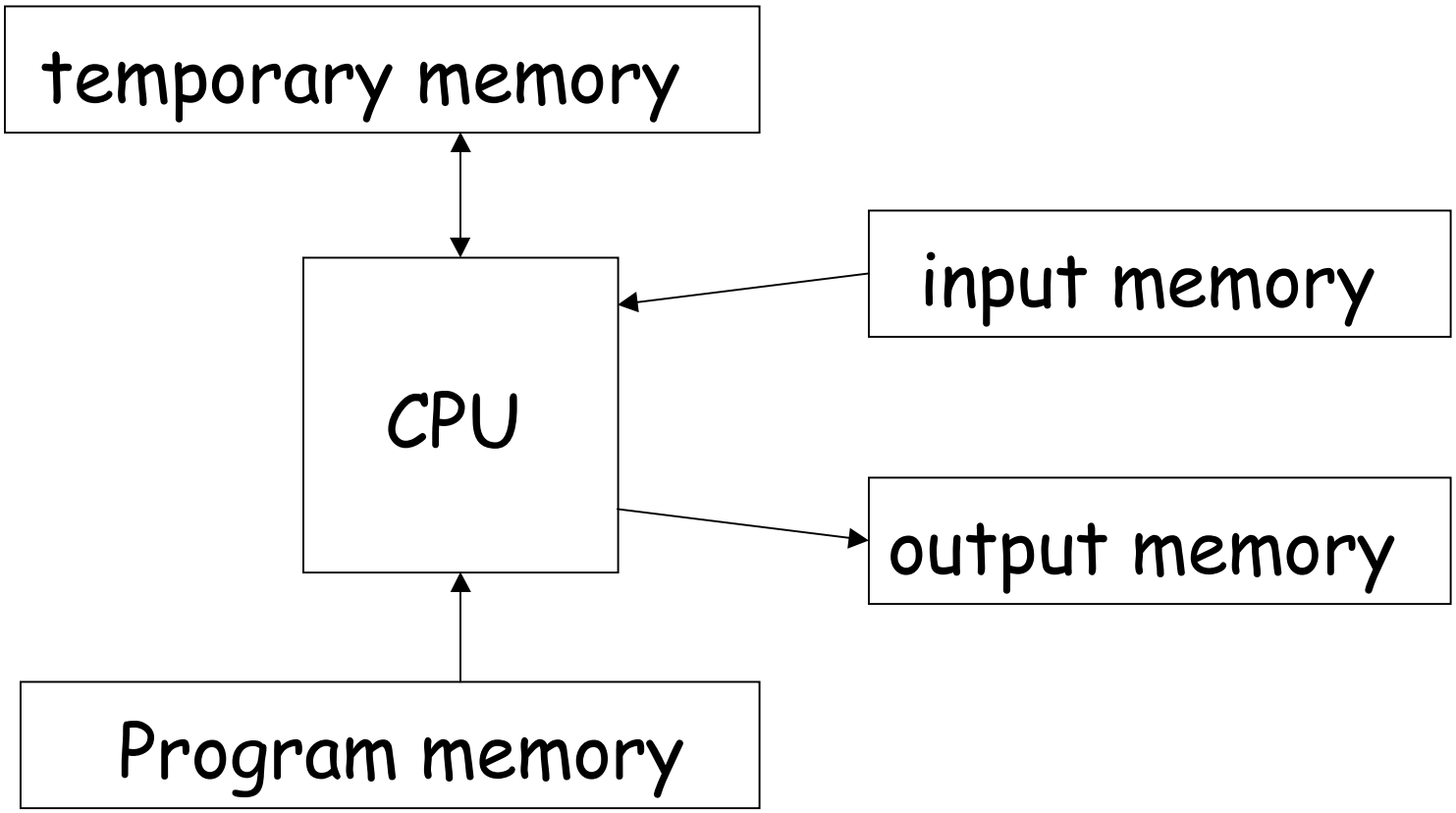
CSCI 2400 section 3

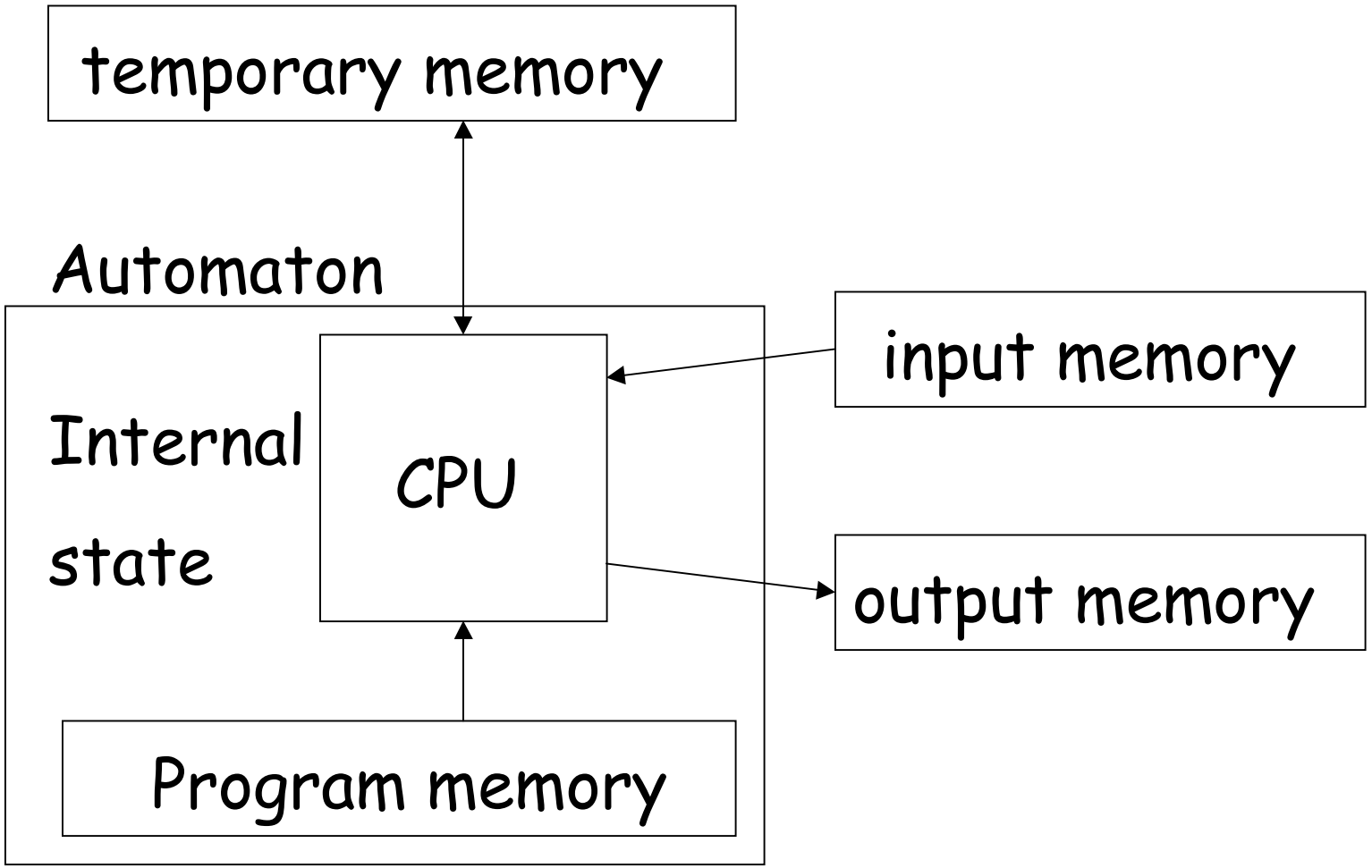
Models of Computation

Instructor: Costas Busch

Computation



A Model Machine



Different Kinds of Automata

Are dinstinguished by the temporary memory

- Finite Automata

  No temporary memory

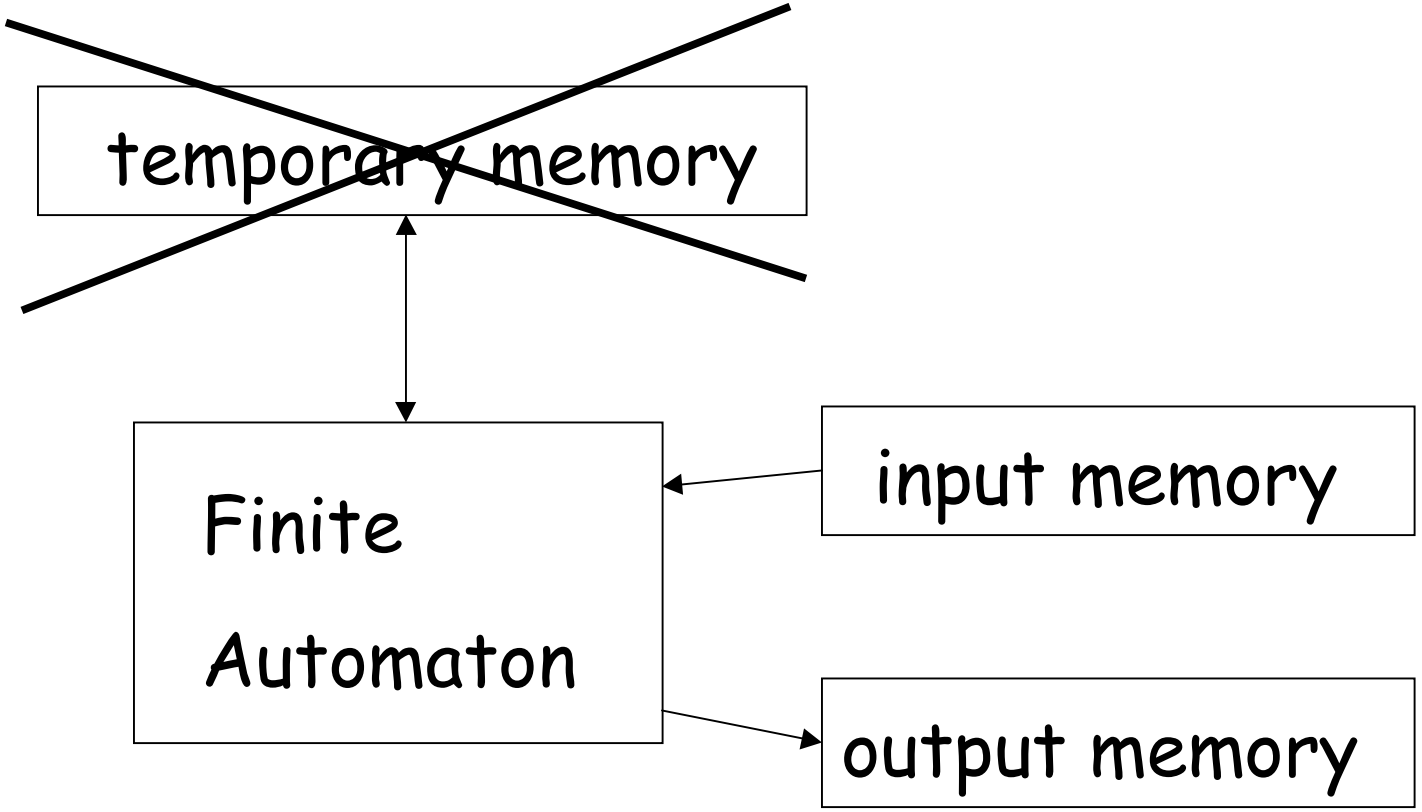
- Pushdown Automata

  Stack

- Turing Machines

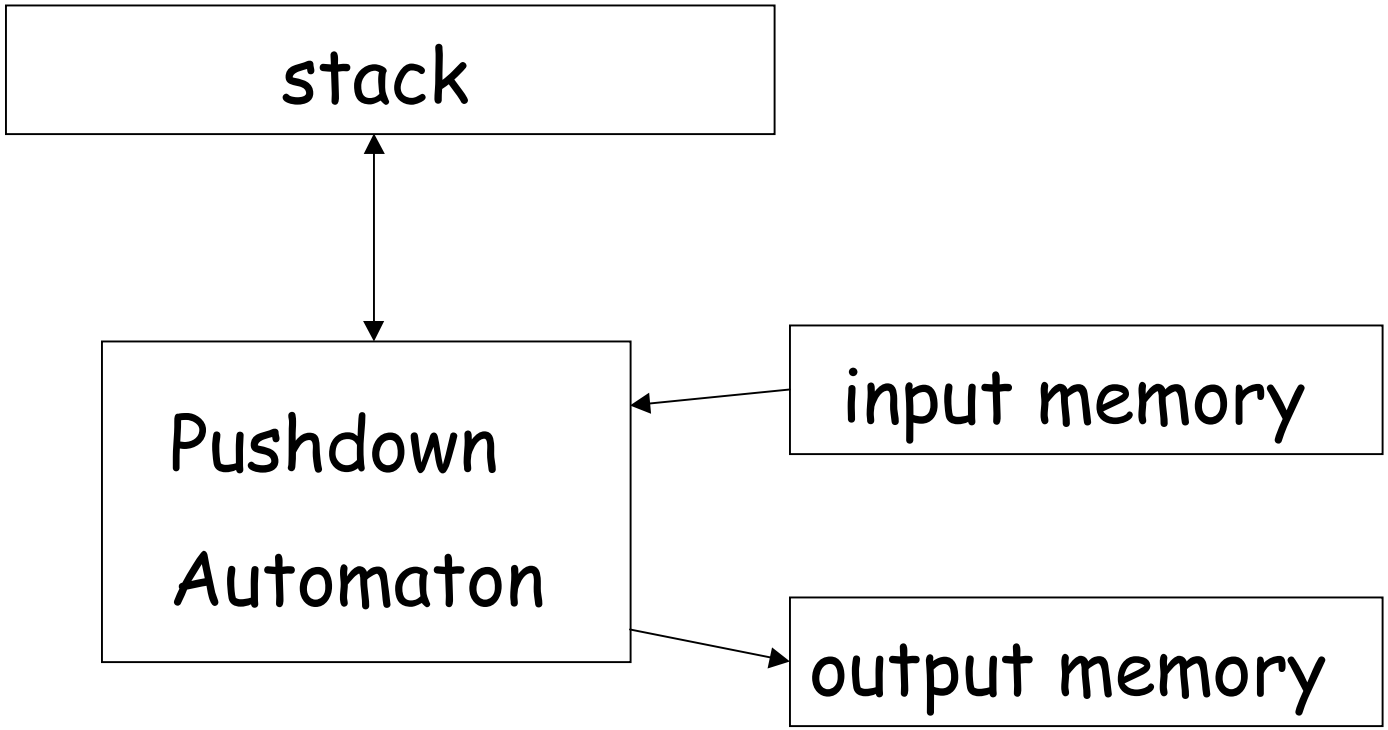
  Unlimited memory

Finite Automaton



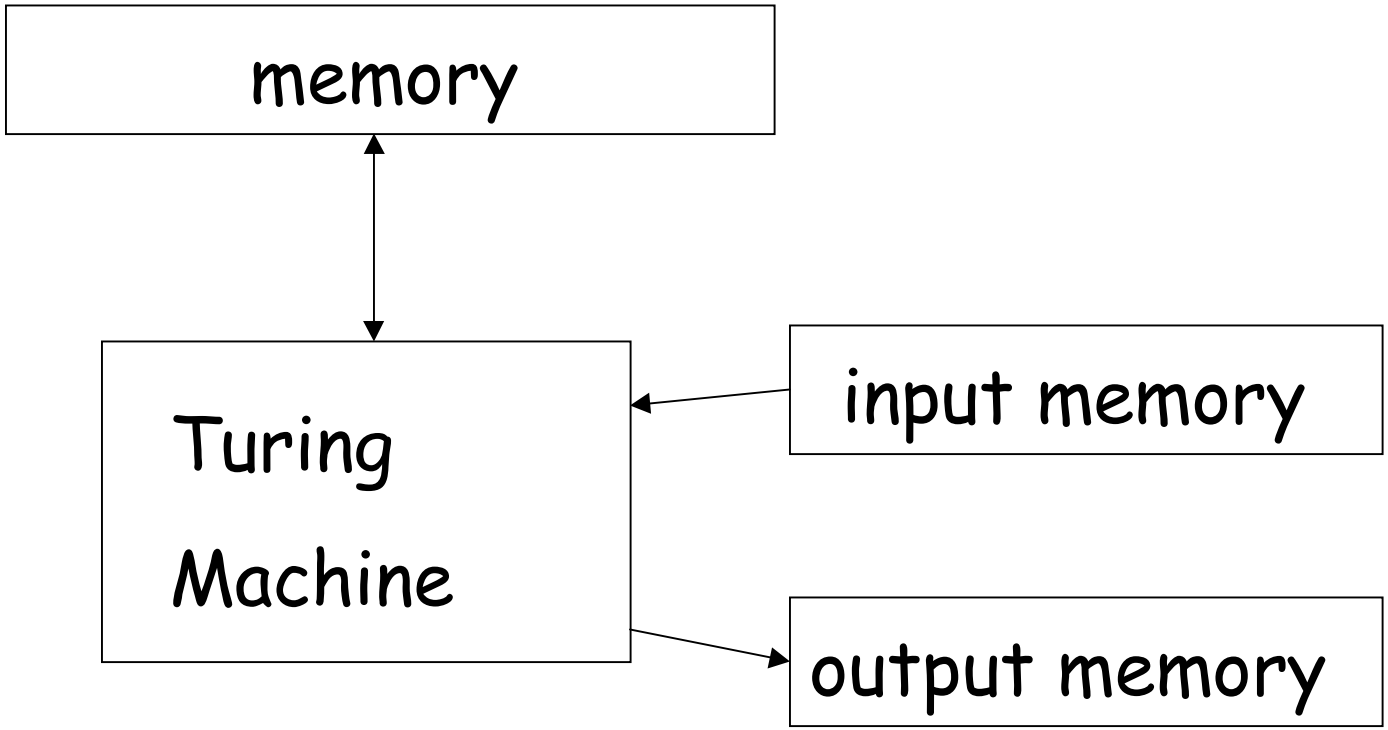
Vending Machines (small computing power)

Pushdown Automaton



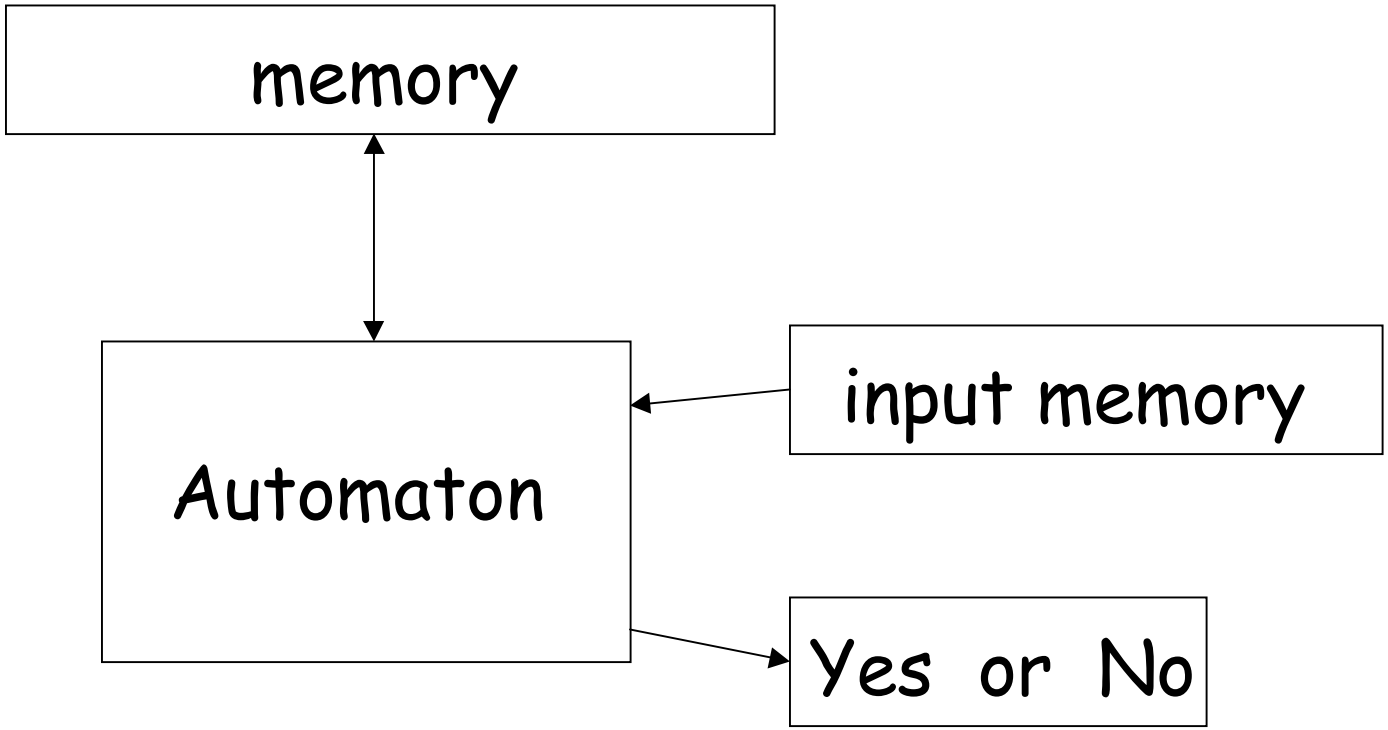
Programming Languages (medium computing power)

Turing Machine



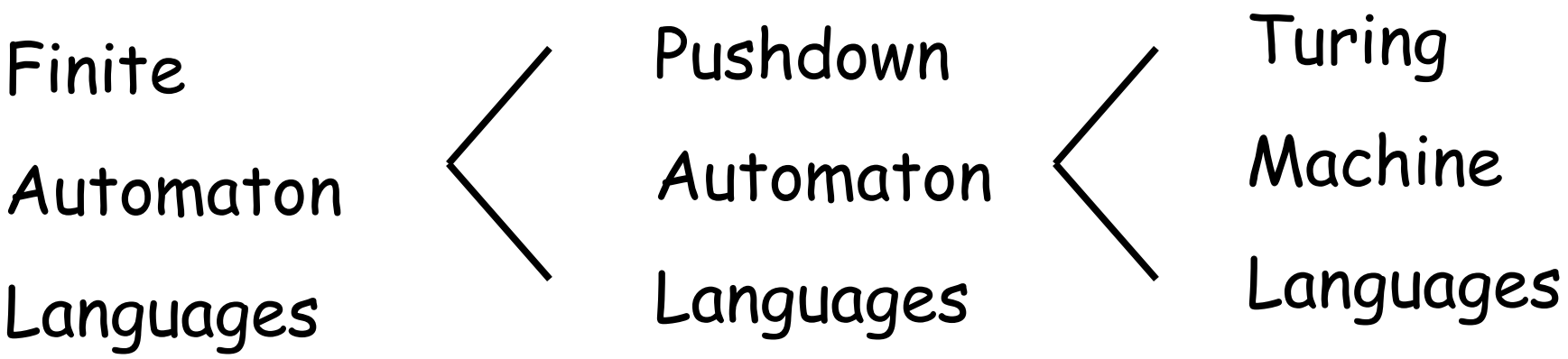
Algorithms (highest computing power)

Accepter



Language = inputs for which answer is Yes

We will show


Finite
Automaton
Languages
Pushdown
Automaton
Languages
Turing
Machine
Languages

We will show

- How to parse Programming Languages

- Some problems have no algorithms

  (cannot be solved)

Mathematical Preliminaries

- Sets
- Functions
- Relations
- Graphs
- Proof Techniques

SETS

A set is a collection of elements

$A = \{ 1, 2, 3 \}$

$B = \{$ *train, car, bicycle, airplane* $\}$

We say:

1 is in A

ship is not in B

Set Representations

$C = \{ a, b, c, d, e, f, g, h, i, j, k \}$

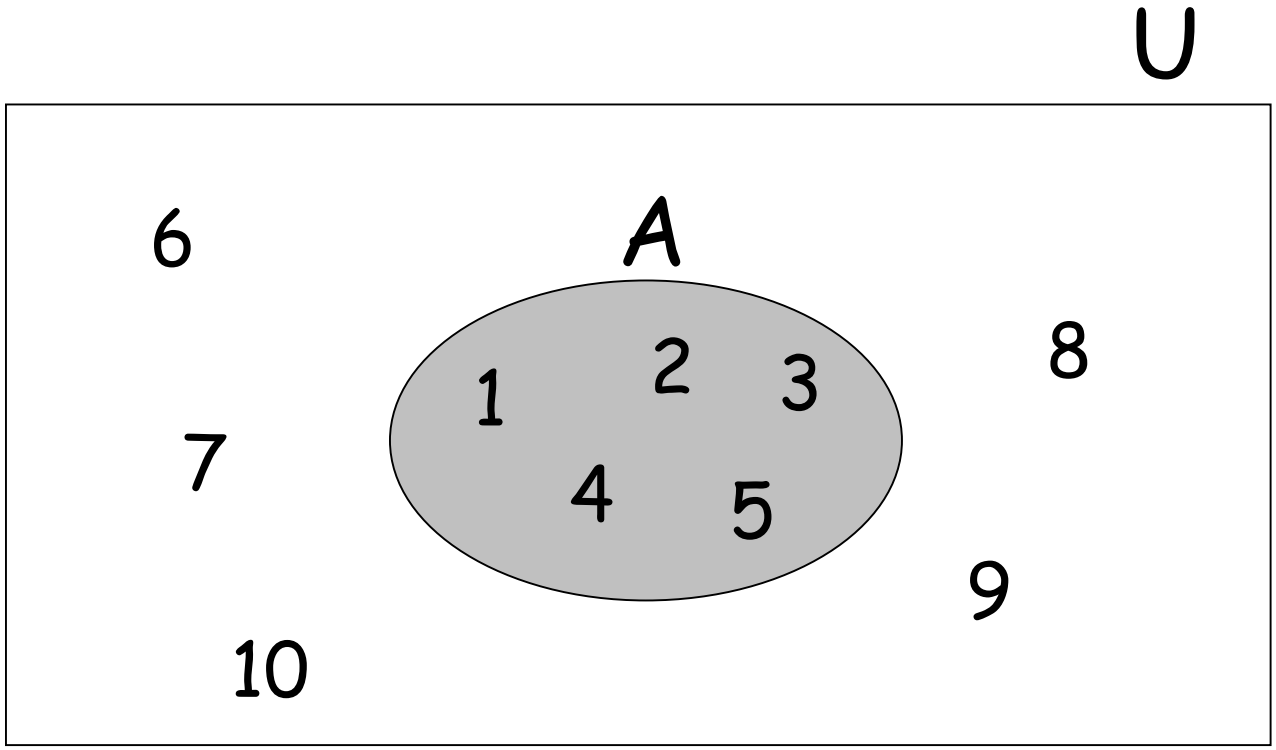
$C = \{ a, b, \ldots, k \}$ → *Finite set*

$S = \{ 2, 4, 6, \ldots \}$ → *Infinite set*

$S = \{ j : j > 0, \text{ and } j = 2k \text{ for some } k>0 \}$

$S = \{ j : j \text{ is nonnegative and even} \}$

$A = \{ 1, 2, 3, 4, 5 \}$



Universal Set: all possible elements

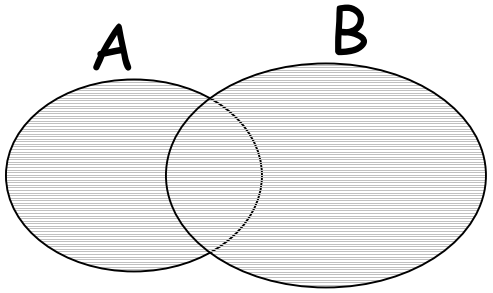
$U = \{ 1 , \ldots , 10 \}$

Set Operations

$A = \{ 1, 2, 3 \}$ $B = \{ 2, 3, 4, 5\}$

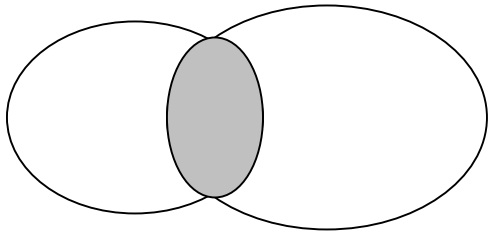
- Union

  $A \cup B = \{ 1, 2, 3, 4, 5 \}$



- Intersection

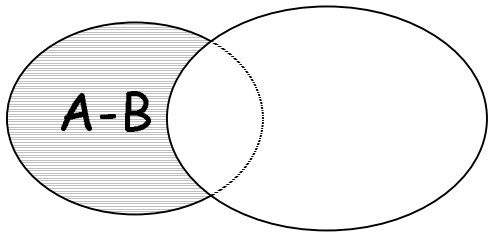
  $A \cap B = \{ 2, 3 \}$



- Difference

  $A - B = \{ 1 \}$

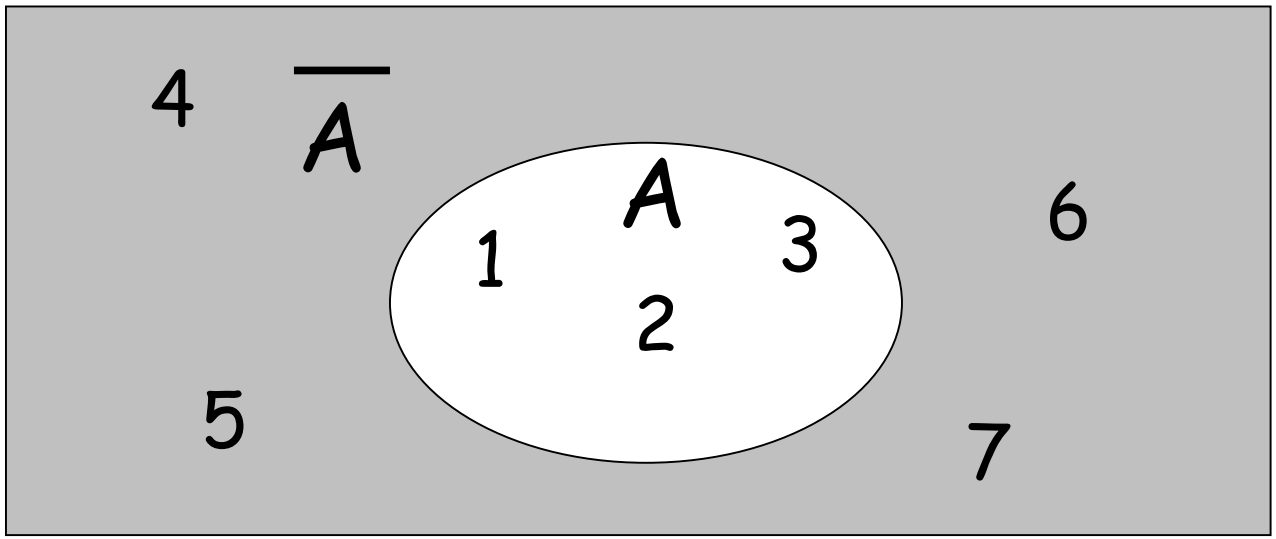
  $B - A = \{ 4, 5 \}$



- Complement

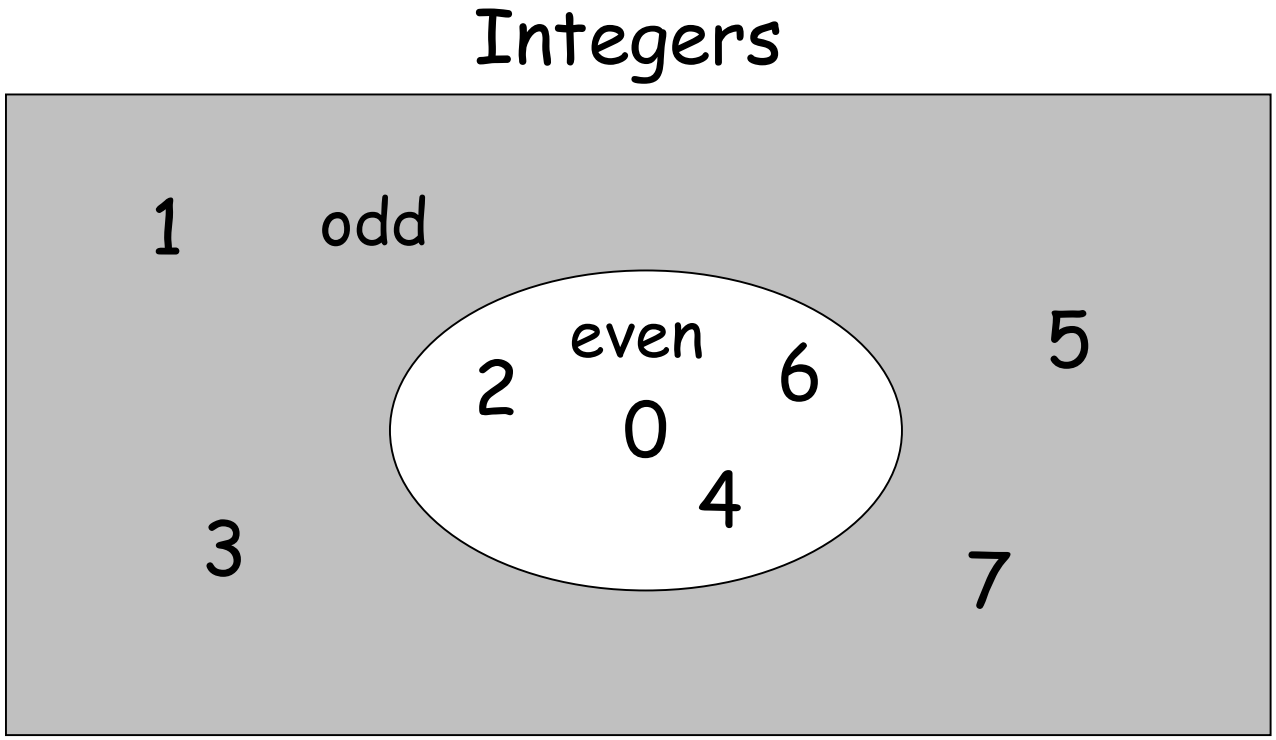
Universal set = {1, …, 7}

$A = \{ 1, 2, 3 \}$ ⇒ $\overline{A} = \{ 4, 5, 6, 7 \}$



$\overline{\overline{A}} = A$

$\overline{\{ \text{even integers} \}} = \{ \text{odd integers} \}$



DeMorgan's Laws

$$\overline{A \cup B} = \overline{A} \cap \overline{B}$$

$$\overline{A \cap B} = \overline{A} \cup \overline{B}$$

Empty, Null Set: $\emptyset$

$\emptyset = \{\}$

$S \cup \emptyset = S$

$S \cap \emptyset = \emptyset$

$S - \emptyset = S$

$\emptyset - S = \emptyset$

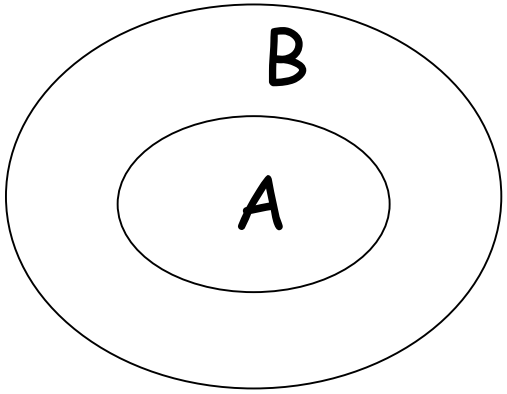
$\overline{\emptyset}$ = Universal Set

Subset

$A = \{ 1, 2, 3\}$ $B = \{ 1, 2, 3, 4, 5 \}$

$A \subseteq B$

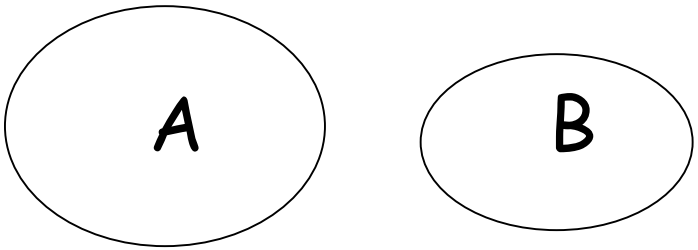
Proper Subset: $A \subset B$



Disjoint Sets

$A = \{ 1, 2, 3 \}$ $B = \{ 5, 6\}$

$A \cap B = \emptyset$



Set Cardinality

- For finite sets

  $A = \{ 2, 5, 7 \}$

  $|A| = 3$

Powersets

A powerset is a set of sets

$S = \{ a, b, c \}$

Powerset of S = the set of all the subsets of S

$2^S = \{ \emptyset, \{a\}, \{b\}, \{c\}, \{a, b\}, \{a, c\}, \{b, c\}, \{a, b, c\} \}$

Observation: $| 2^S | = 2^{|S|}$ $\quad (8 = 2^3)$

Cartesian Product

$A = \{ 2, 4 \}$ $B = \{ 2, 3, 5 \}$

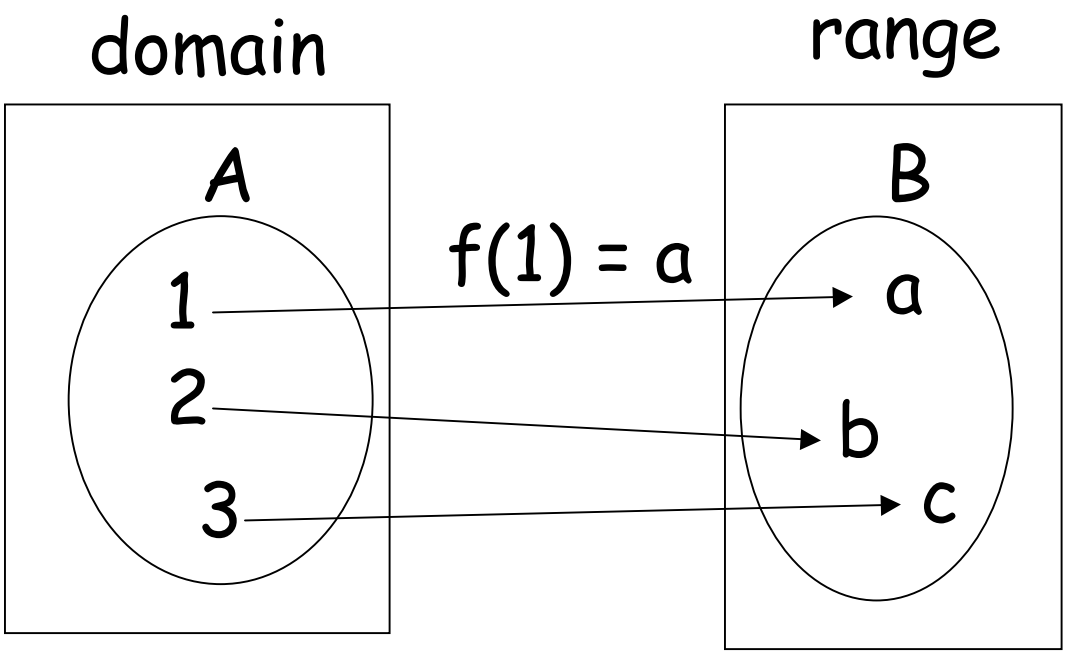
$A \times B = \{ (2, 2), (2, 3), (2, 5), (4, 2), (4, 3), (4, 4) \}$

$|A \times B| = |A| \ |B|$

Generalizes to more than two sets

$A \times B \times \ldots \times Z$

FUNCTIONS



f : A -> B

If A = domain

then f is a total function

otherwise f is a partial function

RELATIONS

$R = \{(x_1, y_1), (x_2, y_2), (x_3, y_3), \ldots\}$

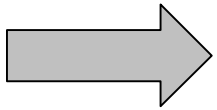
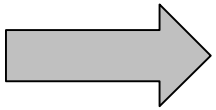
$x_i \; R \; y_i$

e. g. if R = '>': $2 > 1$, $3 > 2$, $3 > 1$

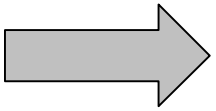
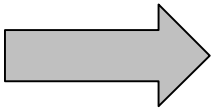
In relations a x_i can be repeated

In functions a x_i cannot be repeated

Equivalence Relations

- Reflexive: x R x
- Symmetric: x R y  y R x
- Transitive: x R Y and y R z  x R z

Example: R = '='

- x = x
- x = y  y = x
- x = y and y = z  x = z

Equivalence Classes

For equivalence relation R

equivalence class of $x = \{y : x \, R \, y\}$

Example:

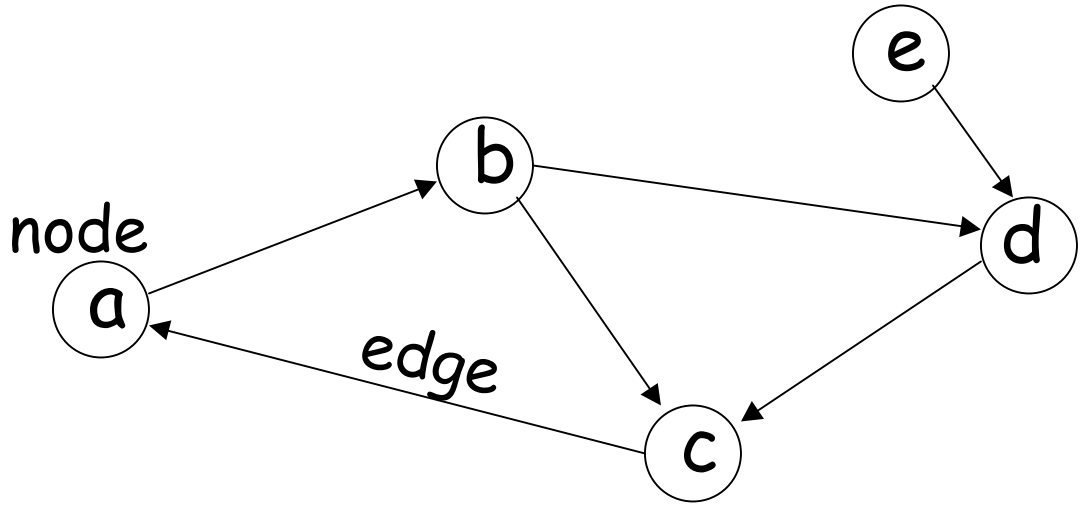
$R = \{ (1, 1), (2, 2), (1, 2), (2, 1),$
$(3, 3), (4, 4), (3, 4), (4, 3) \}$

Equivalence class of $1 = \{1, 2\}$

Equivalence class of $3 = \{3, 4\}$

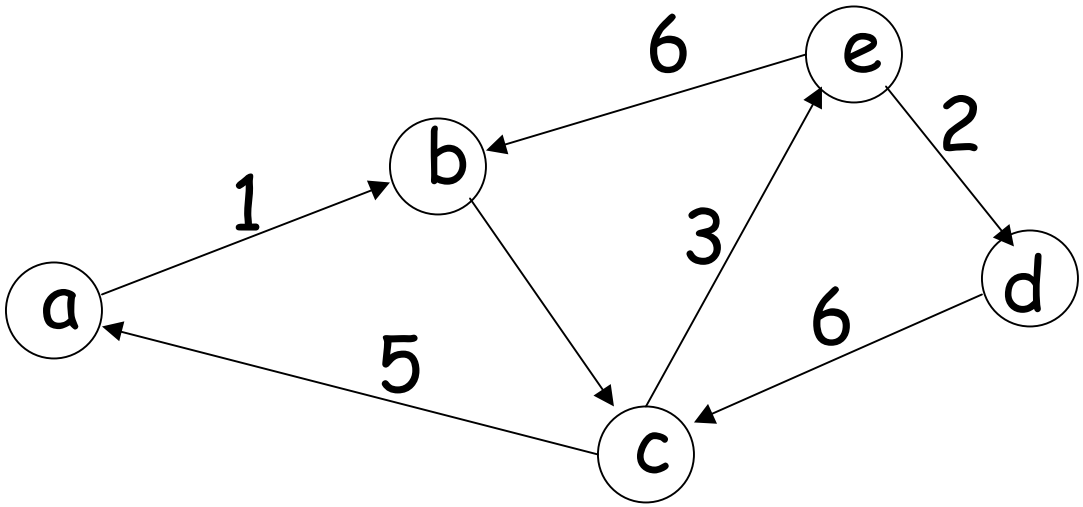
GRAPHS

A directed graph

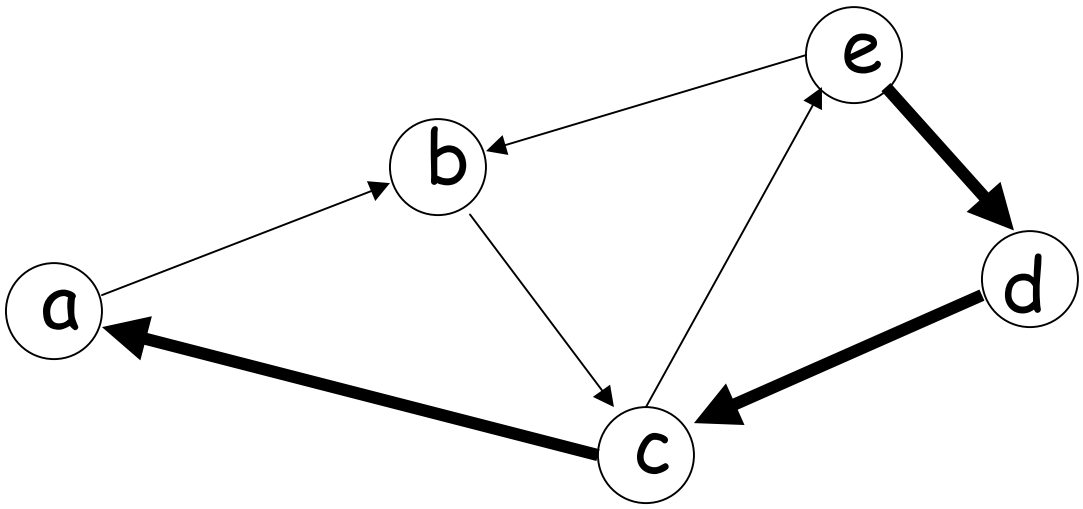


- Nodes (Vertices) V = { a, b, c, d, e }
- Edges E = { (a, b), (b, c), (c, a), (b, d), (d, c), (e, d) }

Labeled Graph



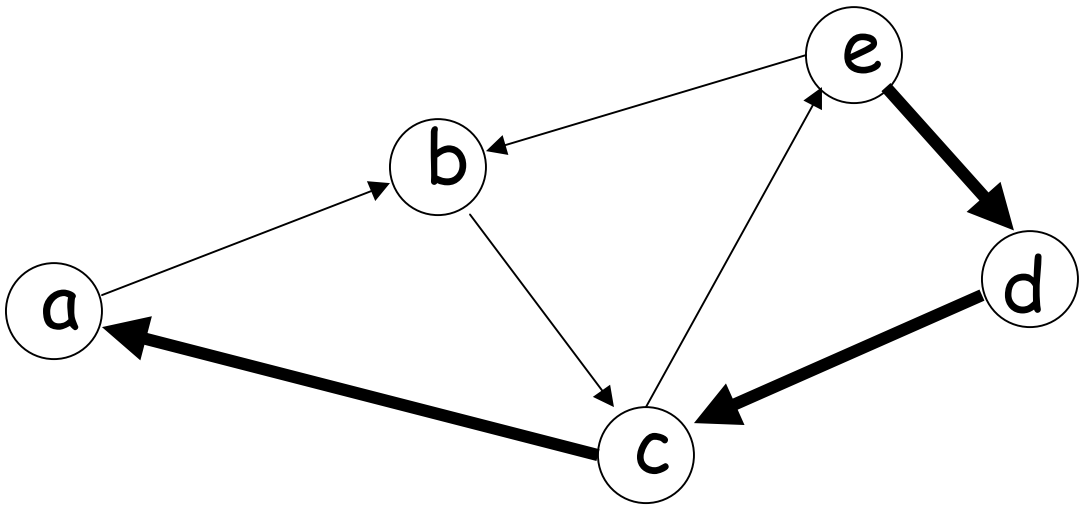
Walk



Walk is a sequence of adjacent edges

(e, d), (d, c), (c, a)

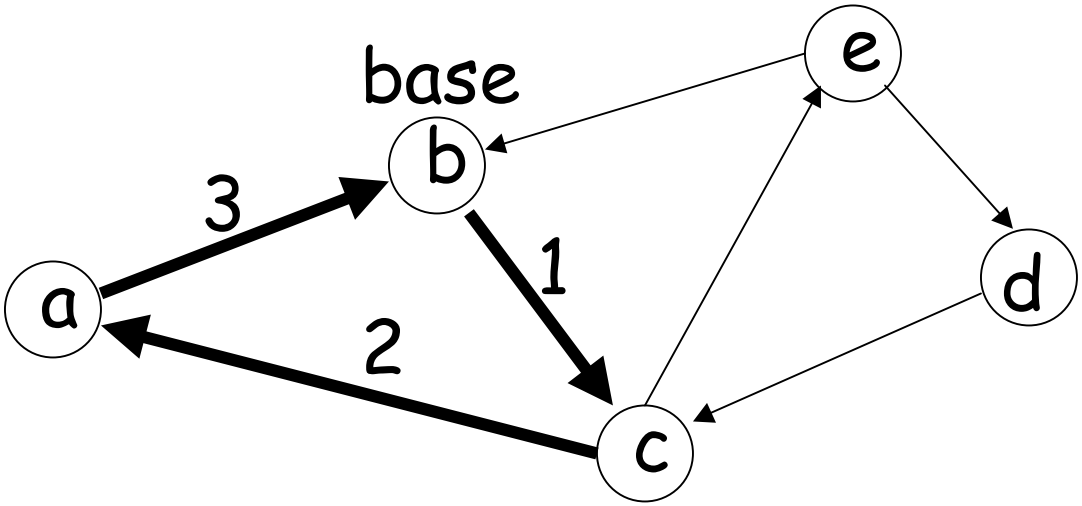
Path



Path is a walk where no edge is repeated

Simple path: no node is repeated

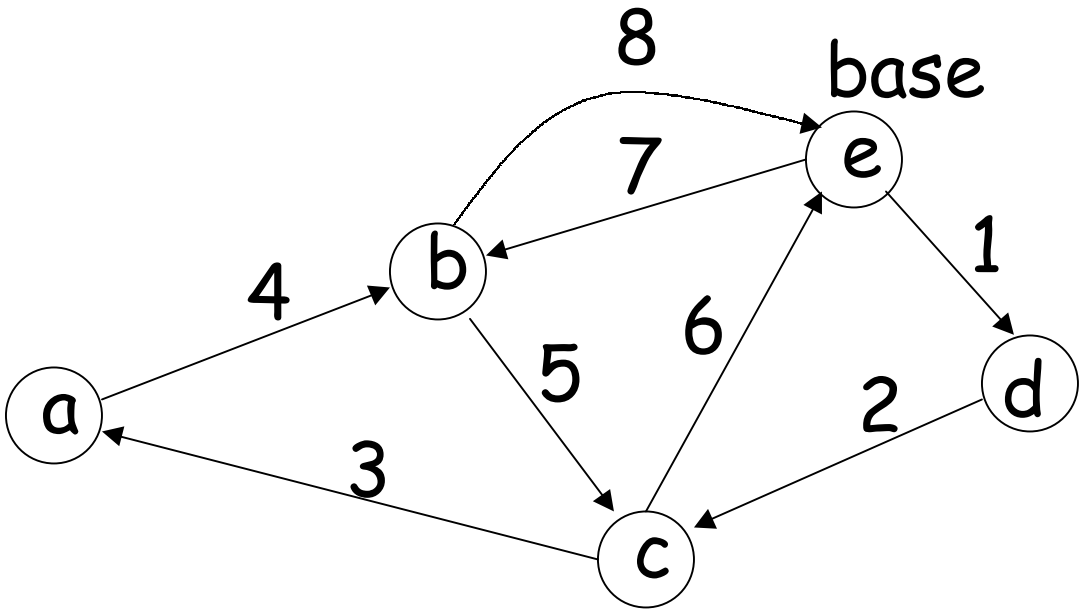
Cycle



Cycle: a walk from a node (base) to itself

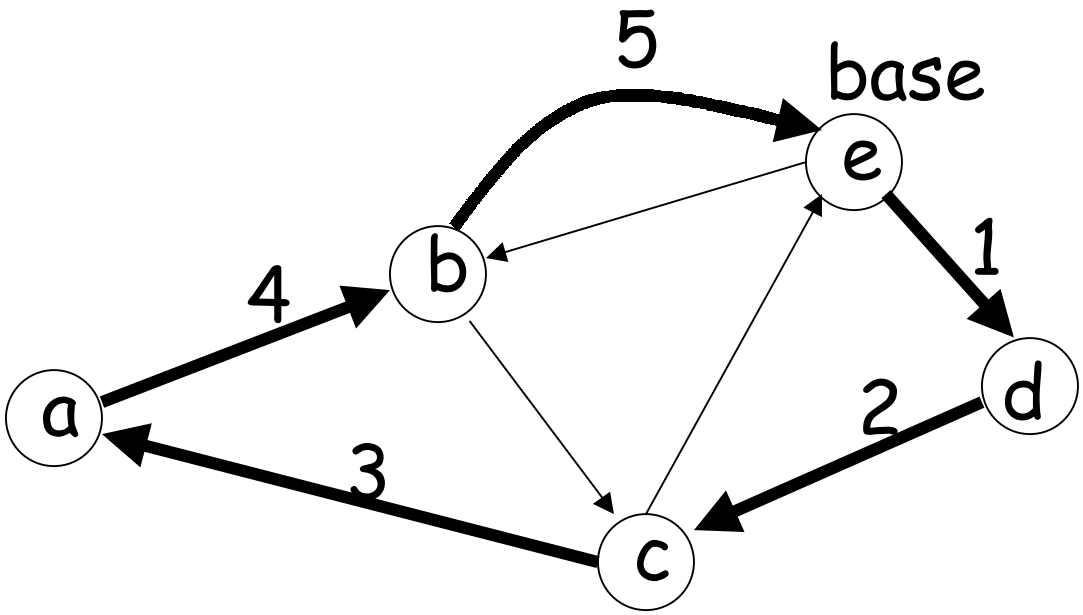
Simple cycle: only the base node is repeated

Euler Tour



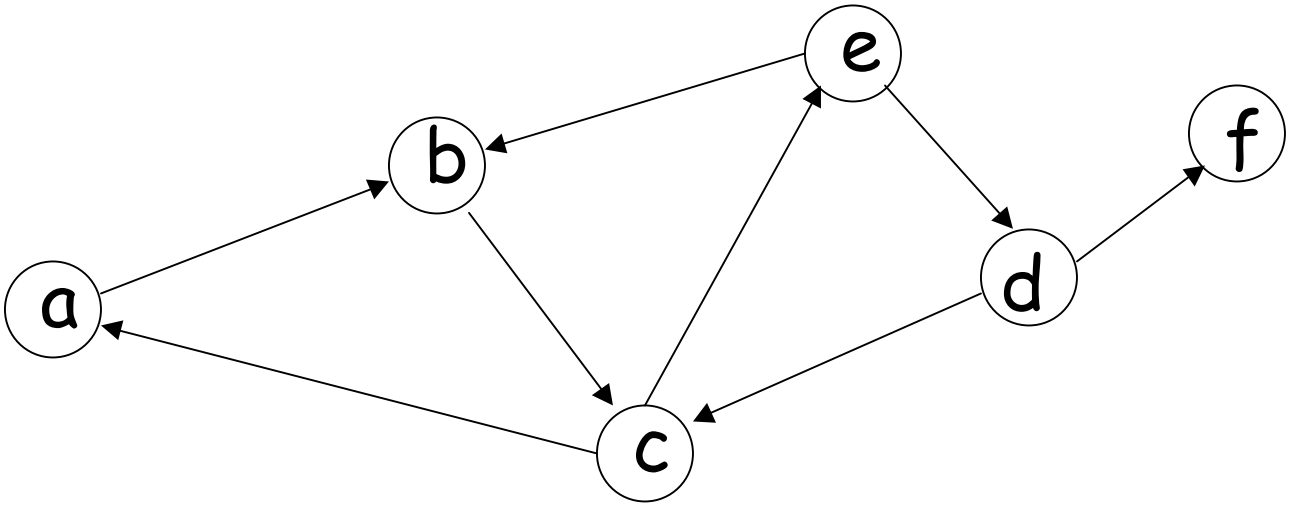
A cycle that contains each edge once

Hamiltonian Cycle

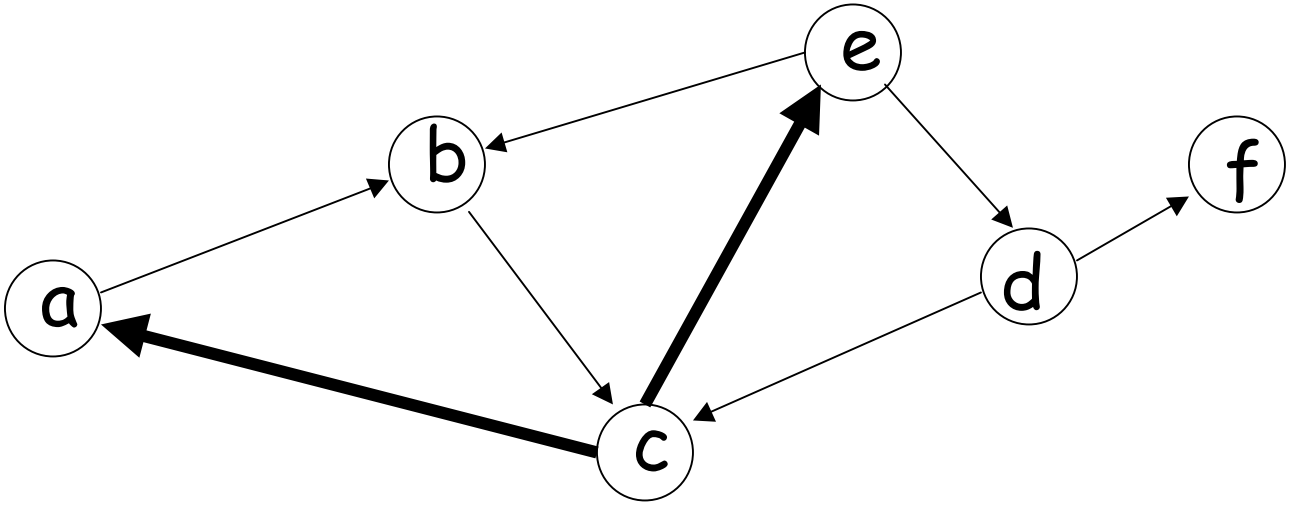


A simple cycle that contains all nodes

Finding All Simple Paths



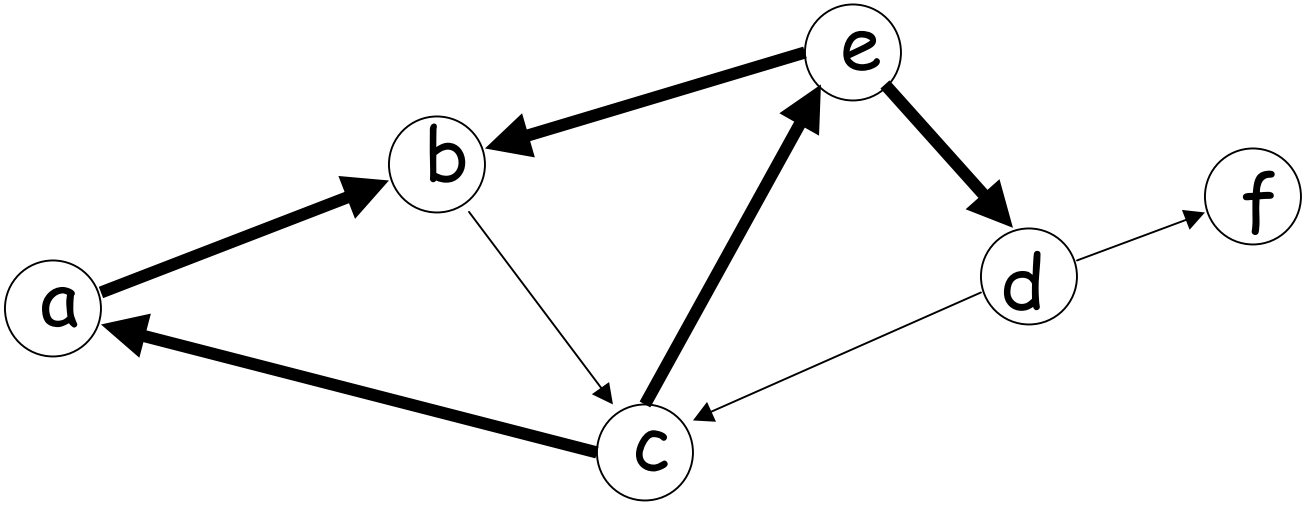
Step 1



(c, a)

(c, e)

Step 2



(c, a)

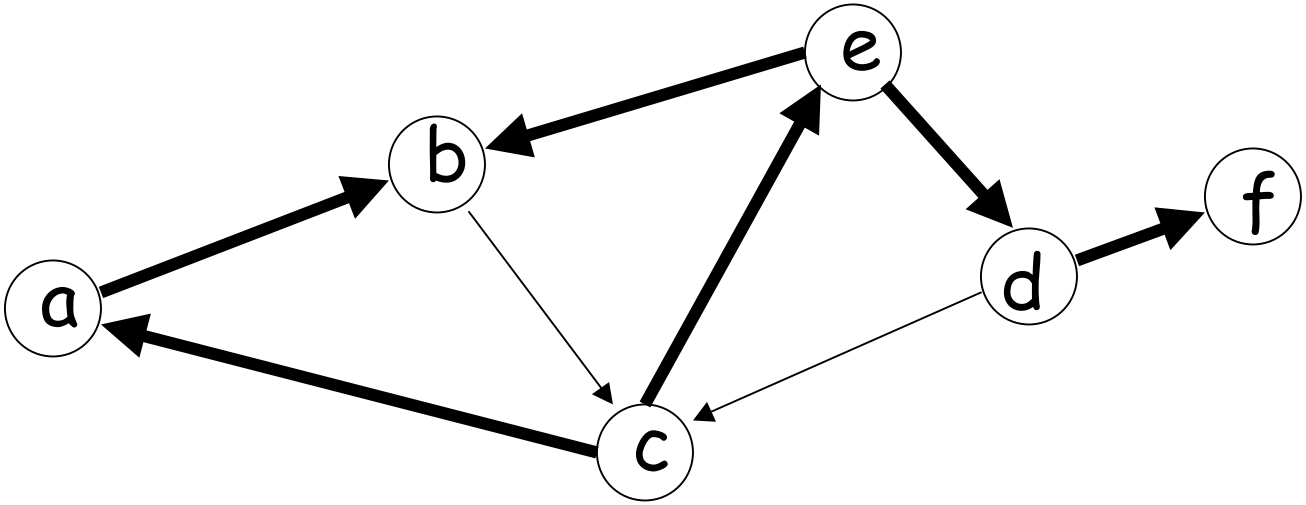
(c, a), (a, b)

(c, e)

(c, e), (e, b)

(c, e), (e, d)

Step 3



(c, a)

(c, a), (a, b)

(c, e)

(c, e), (e, b)

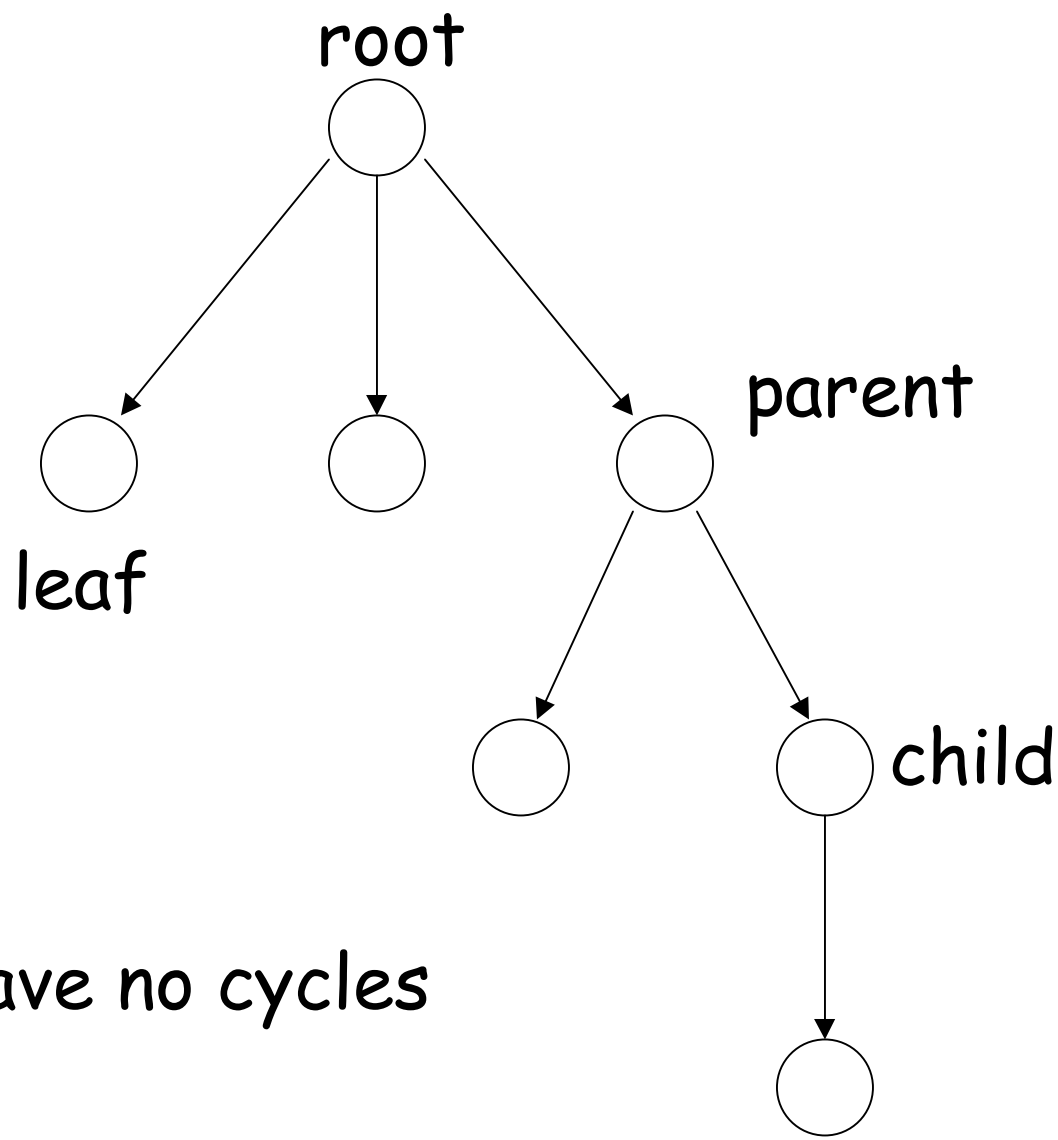
(c, e), (e, d)

(c, e), (e, d), (d, f)

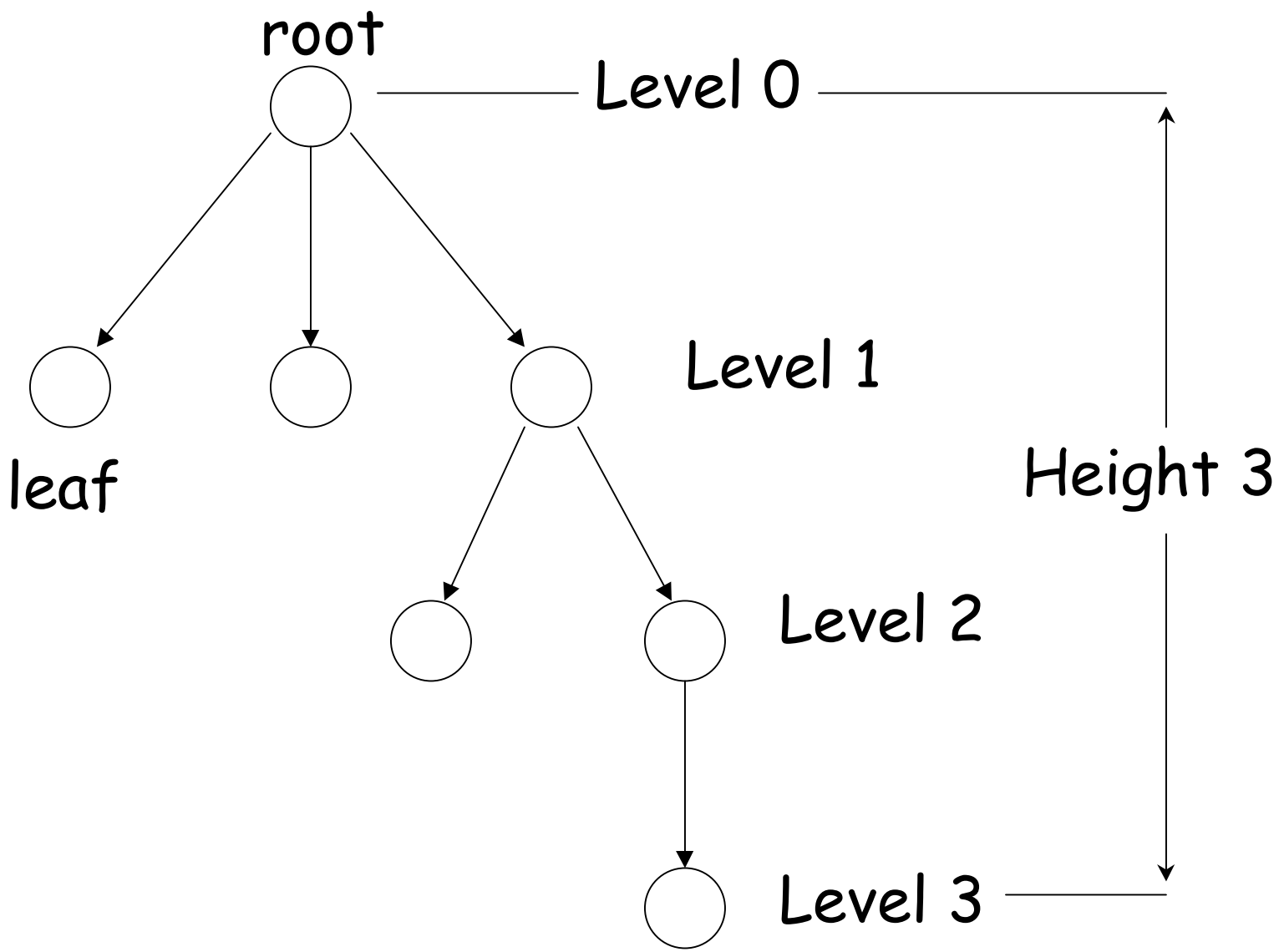
Repeat the same

for each starting node

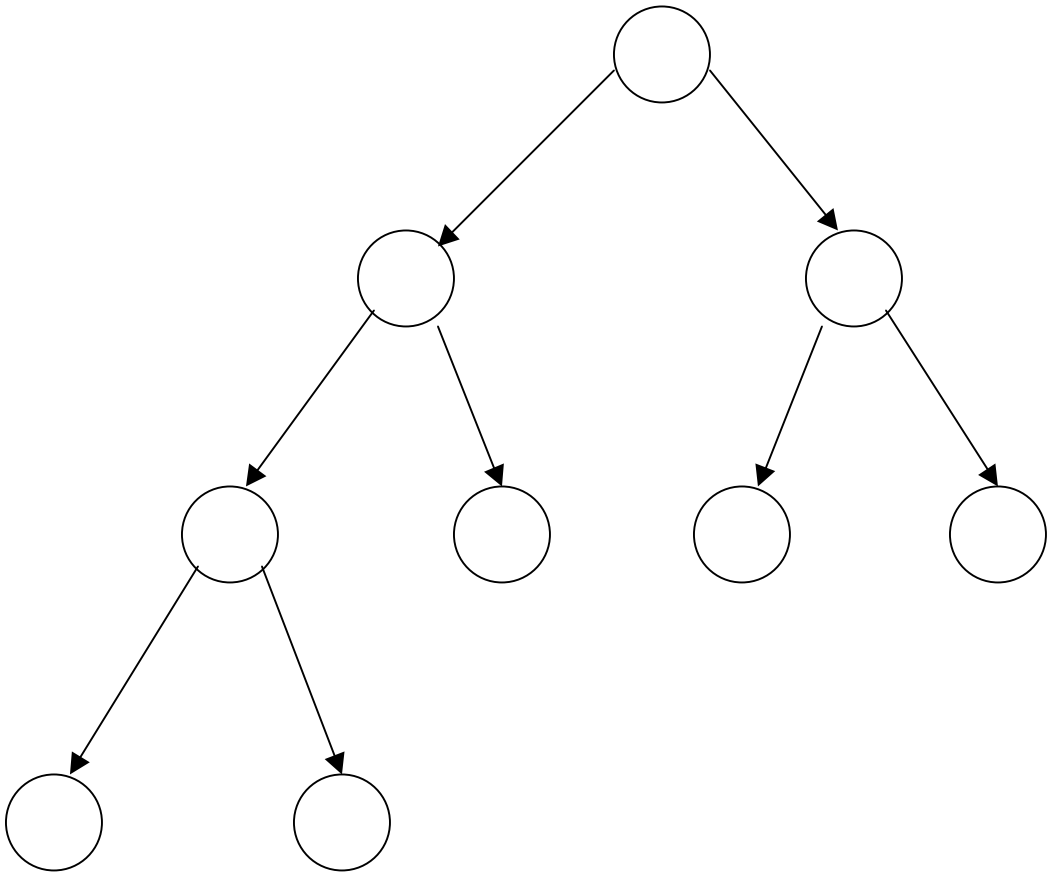
Trees



Trees have no cycles


root
Level 0
Level 1
leaf
Height 3
Level 2
Level 3

Binary Trees



PROOF TECHNIQUES

- Proof by induction

- Proof by contradiction

Induction

We have statements $P_1, P_2, P_3, \ldots$

If we know

- for some k that $P_1, P_2, \ldots, P_k$ are true
- for any n >= k that

$P_1, P_2, \ldots, P_n$ imply P_{n+1}

Then

Every P_i is true

Proof by Induction

- Inductive basis

  Find $P_1, P_2, \ldots, P_k$ which are true

- Inductive hypothesis

  Let's assume $P_1, P_2, \ldots, P_n$ are true,

  for any n >= k

- Inductive step

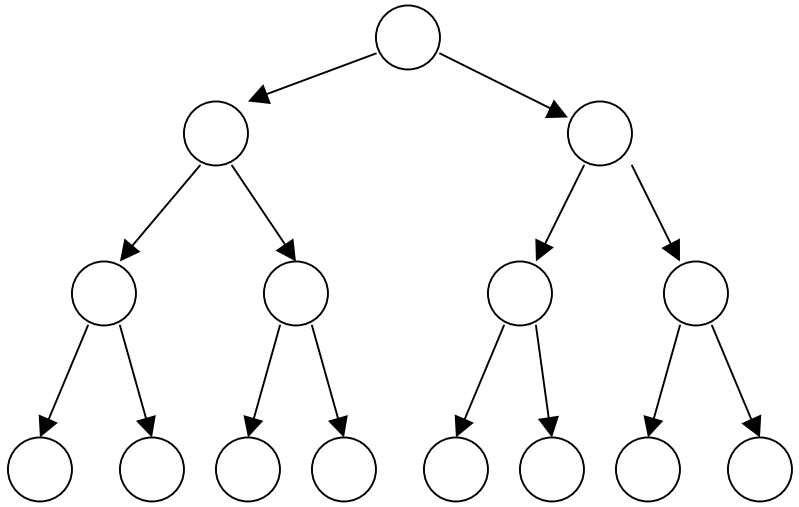
  Show that P_{n+1} is true

Example

Theorem: A binary tree of height n has at most 2^n leaves.

Proof:

let l(i) be the number of leaves at level i



l(0) = 1

l(3) = 8

We want to show: $l(i) <= 2^i$

- Inductive basis

  $l(0) = 1$ (the root node)

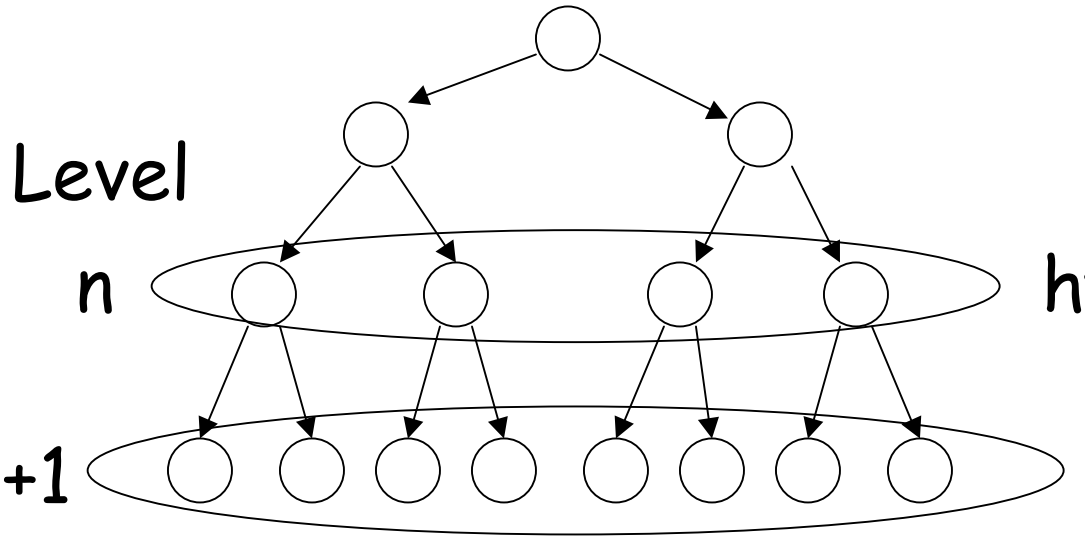
- Inductive hypothesis

  Let's assume $l(i) <= 2^i$ for all $i = 0, 1, \ldots, n$

- Induction step

  we need to show that $l(n + 1) <= 2^{n+1}$

Induction Step



hypothesis: $l(n) <= 2^n$

$l(n+1) <= 2\ l(n) <= 2\ 2^n = 2^{n+1}$

Remark

Recursion is another thing

Example of recursive function:

$$f(n) = f(n-1) + f(n-2)$$

$$f(0) = 1,\ f(1) = 1$$

Proof by Contradiction

We want to prove that a statement P is true

- we assume that P is false
- then we arrive at an incorrect conclusion
- therefore, statement P must be true

Example

Theorem: sqrt(2) is not rational

Proof:

Assume by contradiction that it is rational

sqrt(2) = n/m

n and m have no common factors

We will show that this is impossible

Sqrt(2) = n/m ⇒ $2\,m^2 = n^2$

Therefore, n^2 is even ⇒ n is even

$n = 2\,k$

$2\,m^2 = 4k^2$ ⇒ $m^2 = 2k^2$ ⇒ m is even

$m = 2\,k'$

m and n have common factor 2

Contradiction!