Recursion is a common strategy for solving CS problems.

- take a problem instance
- split it into subproblems
- until they are small

ex binary search

Problem: find an element in a sorted array

$$A = \langle \underbrace{a_1, a_2, a_3,} \; \underbrace{\dots \quad , a_n} \rangle$$

$$\begin{array}{ccccccc} & & A_l & & & A_r & \\ & \swarrow & & \searrow & \swarrow & & \searrow \\ A_{l_l} & & & A_{lr} \quad A_{r_l} & & & A_{r_r} \\ & \vdots & & & & \vdots & \end{array}$$

base case: only one element.

Mathematical induction is a proof technique that is analogous to recursion.

ex to prove that $1+2+3+\cdots+n = \frac{n(n+1)}{2}$,

we prove that the formula holds for $n=0$ (base case) and that if it holds for some $n \geq 1$, then it holds for $n+1$.

Def Let P be a predicate concerning ints ≥ 0. To give a proof by mathematical induction that $\forall n \in \mathbb{Z}^{\geq 0} : P(n)$, we prove 2 things:

(1) Base case: prove $P(0)$.

(2) Inductive case: $\forall n \geq 1$, prove

$$P(n-1) \Rightarrow P(n)$$

If we do (1) and (2), we've proved $\forall n \in \mathbb{Z}^{\geq 0} : P(n)$. why?

ex suppose we have proven $P(0)$ and $P(n-1) \Rightarrow P(n)$. these establish $P(3)$

Proof WTS $P(3)$.

statement	reasoning
$P(0)$	we proved it (base case)

$P(0) \Rightarrow P(1)$

$P(1)$

$P(1) \Rightarrow P(2)$

$P(2)$

$P(2) \Rightarrow P(3)$

$P(3)$

$n=1$ for $P(n-1) \Rightarrow P(n)$

bc $P(0)$ (modus ponens)

bc $P(n-1) \Rightarrow P(n)$

modus ponens

bc $P(n-1) \Rightarrow P(n)$

(1) Base case: prove $P(0)$.

(2) Inductive case: $\forall n \geq 0$, prove

$$P(n) \Rightarrow P(n+1)$$

claim, $\forall n \geq 0$, $\underbrace{\sum_{i \in \{0,1,\ldots,n\}} 2^i}_{\text{LHS}} = \underbrace{2^{n+1}-1}_{\text{RHS}}$

$$2^0 + 2^1 + 2^2 + \cdots + 2^n = 2^{n+1} - 1$$

ex

n	LHS	RHS
0	$2^0 = 1$	$2^1 - 1 = 2 - 1 = 1$
1	$2^0 + 2^1 = 1 + 2 = 3$	$2^2 - 1 = 4 - 1 = 3$
2	$2^0 + 2^1 + 2^2 = 1 + 2 + 4 = 7$	$2^3 - 1 = 8 - 1 = 7$