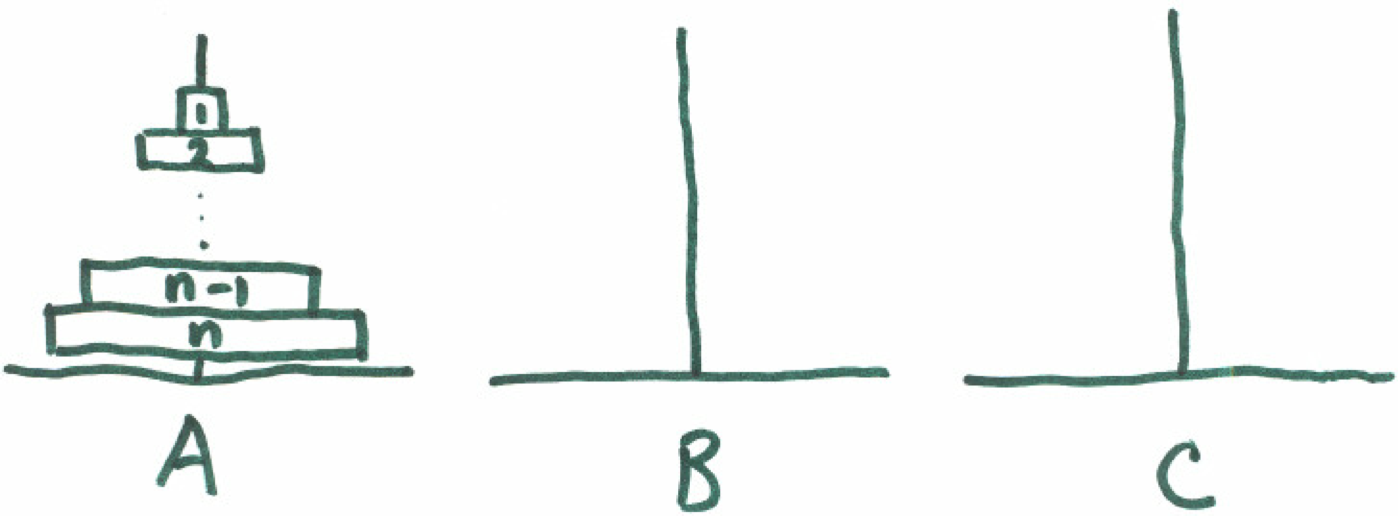
Induction

- Major part of our course
- One programming technique is Recursion.
- The corresponding proof technique is called Induction.
- If you can think recursively, you already know how to think inductively.

Classic Example: Tower of Hanoi

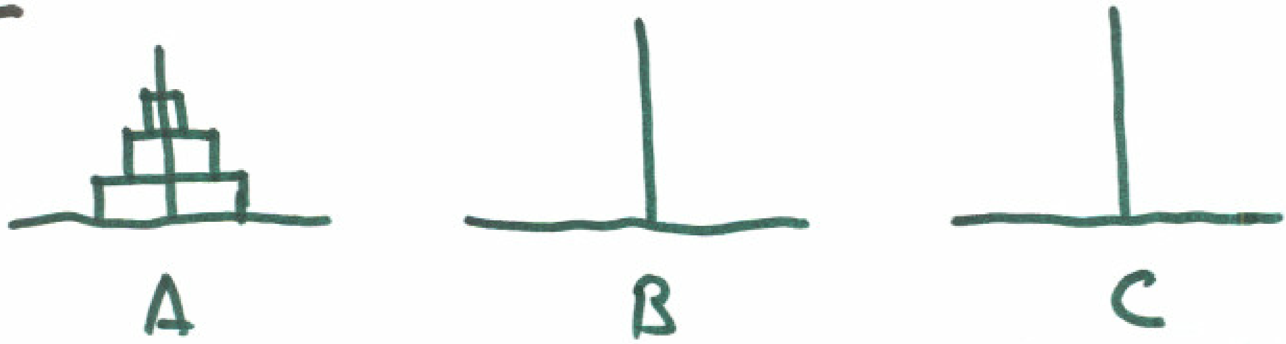
Problem: There are n disks, ~~a~~ stacked on top of each other, from largest (disk n) ~~two~~ on the bottom, to smallest (disk 1) on the top.



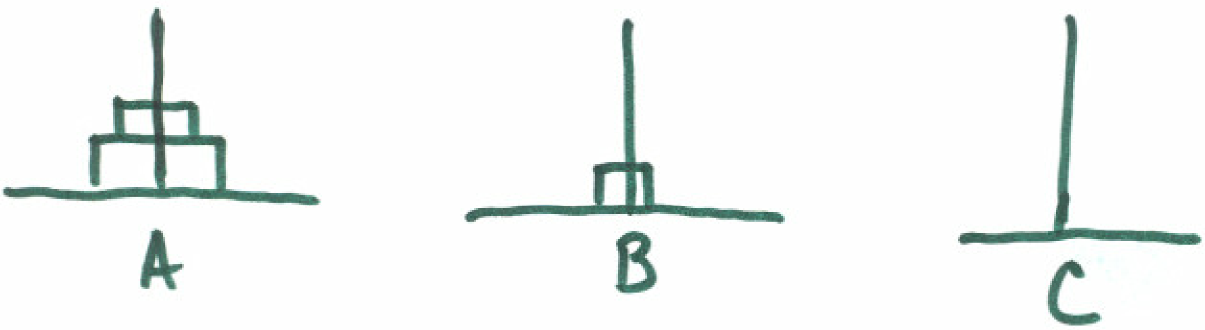
Unfortunately, the disks are at the wrong location. We must move the disks from peg A to peg B. We have a temporary staging area (C) available to us.

At no time can a larger disk rest on a smaller one

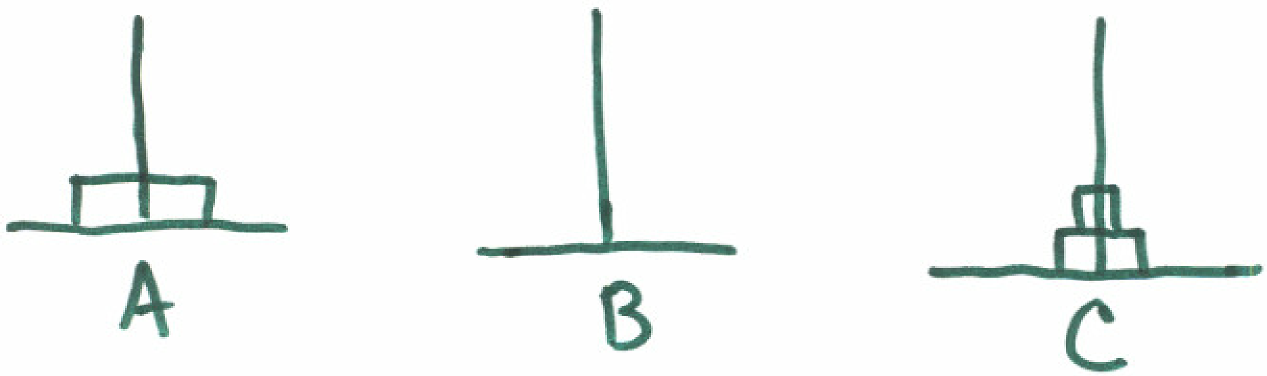
Ex $n=3$



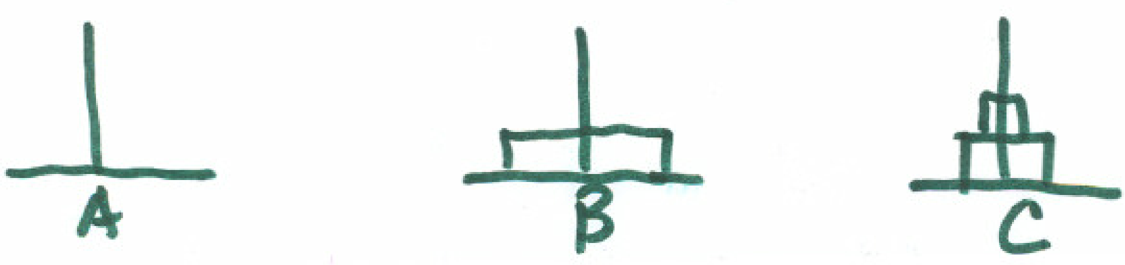
$\Longrightarrow$ move(A,B) $\Longrightarrow$



$\Longrightarrow$ move(A,C) and move(B,C) $\Longrightarrow$



$\Longrightarrow$ move(A,B) $\Longrightarrow$



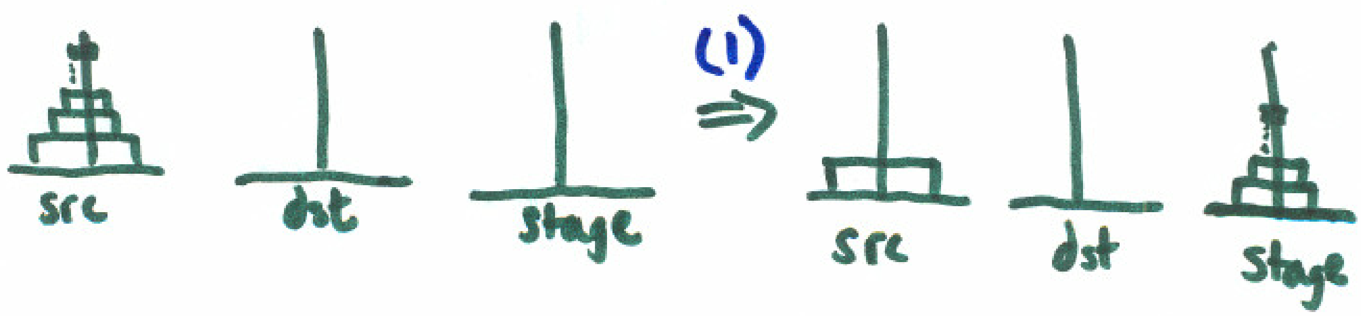
But wait! We already moved the lightest two disks from A to C in the first 3 moves. Now, we want to move the lightest two disks from C to B.

This is a problem we've solved before.

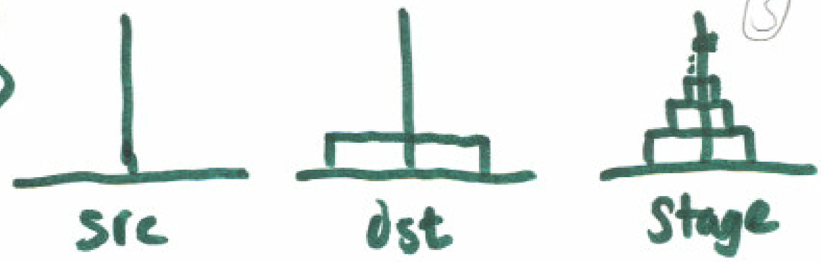
Recursive Solution

```
MoveTower(n, src, dst, stage):

        if n=0 return;

(1)     MoveTower(n-1, src, stage, dst);
(2)     Move(src, dst);
(3)     MoveTower(n-1, stage, dst, src);
        return;
```

Ex:



(2) $\Rightarrow$



(3) $\Rightarrow$

Inductive Proof

Thm For all $n \geq 0$, it is possible to move a tower of n disks between two locations when a third staging location is available.

Pf: By induction on n. If $n = 0$, the statement is clearly true.

Suppose $n \geq 1$. By the inductive hypothesis, we can move the lightest $n-1$ disks from the source location to the staging area, using our destination as the staging area. Because ~~the~~ disk n is the largest, it does not interfere with the moves made inductively.

Next, we can move disk n from our source to our destination. Finally, the inductive hypothesis implies we can move the lightest $n-1$ disks from our staging area to our destination, using our source as the staging area. (Again, because disk n is largest, it does not interfere with the moves made inductively.) ∎

Remark: The inductive proof carries more information than the recursive algorithm because it must explain why the procedure it uses is correct — i.e. why disk n does not interfere with the movement of the $n-1$ smaller disks.

Remark: The theorem and recursive alg. asserts that we can move the tower between any two locations we like. This is called strengthening the inductive hypothesis and is very useful.

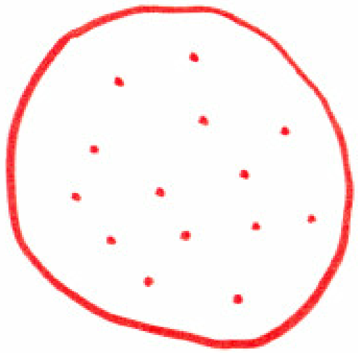
Recursion and Induction

Recursion		Induction
Base Case	⟺	Base Case
Recursive call	⟺	Invoking the **inductive hypothesis**

- When writing a recursive function, making a recursive call to an input that is not smaller gives a function that loops forever.

- When writing an inductive proof, invoking the inductive hypothesis on an input that is not smaller is illegal and results in an incorrect proof.

How does induction work?



S

- You want to prove something of the form

$$\forall x \in S \quad P(x)$$

where $P(x)$ is a statement about x.

- Define a function $\mathrm{size}: S \to \{1,2,\dots\}$.

- Prove the base case:

$$\mathrm{size}(x) = 1 \implies P(x)$$

- Prove the inductive step:

inductive hypothesis

$$\forall x \in S \quad \boxed{(\forall y \ \ \mathrm{size}(y) < \mathrm{size}(x) \implies P(y))} \implies P(x)$$

- Done! Conclude $\forall x \in S \ P(x)$.

Why does induction work?

Suppose we have proved:

(1) $\text{Size}(x) = 1 \implies P(x)$

(2) $\forall x \in S \ \left(\forall y \ \text{size}(y) < \text{size}(x) \implies P(y)\right) \implies P(x)$

Why can we conclude $\forall x \ P(x)$?

Suppose for a contradiction that $P(x)$ does not hold for every $x \in S$. Let z be an element chosen from

$$\{x \in S \mid P(x) \text{ does not hold}\}$$

to minimize $\text{size}(z)$.

By our choice of z, we know

$$\forall y \ \text{size}(y) < \text{size}(z) \implies P(y)$$

Therefore, by (2), $P(z)$ holds, a contradiction.

- Every "Proof by Induction" is shorthand for a Proof by Contradiction:

  Thm $\forall x \in S \; P(x)$

  Pf Suppose for a contradiction that x is a minimal counterexample to $P(x)$....

- Sometimes, proving there is no minimal counterexample is more natural and direct than a proof by induction.

- Often, $\text{size} : S \to \{1, 2, \ldots\}$ is not explicitly defined.

- The simpler the base case, the better.

Thm For all $n \geq 1$, $\sum_{k=1}^{n} (2k-1) = n^2$.

Pf:

- $S = \{1, 2, 3, \ldots\}$
- $P(n) =$ "$\sum_{k=1}^{n} (2k-1) = n^2$"
- $\text{size}(n) = n$

defn of size: By induction on n.

Prove the base case: If $n = 1$, then $\sum_{k=1}^{n} (2k-1) = 1 = n^2$, so the theorem holds.

Prove the inductive step:

If $n \geq 2$, then $\sum_{k=1}^{n} (2k-1) = (2n-1) + \sum_{k=1}^{n-1} (2k-1)$.

By the inductive hypothesis, $P(n-1)$ is true, so $\sum_{k=1}^{n-1} (2k-1) = (n-1)^2$.

Therefore
$$\begin{aligned}\sum_{k=1}^{n} (2k-1) &= (2n-1) + (n-1)^2 \\ &= (2n-1) + n^2 - 2n + 1 \\ &= n^2\end{aligned}$$
and so $P(n)$ is true. ∎

Ex: $n=4$ $\quad \sum_{k=1}^{4}(2k-1) = \underset{k=1}{1} + \underset{k=2}{3} + 5 + 7 = 16 = 4 \cdot 4$

$16 = 1 + 3 + 5 + 7$