

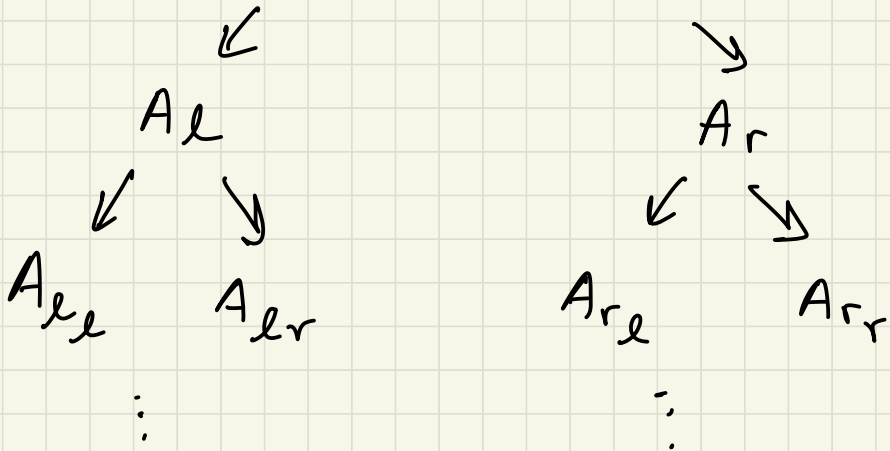
Recursion is a common strategy for solving CS problems.

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

ex binary search

Problem: find an element in a sorted array

$$A = \langle \underbrace{a_1, a_2, a_3, \dots}_{A_l}, \underbrace{\dots, a_n}_{A_r} \rangle$$



base case: only one element.

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

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

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  $\geq 0$ . 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

$P(0)$

reasoning

we proved it (base case)

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

$$P(1)$$

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

$$P(2)$$

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

$$P(3)$$

$$\begin{array}{l} \nearrow n=1 \text{ for } P(n-1) \Rightarrow P(n) \\ \quad bc \ P(0) \text{ (modus ponens)} \end{array}$$

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

modus ponens

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

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

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

$$\text{claim } \forall n \geq 0, \quad \overbrace{\sum_{i \in \{0,1,\dots,n\}} 2^i}^{\text{LHS}} = \overbrace{2^{n+1} - 1}^{\text{RHS}}, \quad P(n) \Rightarrow P(n+1)$$

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

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

$$2^0 + 2^1 + 2^2 = 1 + 2 + 4 = 7$$

$$2^3 - 1 = 8 - 1 = 7$$