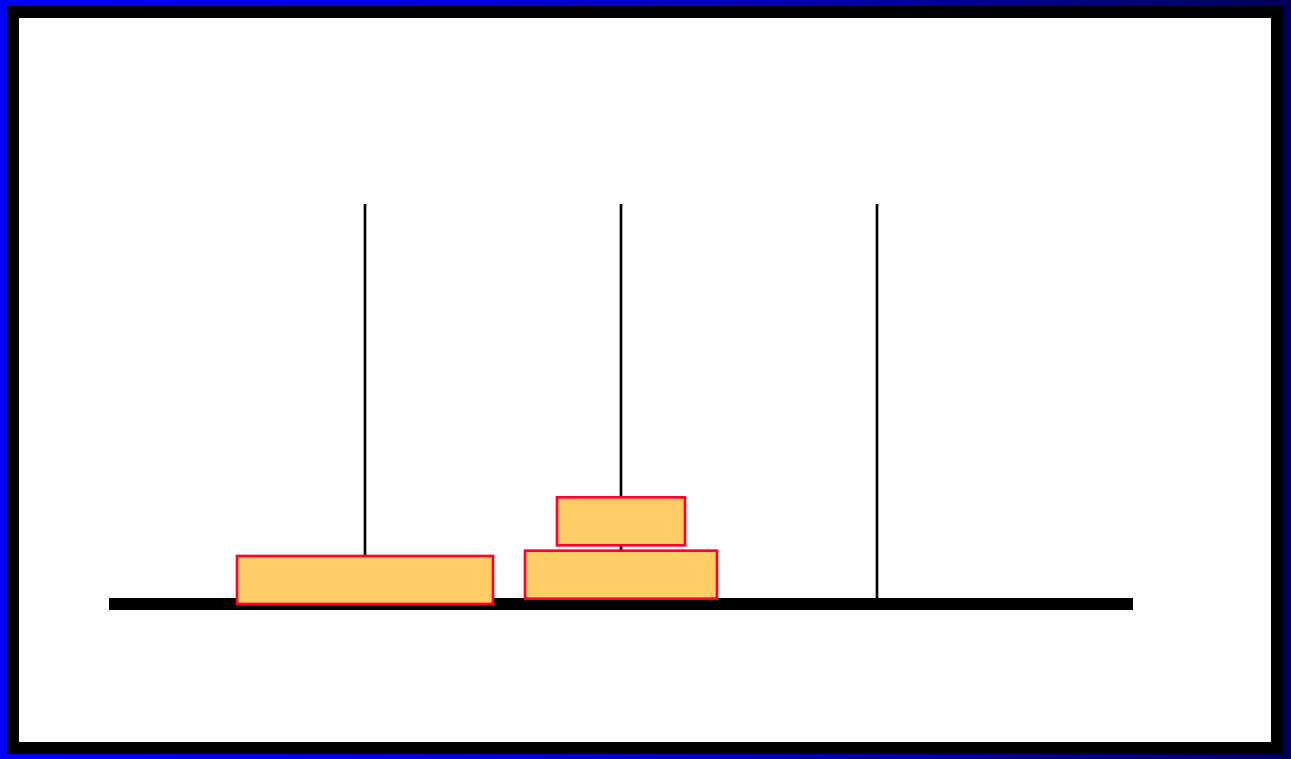
The Towers of Hanoi



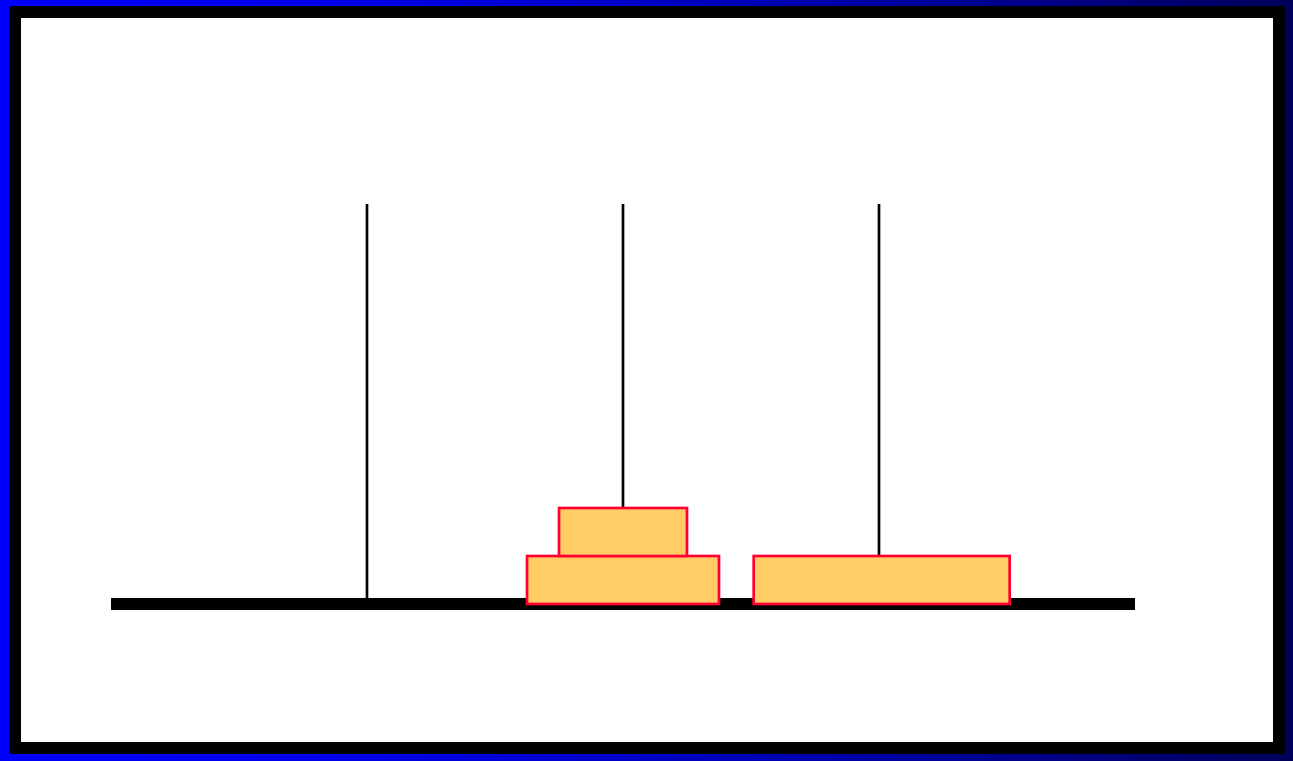
The Towers of Hanoi



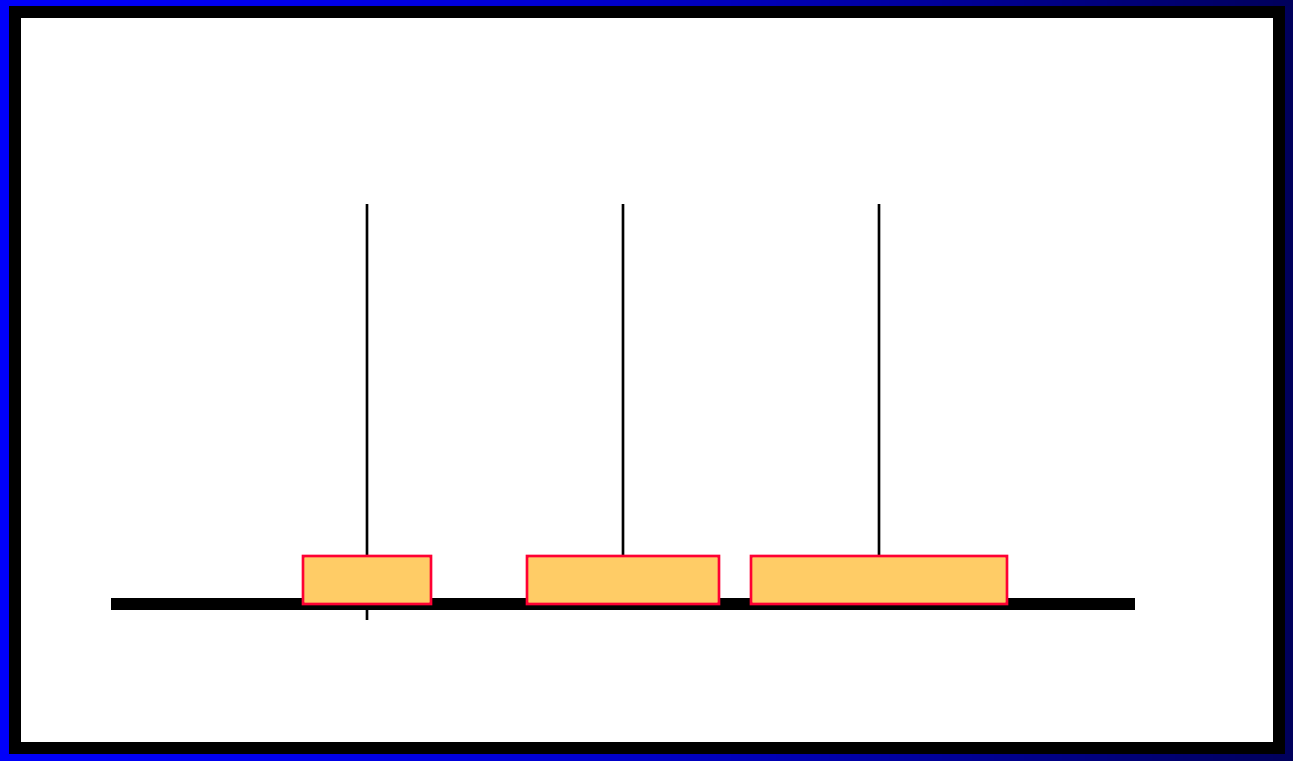
The Towers of Hanoi



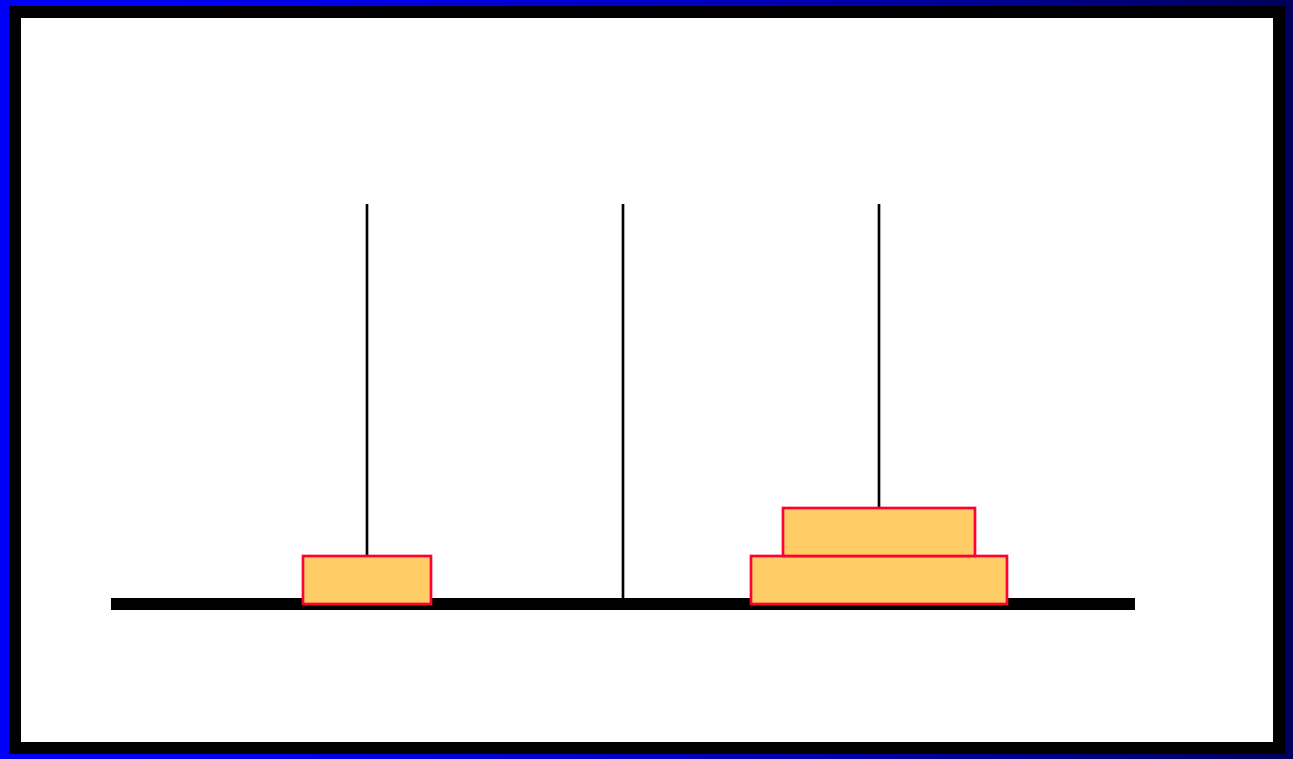
The Towers of Hanoi



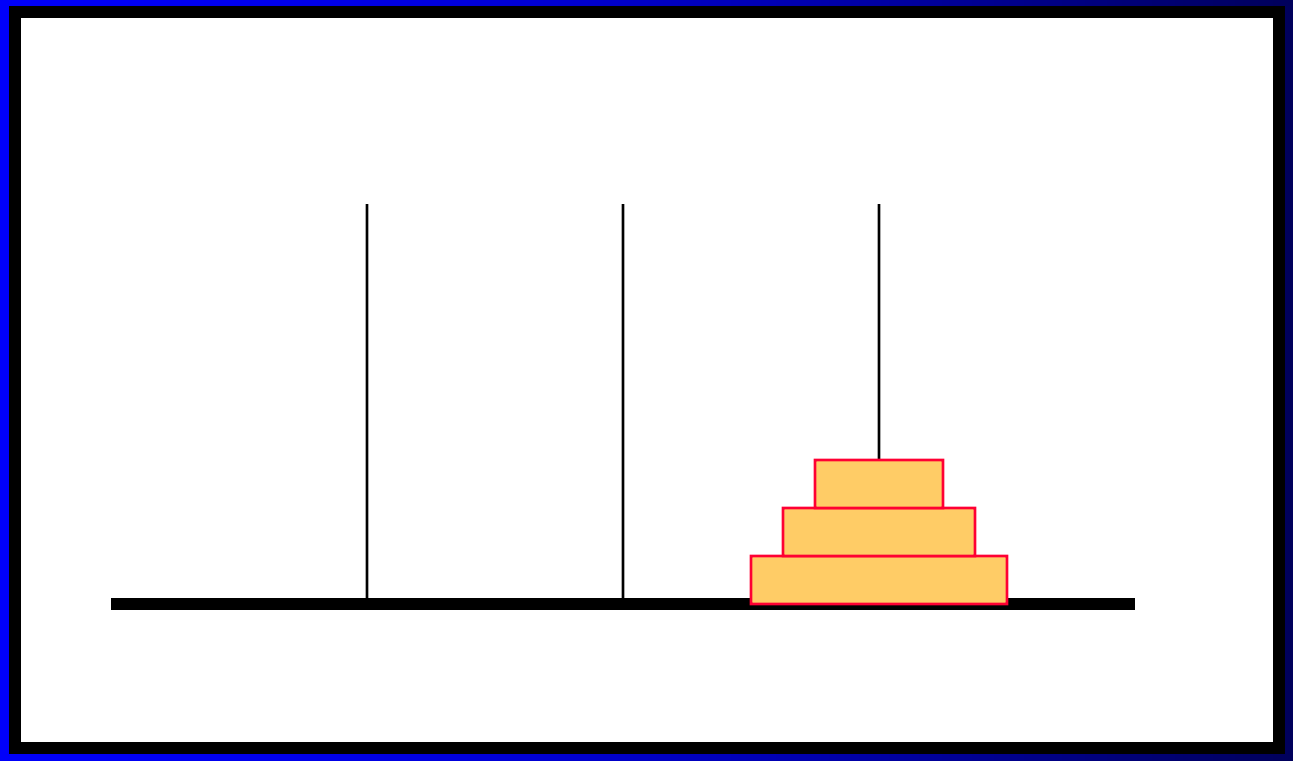
The Towers of Hanoi



The Towers of Hanoi



The Towers of Hanoi



```
//  TowersOfHanoi.java       By Lewis/Loftus

public class TowersOfHanoi {
   private int totalDisks;

    //Sets up the puzzle with the specified number of disks.
    public TowersOfHanoi (int disks) {
      totalDisks = disks;
    }
```

```
//Performs the initial call to moveTower to solve the
//puzzle.
//Moves the disks from tower 1 to tower 3 using tower 2.

public void solve () {
  moveTower (totalDisks, 1, 3, 2);
}
```

```
//Moves the specified number of disks from one tower to
//another by moving a subtower of n-1 disks out of the
//way, moving one disk, then moving the subtower back.
//Base case of 1 disk.
private void moveTower (int numDisks, int start, int end,
int temp) {
  if (numDisks == 1)
     moveOneDisk (start, end);
  else {
     moveTower (numDisks-1, start, temp, end);
     moveOneDisk (start, end);
     moveTower (numDisks-1, temp, end, start);
  }
}
```

```
    //Prints instructions to move one disk from the
    //specified start tower to the specified end tower.

    private void moveOneDisk (int start, int end) {
      System.out.println ("Move one disk from " + start + "
    to " + end);
   }
}
```

```
//  SolveTowers.java       By Lewis/Loftus

public class SolveTowers {

   public static void main (String[] args) {
      TowersOfHanoi towers = new TowersOfHanoi (4);

      towers.solve();
   }
}
```

- Program execution

% java SolveTower

Move one disk from 1 to 2
Move one disk from 1 to 3
Move one disk from 2 to 3
Move one disk from 1 to 2
Move one disk from 3 to 1
Move one disk from 3 to 2
Move one disk from 1 to 2
Move one disk from 1 to 3
Move one disk from 2 to 3
Move one disk from 2 to 1
Move one disk from 3 to 1
Move one disk from 2 to 3
Move one disk from 1 to 2
Move one disk from 1 to 3
Move one disk from 2 to 3