Discrete Mathematics

Chapter 10

Section 10.1

Graphs and graph Models

Introduction to Graphs

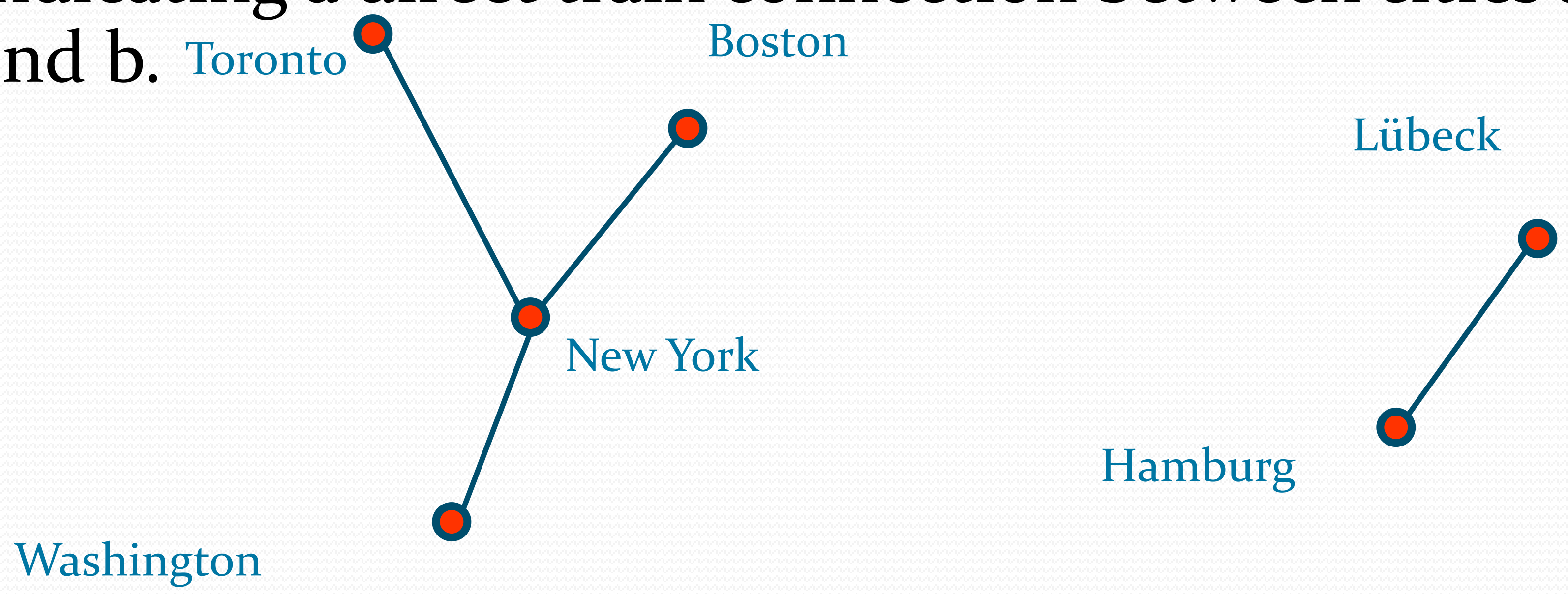
- **Definition:** A **simple graph** G = (V, E) consists of V, a nonempty set of vertices, and E, a set of **unordered pairs** of distinct elements of V called edges.
- For each $e \in E$, $e = \{u, v\}$ where $u, v \in V$.
- An undirected graph (not simple) may contain:
  - Loops: An edge e is a loop if $e = \{u, u\}$ for some $u \in V$
  - Duplicate edges: A graph is called a multi-graph if there is at least one duplicate edge.

Introduction to Graphs

- **Definition:** A **directed graph** $G = (V, E)$ consists of a set V of vertices and a set E of edges that are ordered pairs of elements in V.
- For each $e \in E$, $e = (u, v)$ where $u, v \in V$.
- An edge e is a loop if $e = (u, u)$ for some $u \in V$.
- A simple graph is just like a directed graph, but with no specified direction of its edges.

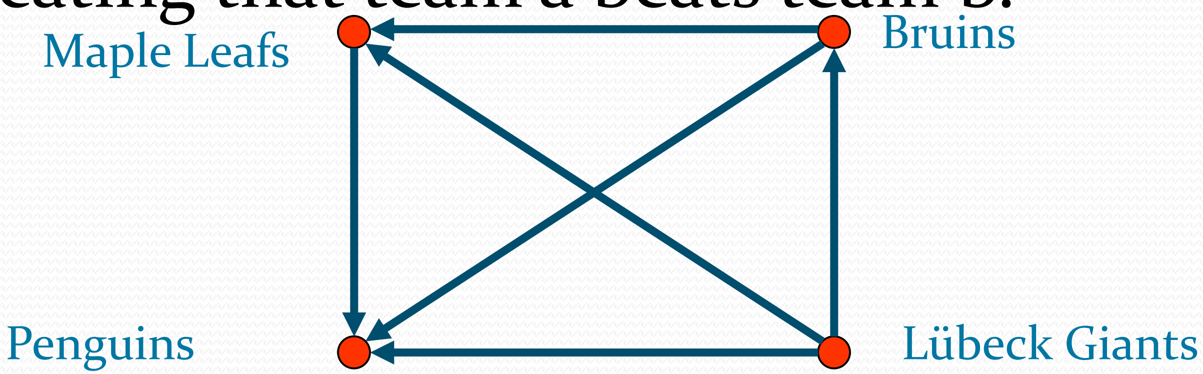
Graph Models

- **Example I:** How can we represent a network of (bi-directional) railways connecting a set of cities?
- We should use a **simple graph** with an edge {a, b} indicating a direct train connection between cities a and b.



Graph Models

- **Example II:** In a round-robin tournament, each team plays against each other team exactly once. How can we represent the results of the tournament (which team beats which other team)?
- We should use a **directed graph** with an edge (a, b) indicating that team a beats team b.



Exercise 1

- What might the nodes/edges be if we modeled the following data? Would the graph be best undirected or directed?
  - Influence Graph: Identifying data where one person has influence over another.
  - Computer network
  - Road Map
  - The World Wide Web

Graph Terminology

- **Definition:** Two vertices u and v in an undirected graph G are called **adjacent** (or **neighbors**) in G if {u, v} is an edge in G.
- If e = {u, v}, the edge e is called **incident with** the vertices u and v. The edge e is also said to **connect** u and v.
- The vertices u and v are called **endpoints** of the edge {u, v}.

Graph Terminology

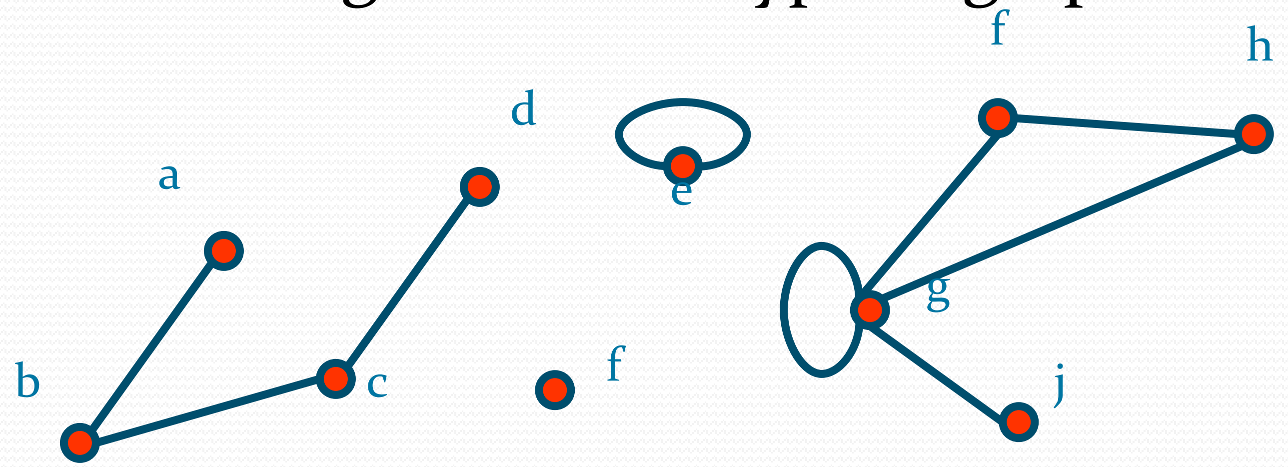
- **Definition:** The **degree** of a vertex in an undirected graph is the number of edges incident with it, except that a loop at a vertex contributes twice to the degree of that vertex.
- In other words, you can determine the degree of a vertex in a displayed graph by **counting the lines** that touch it.
- The degree of the vertex v is denoted by **deg(v).**

Graph Terminology

- A vertex of degree 0 is called **isolated**, since it is not adjacent to any vertex.
- **Note:** A vertex with a **loop** at it has at least degree 2 and, by definition, is **not isolated**, even if it is not adjacent to any **other** vertex.
- A vertex of degree 1 is called **pendant**. It is adjacent to exactly one other vertex.

Graph Terminology

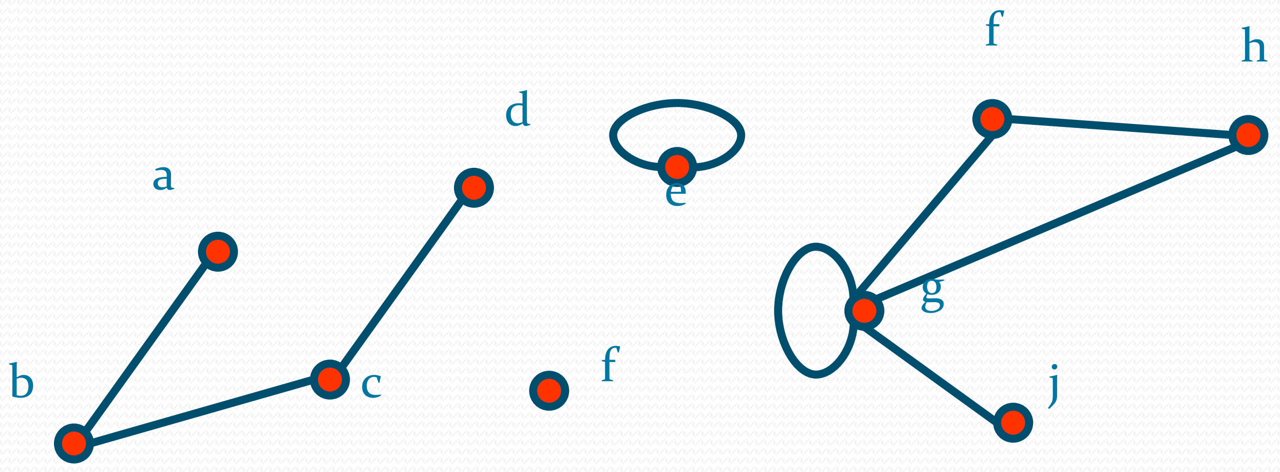
- **Example:** Which vertices in the following graph are isolated, which are pendant, and what is the maximum degree? What type of graph is it?



Solution: Vertex f is isolated, and vertices a, d and j are pendant. The maximum degree is deg(g) = 5.
This graph is a non-simple undirected graph.

Graph Terminology

- Determine the number of its edges and the sum of the degrees of all its vertices:



Result: There are 9 edges, and the sum of all degrees is 18. This is easy to explain: Each new edge increases the sum of degrees by exactly two.

Graph Terminology

- **The Handshaking Theorem:** Let G = (V, E) be an undirected graph with e edges. Then
- $2e = \sum_{v \in V} \deg(v)$
  - Corollary: The total degree of any undirected graph is always even!
- **Example:** How many edges are there in a graph with 10 vertices, each of degree 6?
- **Solution:** The sum of the degrees of the vertices is $6 \cdot 10 = 60$. According to the Handshaking Theorem, it follows that $2e = 60$, so there are 30 edges.

Graph Terminology

- **Theorem:** An undirected graph has an even number of vertices of odd degree.
- **Proof:** Let V_1 and V_2 be the set of vertices of even and odd degrees, respectively (Thus $V_1 \cap V_2 = \emptyset$, and $V_1 \cup V_2 = V$).
- Then by Handshaking theorem
  - $2|E| = \sum_{v \in V} \deg(v) = \sum_{v \in V_1} \deg(v) + \sum_{v \in V_2} \deg(v)$
- Since both $2|E|$ and $\sum_{v \in V_1} \deg(v)$ are even,
- $\sum_{v \in V_2} \deg(v)$ must be even.
- Since $\deg(v)$ is odd for all $v \in V_2$, $|V_2|$ must be even.

QED

Exercise 2

- Draw a graph with the specified properties or show that no such graph exists:
  - A graph with 6 vertices with the following degrees: 1,1,2,2,3,4
  - A graph with 4 vertices of degrees 1,2,3,4
  - A *simple* graph with 4 vertices of degrees 1,2,3,4

Exercise 3

- A graph has 5 vertices of degrees 1,1,4,4, and 6. How many edges does the graph have?
- Is it possible in a group of 13 people for each to shake hands with exactly 7 others?
- Is it possible to have a graph with 15 edges where each vertex has degree 4?
- Is it possible to have a simple graph with 10 edges where each vertex has degree 4?

Graph Terminology

- **Definition:** When (u, v) is an edge of the graph G with directed edges, u is said to be **adjacent to** v, and v is said to be **adjacent from** u.
- The vertex u is called the **initial vertex (or source)** of (u, v), and v is called the **terminal vertex (or target)** of (u, v).
- The initial vertex and terminal vertex of a loop are the same.

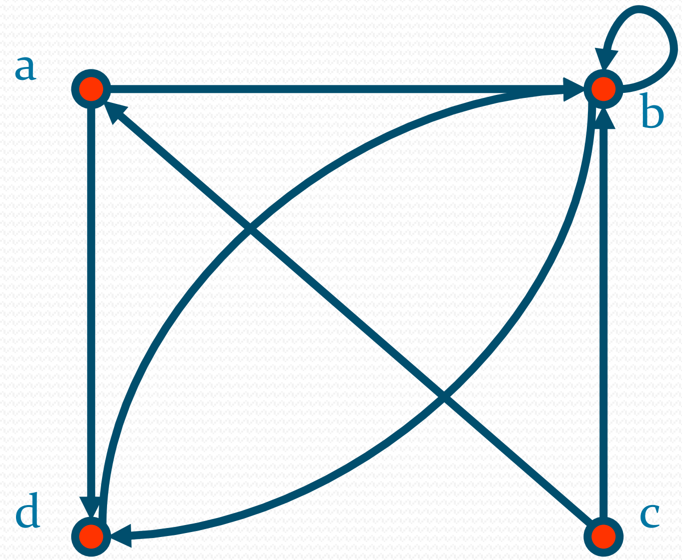
Graph Terminology

- **Definition:** In a graph with directed edges, the **in-degree** of a vertex v, denoted by **deg$^-$(v)**, is the number of edges with v as their **terminal vertex**.
- The **out-degree** of v, denoted by **deg$^+$(v)**, is the number of edges with v as their initial vertex.
- **Question:** How does adding a loop to a vertex change the in-degree and out-degree of that vertex?
- **Answer:** It increases both the in-degree and the out-degree by one.

Graph Terminology

- **Example:** What are the in-degrees and out-degrees of the vertices a, b, c, d in this graph:

$deg^-(a) = 1$
$deg^+(a) = 2$



$deg^-(b) = 4$
$deg^+(b) = 2$

$deg^-(c) = 0$
$deg^+(c) = 2$

$deg^-(d) = 2$
$deg^+(d) = 1$

Graph Terminology

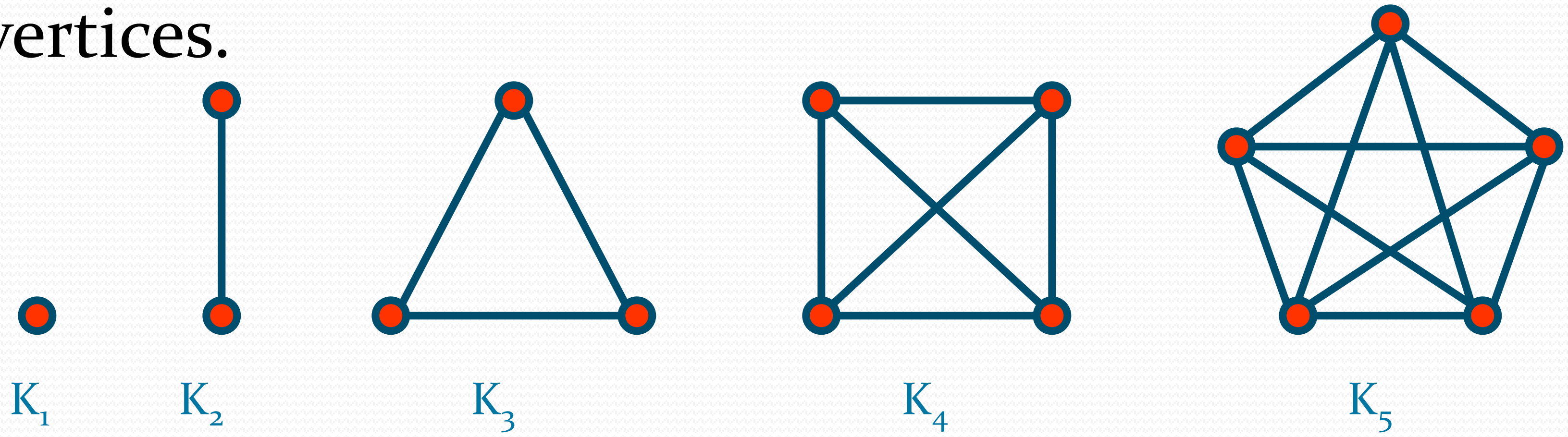
- **Theorem:** Let G = (V, E) be a graph with directed edges. Then:
- $\sum_{v \in V} \deg^-(v) = \sum_{v \in V} \deg^+(v) = |E|$
- This is easy to see, because every new edge increases both the sum of in-degrees and the sum of out-degrees by one.

Exercise 4

- What is the maximum number of edges possible in a simple graph on *n* vertices?
- What is the maximum number of edges possible in a directed graph on n vertices (loops included)?

Special Graphs

- **Definition:** The **complete graph** on n vertices, denoted by K_n, is the simple graph that contains exactly one edge between each pair of distinct vertices.

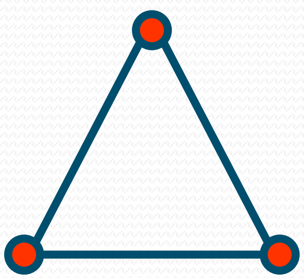


Exercise 5

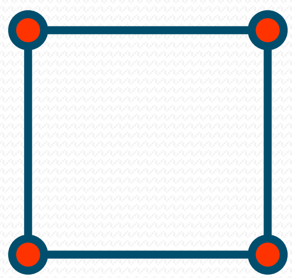
- What is the degree of each vertex in the complete graph K_9?
- What is the total degree of K_9?
- How many edges are there in K_9?
- How many edges are there in K_n?
- What is the degree of a vertex in K_n?
- What is the total degree of K_n?

Special Graphs

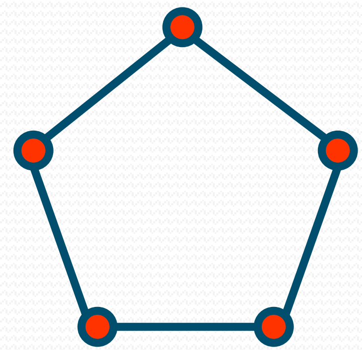
- **Definition:** The **cycle** C_n, $n \geq 3$, consists of n vertices $v_1, v_2, \ldots, v_n$ and edges $\{v_1, v_2\}, \{v_2, v_3\}, \ldots, \{v_{n-1}, v_n\}, \{v_n, v_1\}$.



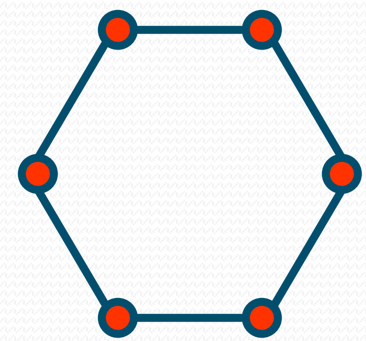
C_3



C_4



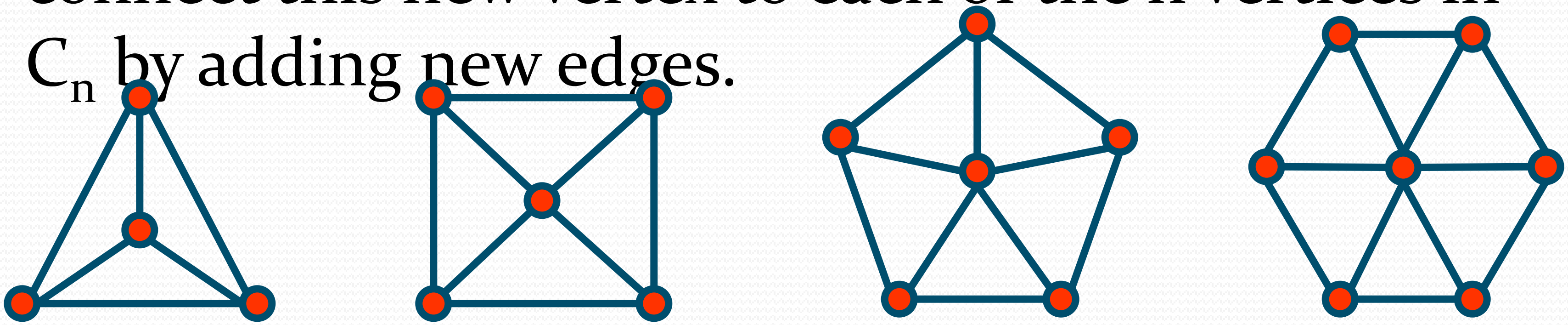
C_5



C_6

Special Graphs

- **Definition:** We obtain the **wheel** W_n when we add an additional vertex to the cycle C_n, for $n \geq 3$, and connect this new vertex to each of the n vertices in C_n by adding new edges.



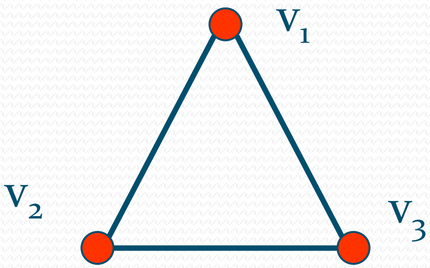
W_3 W_4 W_5 W_6

Special Graphs

- **Definition:** A simple graph is called **bipartite** if its vertex set V can be partitioned into two disjoint nonempty sets V_1 and V_2 such that every edge in the graph connects a vertex in V_1 with a vertex in V_2 (so that no edge in G connects either two vertices in V_1 or two vertices in V_2).
- For example, consider a graph that represents each person in a village by a vertex and each marriage by an edge.
- This graph is **bipartite**, because each edge connects a vertex in the **subset of males** with a vertex in the **subset of females** (if we think of traditional marriages).

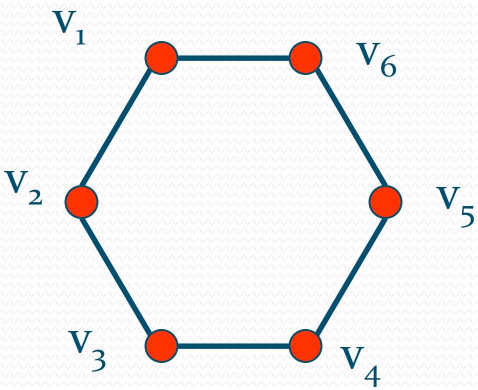
Special Graphs

- **Example I:** Is C_3 bipartite?

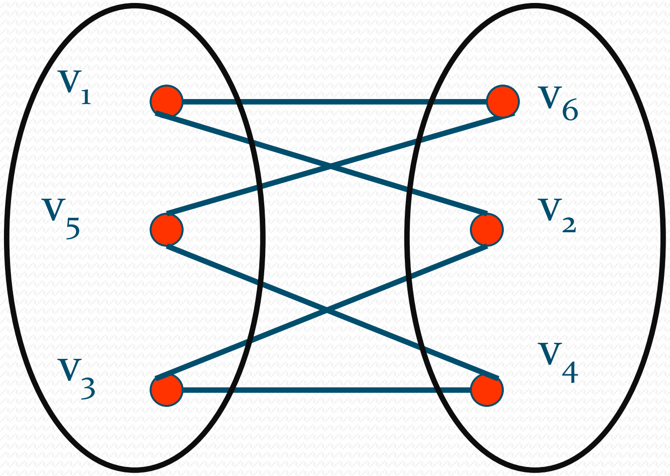


No, because there is no way to partition the vertices into two sets so that there are no edges with both endpoints in the same set.

Example II: Is C_6 bipartite?

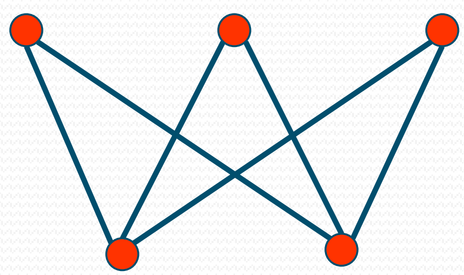


Yes, because we can display C_6 like this:

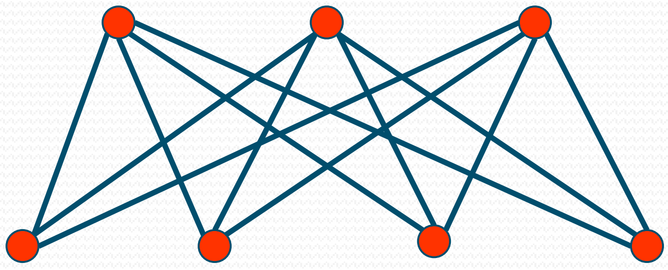


Special Graphs

- **Definition:** The **complete bipartite graph** $K_{m,n}$ is the graph that has its vertex set partitioned into two subsets of m and n vertices, respectively. Two vertices are connected if and only if they are in different subsets.



$K_{3,2}$



$K_{3,4}$

Exercise 6

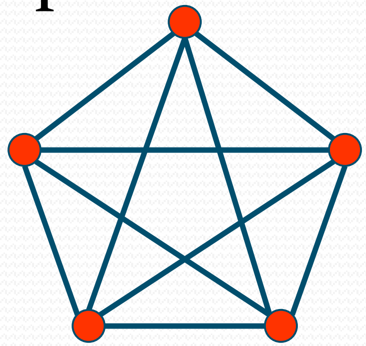
- Draw the complete bipartite graphs for $K_{2,2}$, $K_{2,3}$, and $K_{3,4}$

Exercise 8

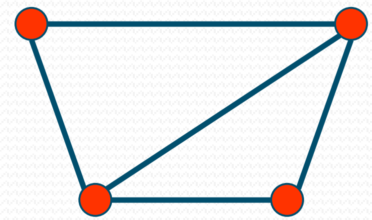
- What is the degree of each vertex in the complete bipartite graph $K_{4,5}$?

Operations on Graphs

- **Definition:** A **subgraph** of a graph $G = (V, E)$ is a graph $H = (W, F)$ where $W \subseteq V$ and $F \subseteq E$.
- **Note:** Of course, H is a valid graph, so we cannot remove any endpoints of remaining edges when creating H.
- **Example:**



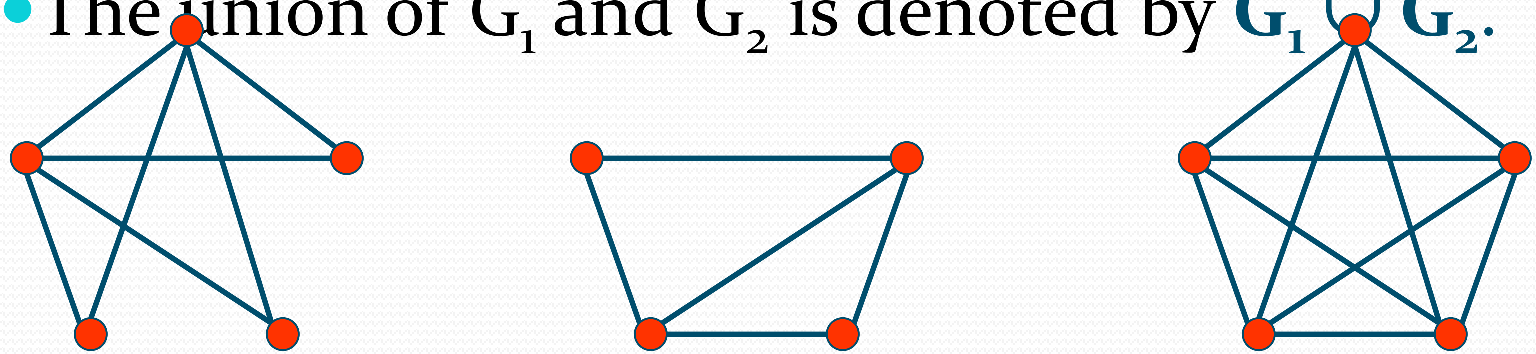
K_5



subgraph of K_5

Operations on Graphs

- **Definition:** The **union** of two simple graphs $G_1 = (V_1, E_1)$ and $G_2 = (V_2, E_2)$ is the simple graph with vertex set $V_1 \cup V_2$ and edge set $E_1 \cup E_2$.
- The union of G_1 and G_2 is denoted by $\mathbf{G_1 \cup G_2}$.



G_1 G_2 $G_1 \cup G_2 = K_5$

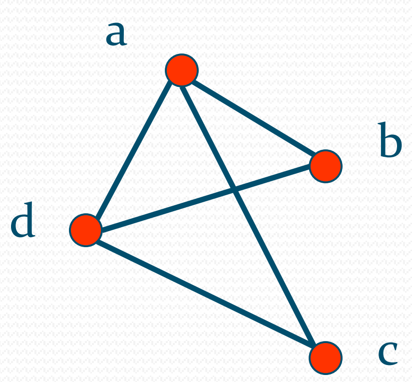
Exercise 9

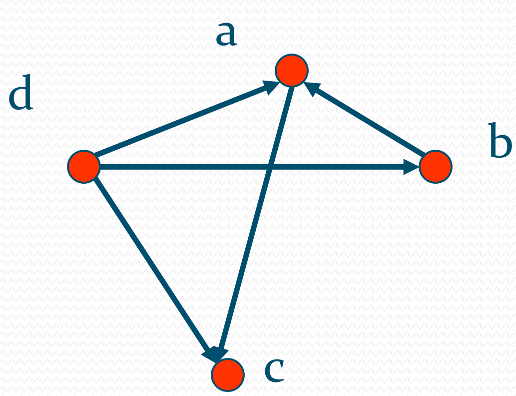
- Let G be a simple graph with V = {a,b,c,d,e} and E = {{a,a},{a,b},{a,c},{b,c},{c,d}}.
  - Is H = (V_H,E_H) with V_H={a,b,c,d} and E_H={{a,c},{b,c},{c,d}} a subgraph of G?
  - If so, find a second subgraph L such that H ∪ L = G

Section 10.3

Representing Graphs, Walks, Paths and Circuits, Connectedness

Representing Graphs





Vertex	Adjacent Vertices
a	b, c, d
b	a, d
c	a, d
d	a, b, c

Initial Vertex	Terminal Vertices
a	c
b	a
c	
d	a, b, c

Representing Graphs

- **Definition:** Let $G = (V, E)$ be a simple graph with $|V| = n$. Suppose that the vertices of G are listed in arbitrary order as $v_1, v_2, \ldots, v_n$.
- The **adjacency matrix** A (or A_G) of G, with respect to this listing of the vertices, is the $n \times n$ zero-one matrix with 1 as its (i, j)th entry when v_i and v_j are adjacent, and 0 otherwise.
- In other words, for an adjacency matrix $A = [a_{ij}]$,
- $a_{ij} = 1$ if $\{v_i, v_j\}$ is an edge of G,

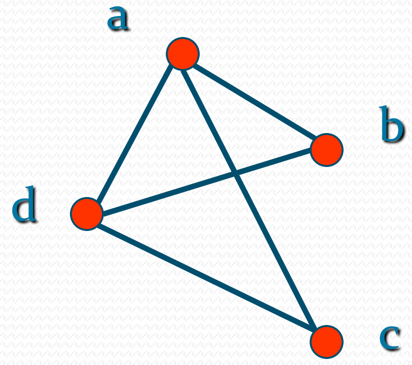
$a_{ij} = 0$ otherwise.

Representing Graphs

- **Example:** What is the adjacency matrix A_G for the following graph G based on the order of vertices a, b, c, d ?

Solution:

$$A_G = \begin{bmatrix} 0 & 1 & 1 & 1 \\ 1 & 0 & 0 & 1 \\ 1 & 0 & 0 & 1 \\ 1 & 1 & 1 & 0 \end{bmatrix}$$



Note: Adjacency matrices of undirected graphs are always symmetric.

Graph Walks

- **Definition:** A **walk** of length n from u to v, where n is a positive integer, in an **undirected graph** is a sequence of edges $e_1, e_2, \ldots, e_n$ of the graph such that $e_1 = \{x_0, x_1\}, e_2 = \{x_1, x_2\}, \ldots, e_n = \{x_{n-1}, x_n\}$, where $x_0 = u$ and $x_n = v$.
- When the graph is simple, we denote this path by its **vertex sequence** $x_0, x_1, \ldots, x_n$, since it uniquely determines the path.
- The path or circuit is said to **pass through** or traverse $x_1, x_2, \ldots, x_{n-1}$.

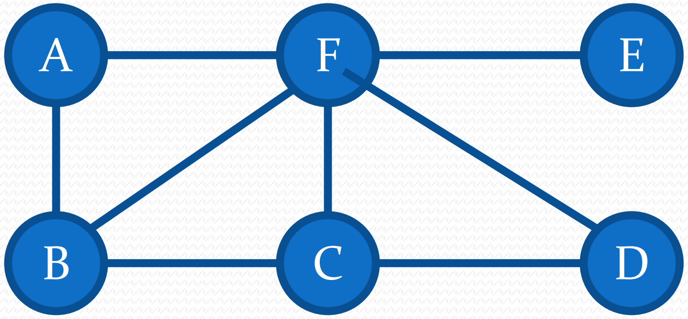
Paths and Circuits

- A **trivial walk** from **v** to **v** consists of the single vertex **v**, and no edges.
- The path is a **closed-walk** if it begins and ends at the same vertex, that is, if u = v.
- The **length** of a walk is the number of edges in the walk
- A **trail** is a walk with no repeated edges
- A **path** is a walk with no repeated vertices
- A **circuit** is a closed walk with no repeated edges
- A path or circuit is **simple** if it has no repeated vertices (except the first and last in a circuit).
- A *non-trivial simple circuit* is called a **cycle**

Exercise 1

- In the given graph, determine whether each of the following is a path, simple path, circuit, or simple circuit

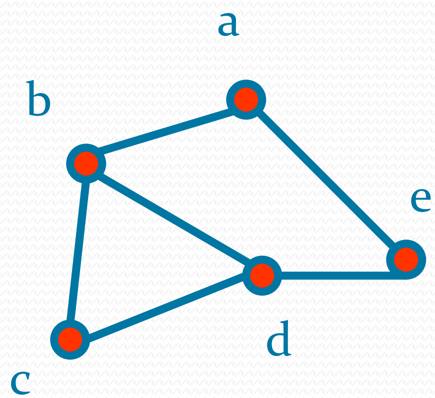
1. abcfb
2. abcf
3. fabfcdf
4. fabcdf
5. abfb
6. cfbbc
7. bb
8. e



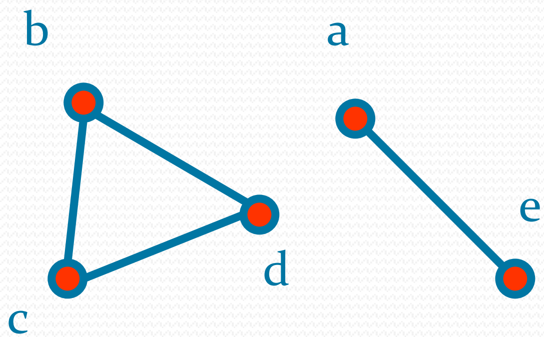
Connectivity

- Let us now look at something new:
- **Definition:** An undirected graph is called **connected** if there is a walk (or a path, or a simple path) between every pair of distinct vertices in the graph.
- For example, any two computers in a network can communicate if and only if the graph of this network is connected.
- **Note:** A graph consisting of only one vertex is always connected, because it does not contain any pair of distinct vertices.

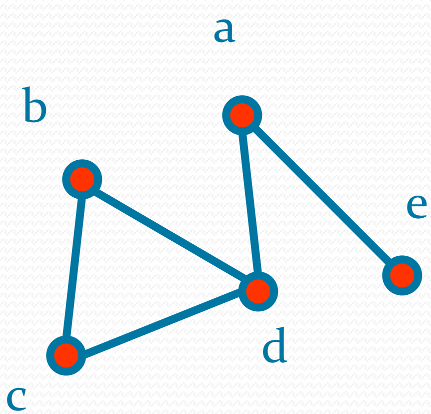
Exercise 2: Connected or not?



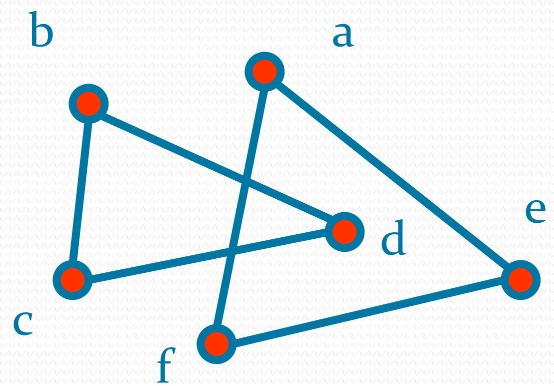
Yes.



No.



Yes.



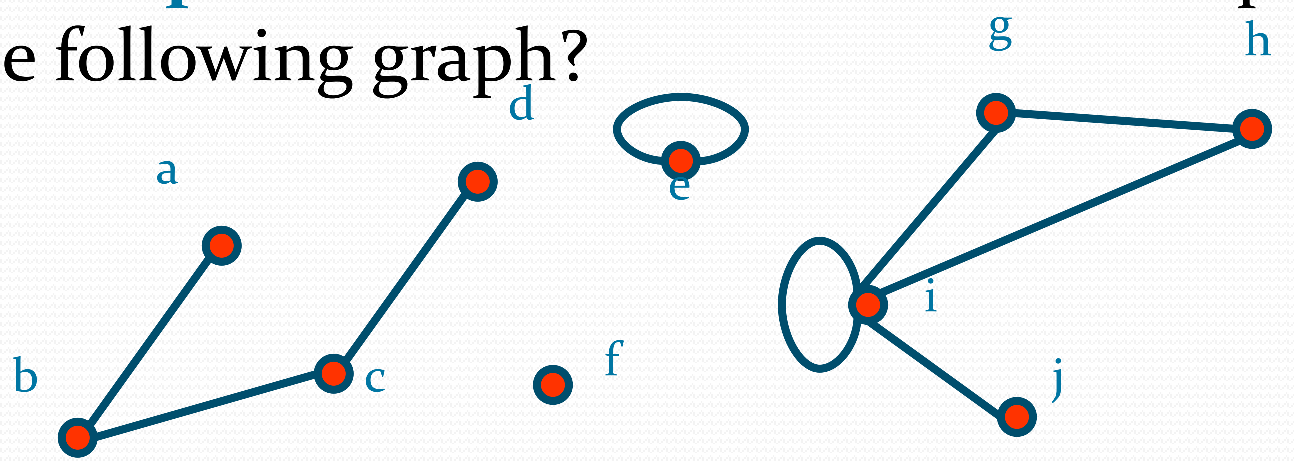
No.

Connectivity

- A graph that is not connected is the union of two or more connected subgraphs, each pair of which has no vertex in common.
- A subgraph H is a **connected component** of a graph G if:
  - H is connected
  - H is not a proper subgraph of any connected subgraph of G
- It follows that a graph is connected $\Leftrightarrow$ G has only one connected component

Exercise 3

- **Example:** What are the connected components in the following graph?



Solution: The connected components are the graphs with vertices {a, b, c, d}, {e}, {f}, {g, h, i, j}.

Exercise 4

- What is the minimum number of edges possible in a connected graph on 4 vertices?

Section 10.5

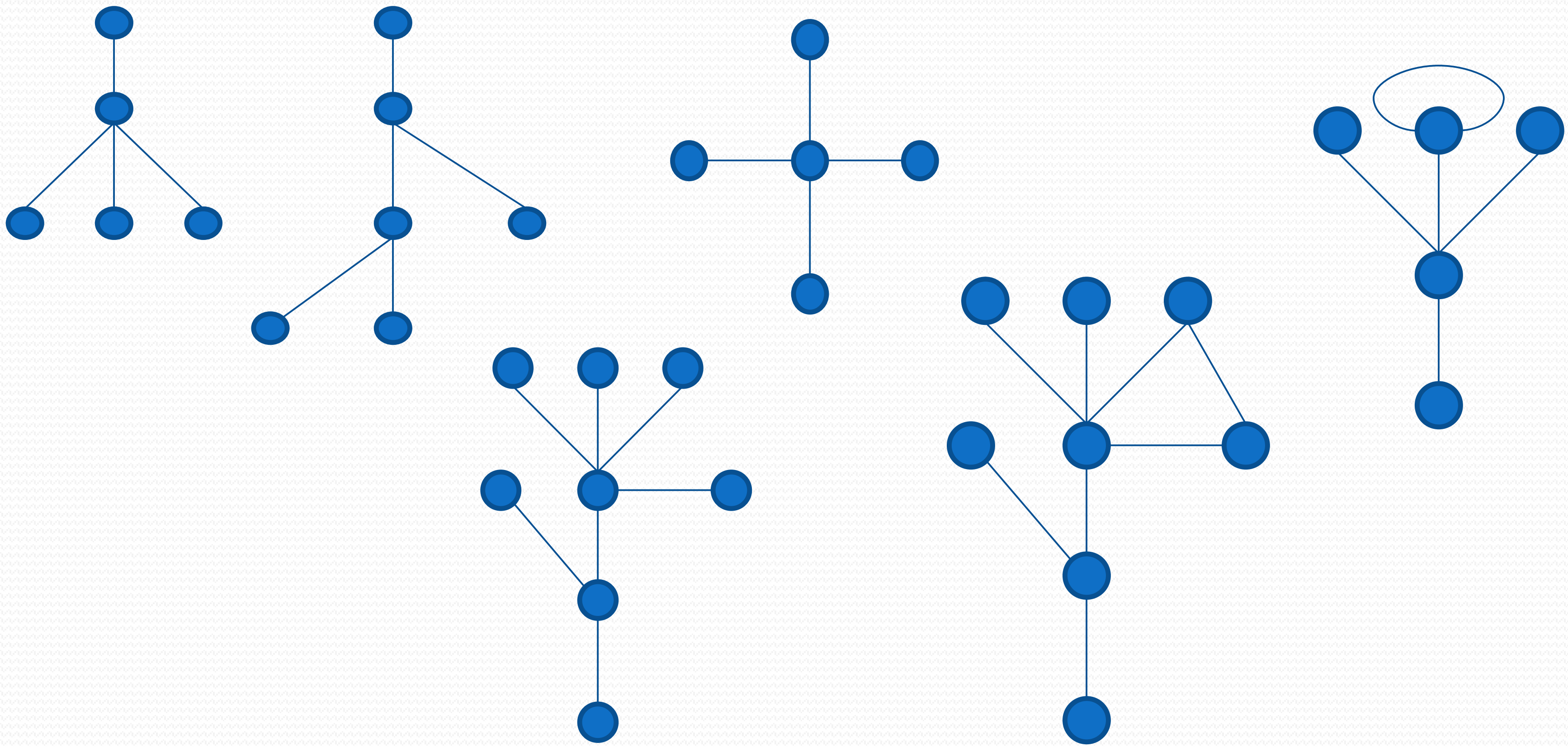
Trees

What is a Tree?

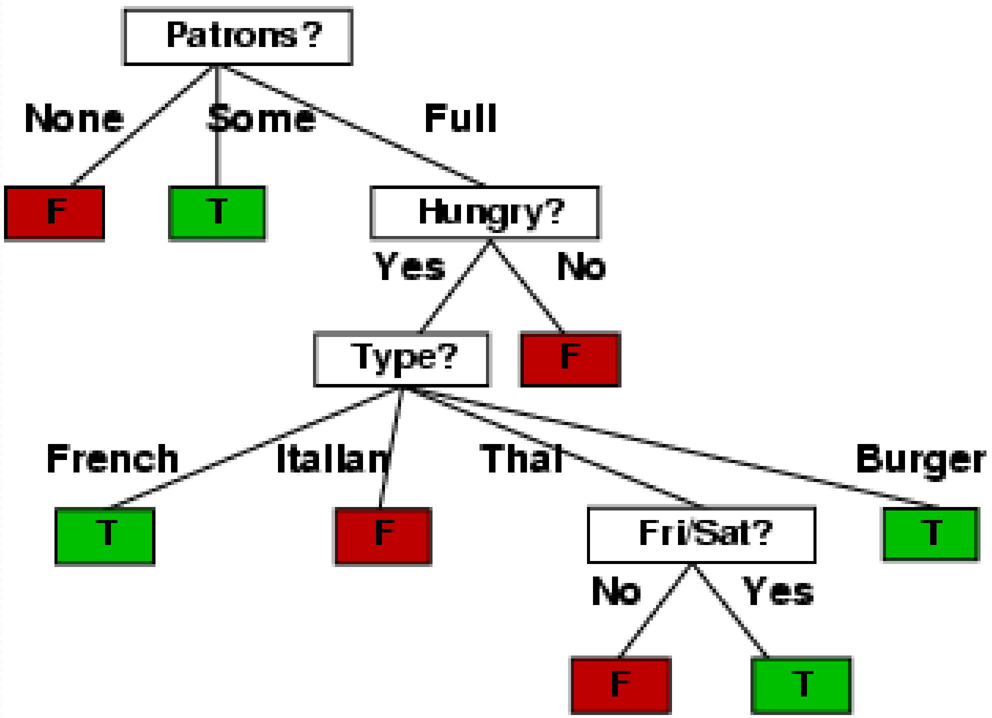
- A graph is called **acyclic** if it has no non-trivial circuits
  - A **tree** is an acyclic, connected graph
  - A **trivial tree** is a graph with a single vertex
  - A **forest** is an acyclic, disconnected graph
- A tree is a special kind of **simple graph**

Exercise 1

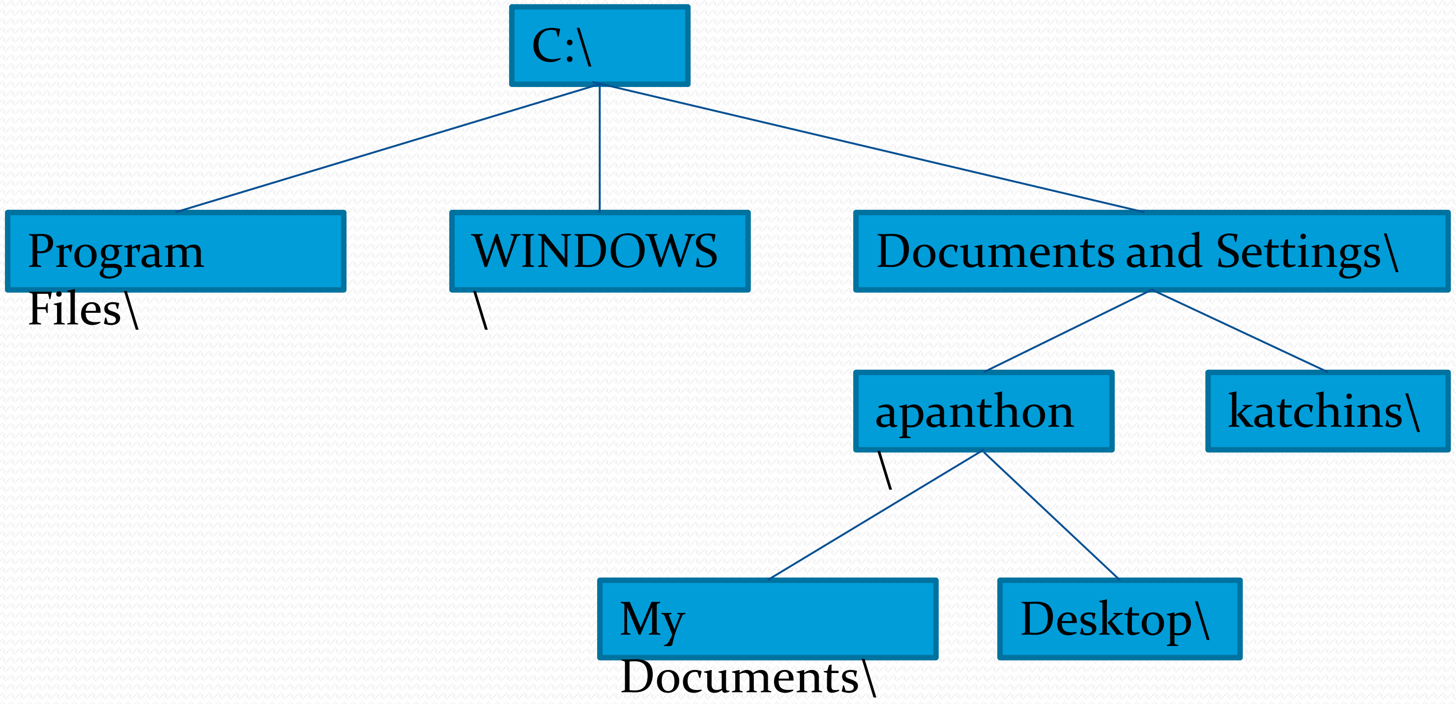
- Which of the following is a tree?



Tree Example—Decision Tree

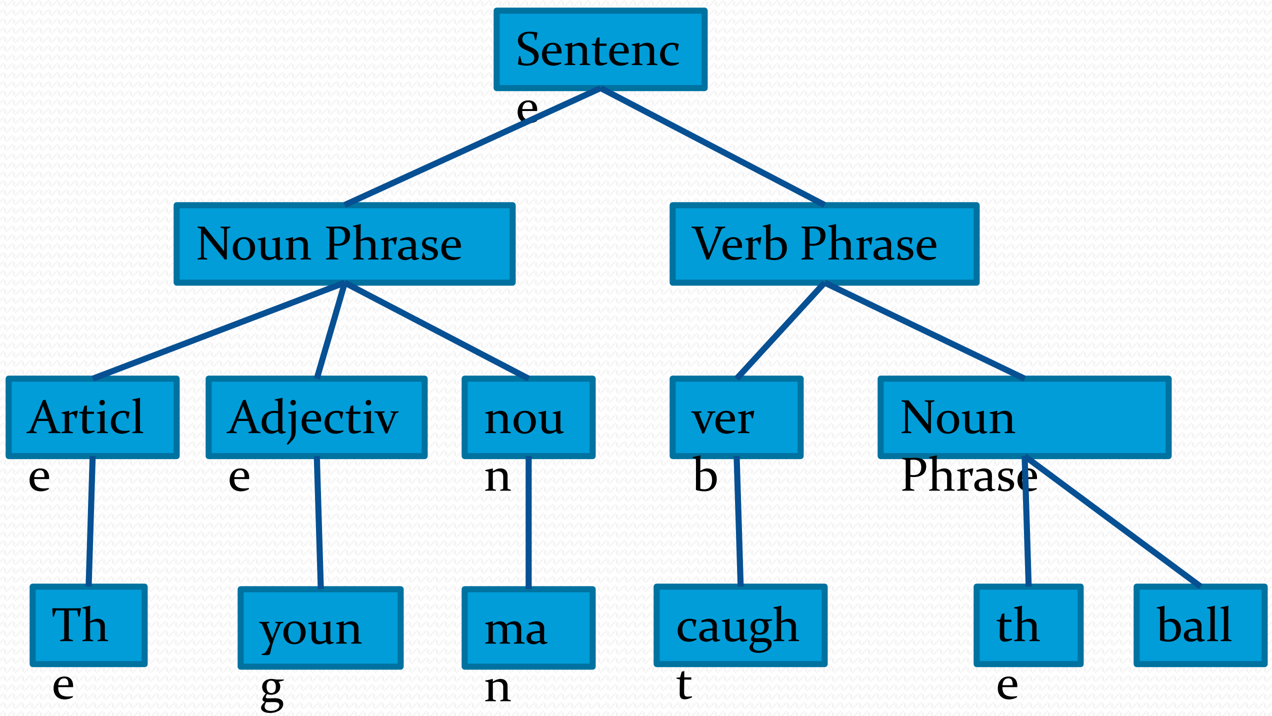


Tree Example: Directory Structures



Tree Example: Parse Tree

- “The young man caught the ball”



Every Graph Contains A Tree

- Let G be any graph
  - If G has no cycles, then it is a tree!
  - If G does have cycles, for each cycle
    - Remove one edge from the cycle
      - G will still be connected. Why?
  - The resulting graph G' will be an acyclic tree

Tree Properties

- Let T = {V,E} be a graph. The following are equivalent:
  a) T is connected and removing any edge from T disconnects T into two subgraphs that are trees (subtrees)
  b) There is a unique path between any two distinct vertices v and w in T
  c) T is a tree

Exercise 3

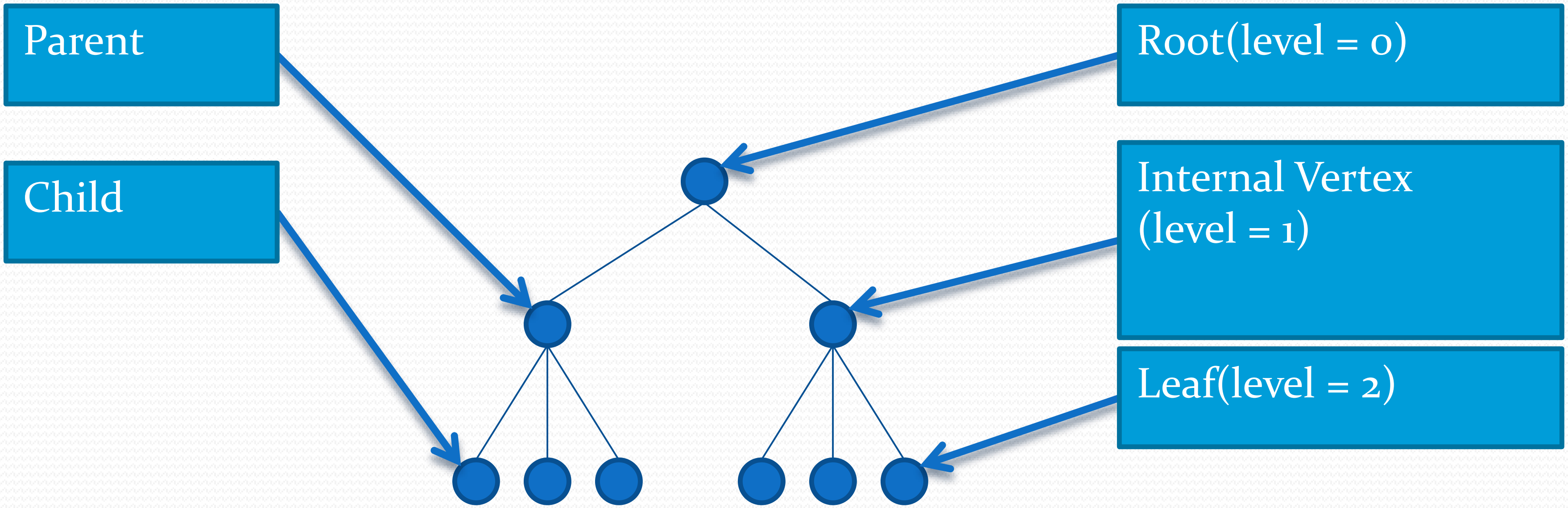
- Is a graph with 12 vertices and 12 edges a tree?
- Is ***any*** graph with 5 vertices and 4 edges a tree?
- Is any connected graph with 5 vertices and 4 edges a tree?

Exercise 4

- Let T be a graph on n vertices. Prove that the following are all equivalent:
  a) T is a tree
  b) T is connected and has $n - 1$ edges
  c) T is acyclic and has $n - 1$ edges

Rooted Trees and Tree Traversals

What is a rooted tree?



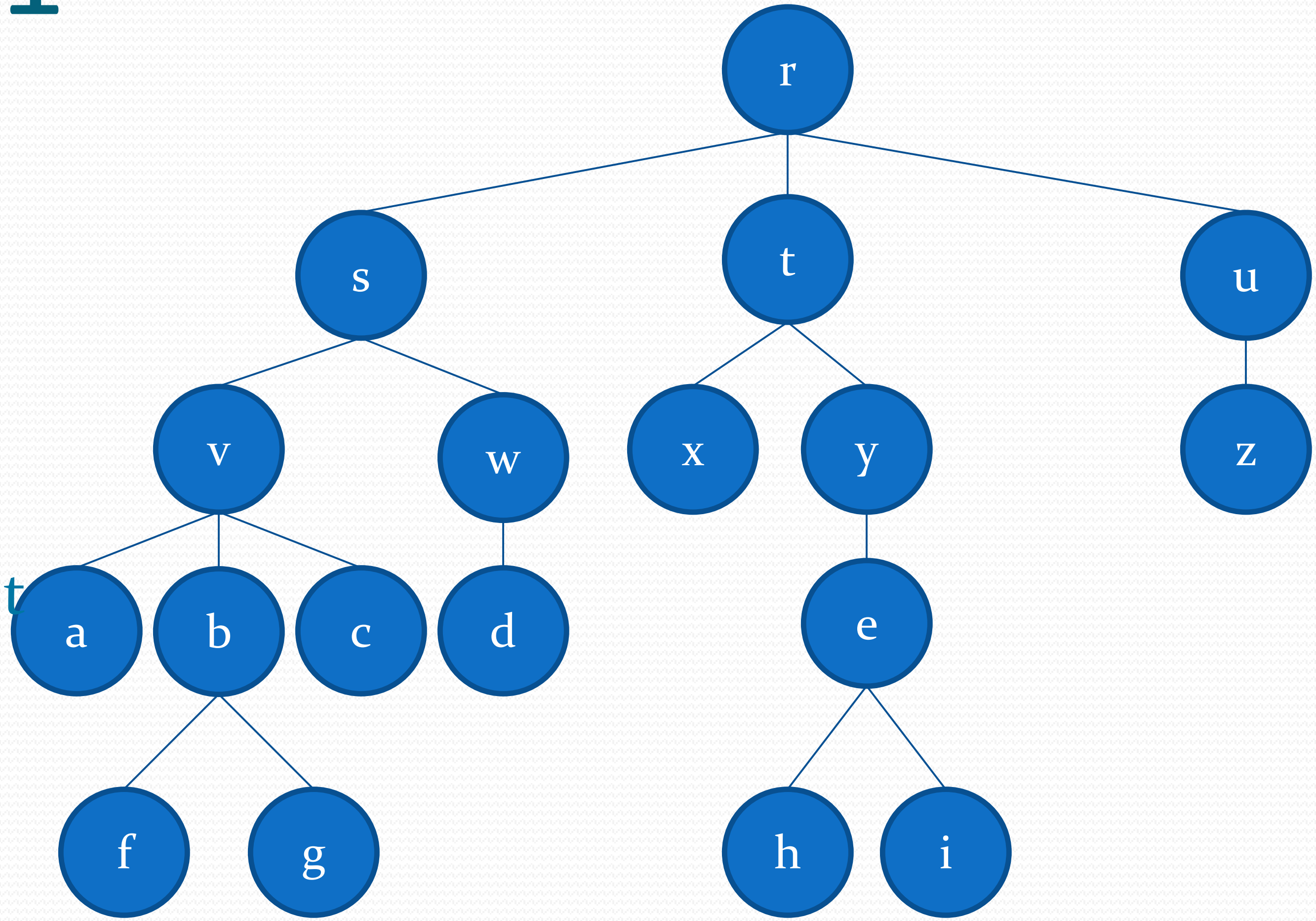
Tree Definitions

- The root is any vertex in a tree that is selected to be the root
- Level(v) = #edges it takes to reach v from the root
- height(T) = the maximum level of any vertices in T
- Children(v) → all vertices adjacent to v whose level is Level(v) + 1
- Parent(v) →The ***unique*** vertex that is adjacent to v whose level is Level(v) – 1
- A ***leaf*** is a node with no children.
- An ***ancestor*** v is any vertex w that lies on the path from the root to v. v would be considered a ***descendant*** of all such vertices.

Exercise 1

Find the:

a) Level of **e**
b) Height of the tree
c) Children of t
d) Parent of t
e) Ancestors of g
f) Descendants of t
g) Leaves of the tree



Exercise 2

- Design a tree to represent the table of contents of a book:

C1

S1.1

S1.2

S1.3

C2

S2.1

S2.2

S2.2.1

S2.2.2

C3

S3.1

S3.2

Binary Trees

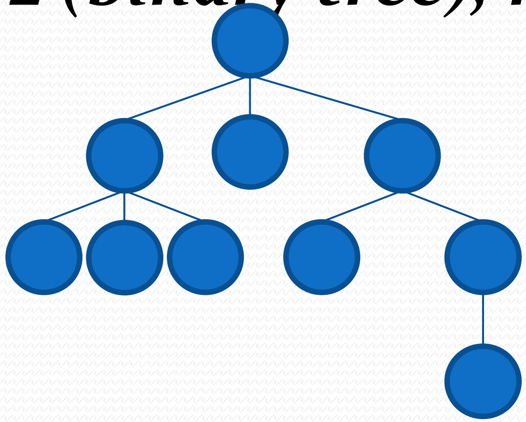
- A ***Binary Tree*** is a rooted tree in which each internal vertex has at most ***two children***
- Since there are only two, they get special names:
  - ***Left child***
  - ***Right child***
  - If there is only one, call it the left child
- Subtrees get special names too!
  - The ***left subtree*** of a vertex v is the subtree rooted at the left child of v
  - The right subtree of a vertex v is the subtree rooted at the right child of v
- A ***full (or complete) binary tree*** is a binary tree in which each internal vertex has exactly two children

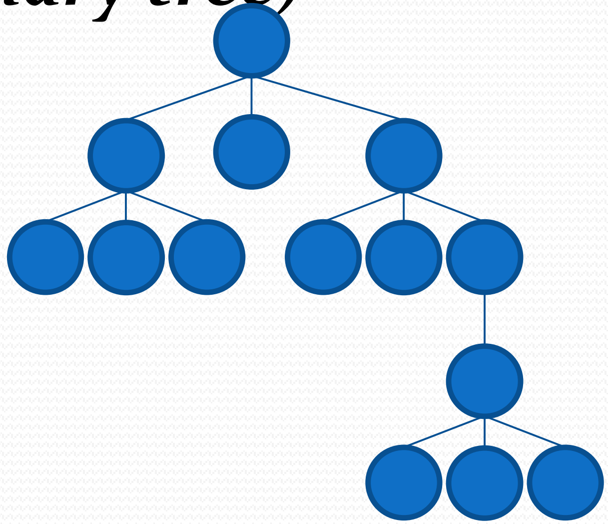
Exercise 3

- Use a complete binary tree to represent the following mathematical expressions:
  - a + b
  - (a + b) ÷ ((c * d) – e)

M-ary Trees

- An m-ary tree is a rooted tree in which each internal vertex has ***at most m children***
- A ***full m-ary tree*** is an m-ary tree in which each internal vertex has ***exactly m children***
- ***m = 2 (binary tree), m = 3 (ternary tree)***





Theorem 1

- Let T be a full m-ary tree with ***n*** vertices, ***i*** internal vertices, and ***L*** leaves. Then each of the following is true:

$$\text{a) } n = mi + 1$$

$$\text{b) } L = i(m-1) + 1$$

$$\text{c) } i = \frac{L-1}{m-1} = \frac{n-1}{m}$$

Corollary to Theorem 1

- Let T be a full binary tree with ***n*** vertices, ***i*** internal vertices, and ***L*** leaves. Then each of the following is true:

$$\text{a) } n = 2i + 1$$

$$\text{b) } L = i + 1 = \frac{n+1}{2}$$

$$\text{c) } i = L - 1 = \frac{n-1}{2}$$

Exercise 4

- How many vertices does a full ternary tree with 11 leaves have?
- Is there a full binary tree with 12 vertices?
- How many edges does a full 5-ary tree with 100 internal vertices have?
- Is there a full binary tree that has 10 internal vertices and 13 leaves?

Exercise 5

- The Wimbledon tennis championship is a single-elimination tournament in which a player is eliminated after a single loss. If 31 women compete in the championship, how many matches must be played to determine the champion?

Exercise 6

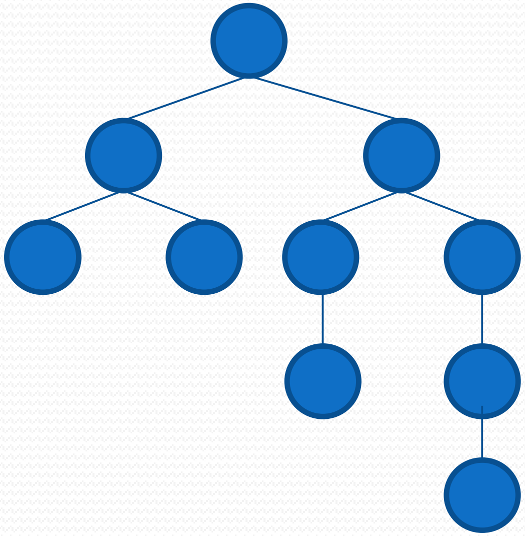
- Suppose someone starts a chain letter. Each person who receives the letter is asked to send it to 5 other people.
  - If everyone who receives the letter follows the instructions, how many people can be reached in a tree of height 2?
  - If 125 people received the letter, but did not send it, determine the following:
    - How many people sent the letter?
    - How many people in total have seen the letter, including the person who started it?

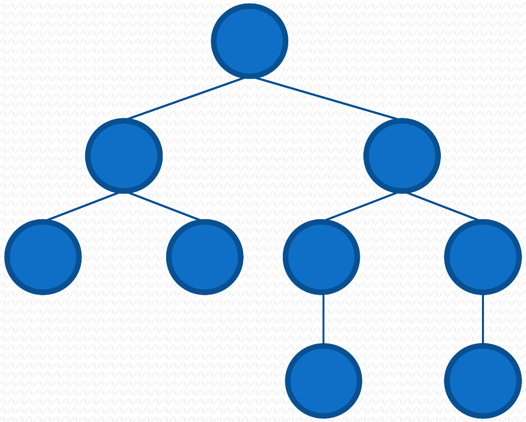
Exercise 7

- A computer lab has a single wall socket with 6 outlets in it. Using power strips with 6 connections each, how many extension cords do we need to power 46 all-in-one computers?

Balanced Trees

- An m-ary tree of height ***h*** is balanced if every leaf is at level ***h*** or ***h – 1.***
- Which of the following trees are balanced?





Binary Tree Traversals

Tree Traversals

- In mathematics, we’re always studying tree properties, and they are useful
- In computing, we’re not just studying trees, we are ***storing and processing trees***
- A tree traversal is a method for ***efficiently retrieving information from a tree***
- There are three different traversals that have different applications:
  - Pre-order
  - In-order
  - Post-order

Preorder Traversal of a Binary Tree

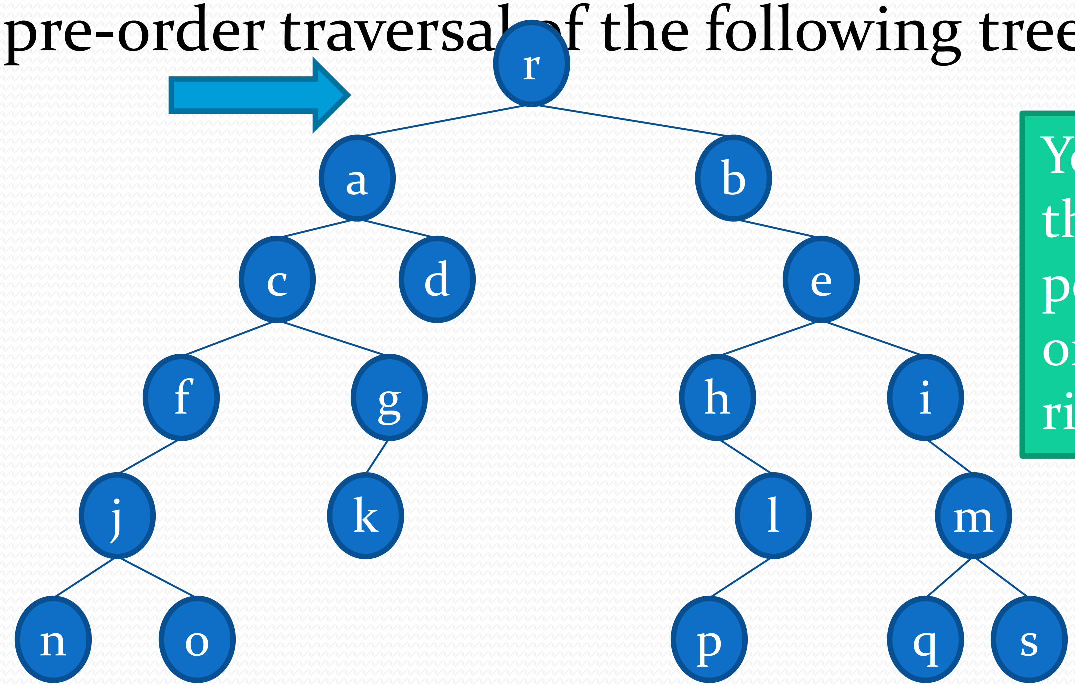
- Let T be a rooted binary tree with root R, and left subtree T_L and right subtree T_R.
- The Preorder traversal of T is as follows:
  - ‘Visit’ R (print value, perform computation, etc.)
  - Perform a preorder traversal on T_L
  - Perform a preorder traversal on T_R

About Recursion

- Recursion is weird…but really cool!
- The pre-order procedure is very easy to define because it ***uses itself as part of the definition!***
  - 'Visit' R (print value, perform computation, etc.)
  - Perform a preorder traversal on T_L
  - Perform a preorder traversal on T_R
- If we follow this precisely, then it will gradually take us throughout the entire tree!

Exercise 1

- Find the pre-order traversal of the following tree (ROOT-L-R):

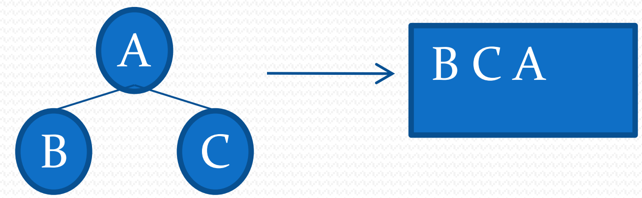


Your Turn! Finish the traversal by performing a pre-order traversal of the right subtree.

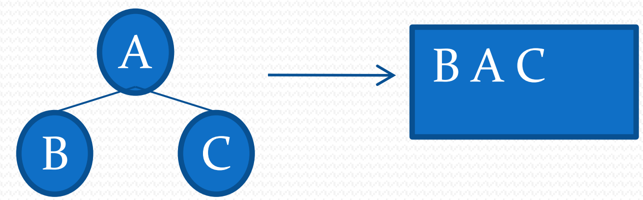
Result:

Post-Order and In-Order Traversals

- As before, let T be a rooted binary tree with root R, and left subtree T_L and right subtree T_R.
- The post-order traversal of T is:
  - Perform a post-order traversal of T_L
  - Perform a post-order traversal of T_R
  - ‘Visit’ R

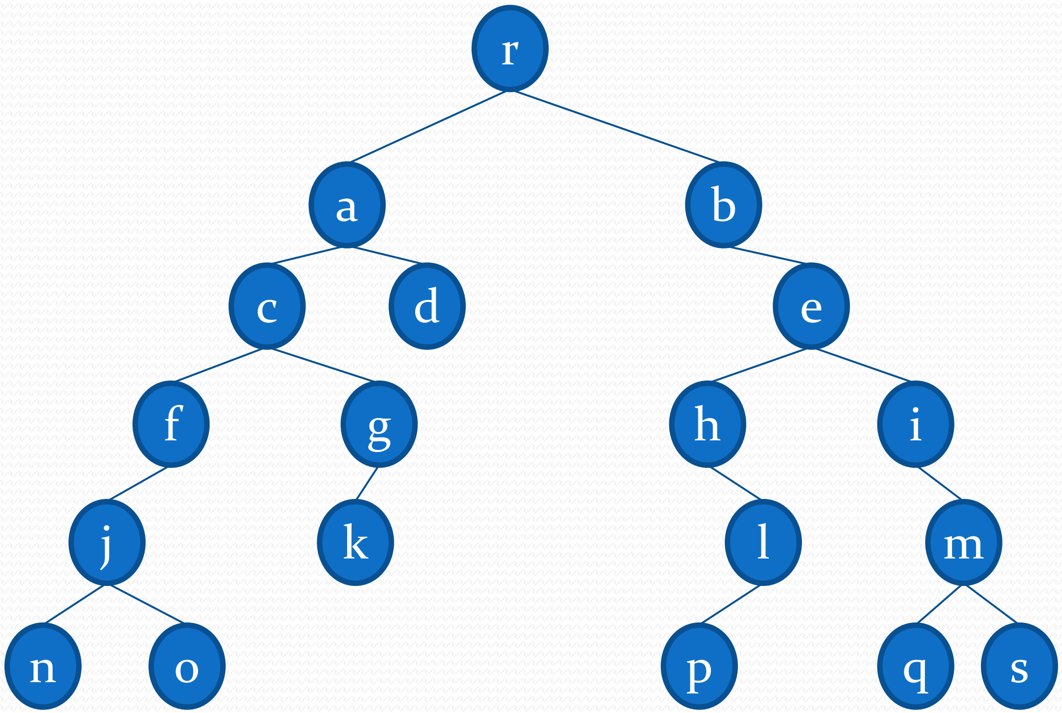


- The in-order traversal of T is:
  - Perform a post-order traversal of T_L
  - ‘Visit’ R
  - Perform a post-order traversal of T_R



Exercise 2

- Find the Post-Order (L-R-Root) traversal:



Exercise 3

- Find the in-order (L-Root-R) traversal: