Introduction to Trees

By

Dr. Yogesh M. Rajput

Assistant Professor

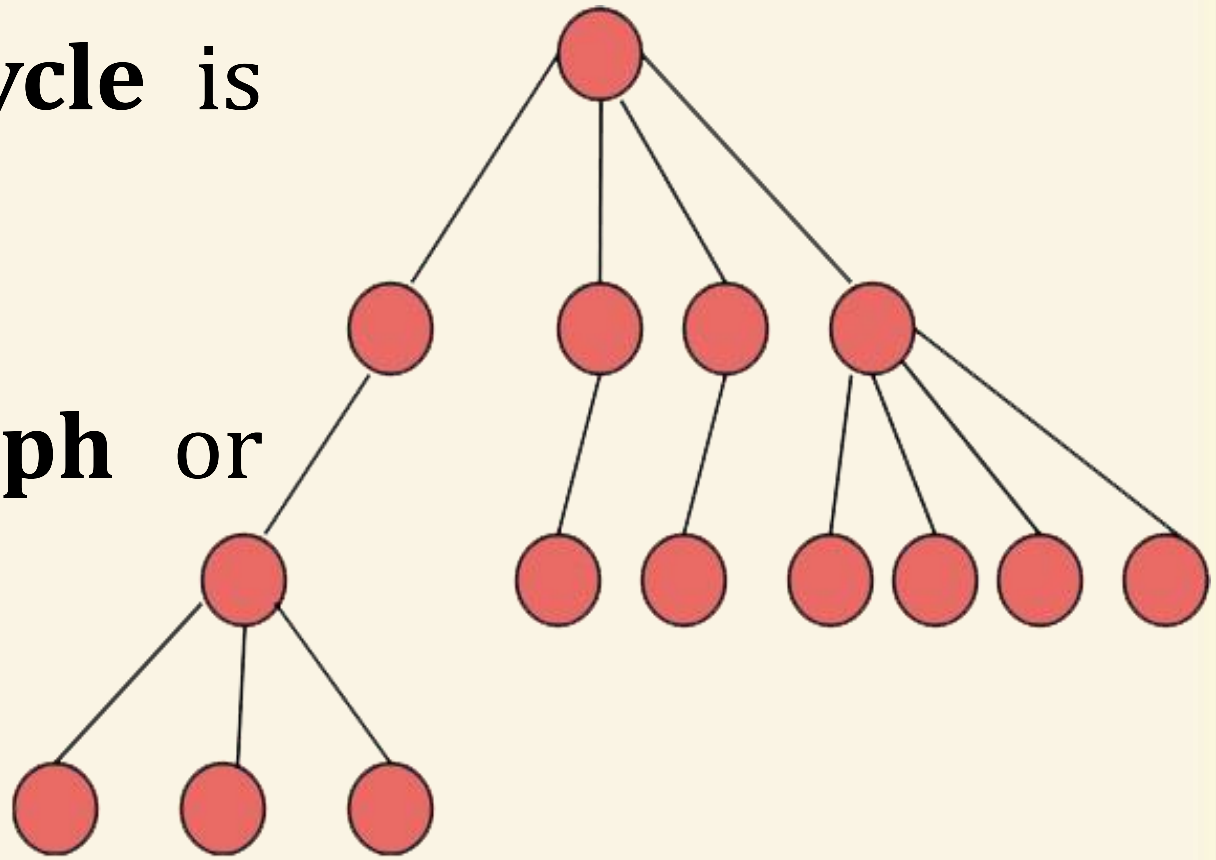
Dr. G. Y. Pathrikar College of Computer Science and Information Technology,

MGM University, Aurangabad.

Introduction to Trees

Introduction to Trees

- A **graph** which has **no cycle** is called an **acyclic graph**.
- A **tree** is an **acyclic graph** or **graph** having **no cycles**.

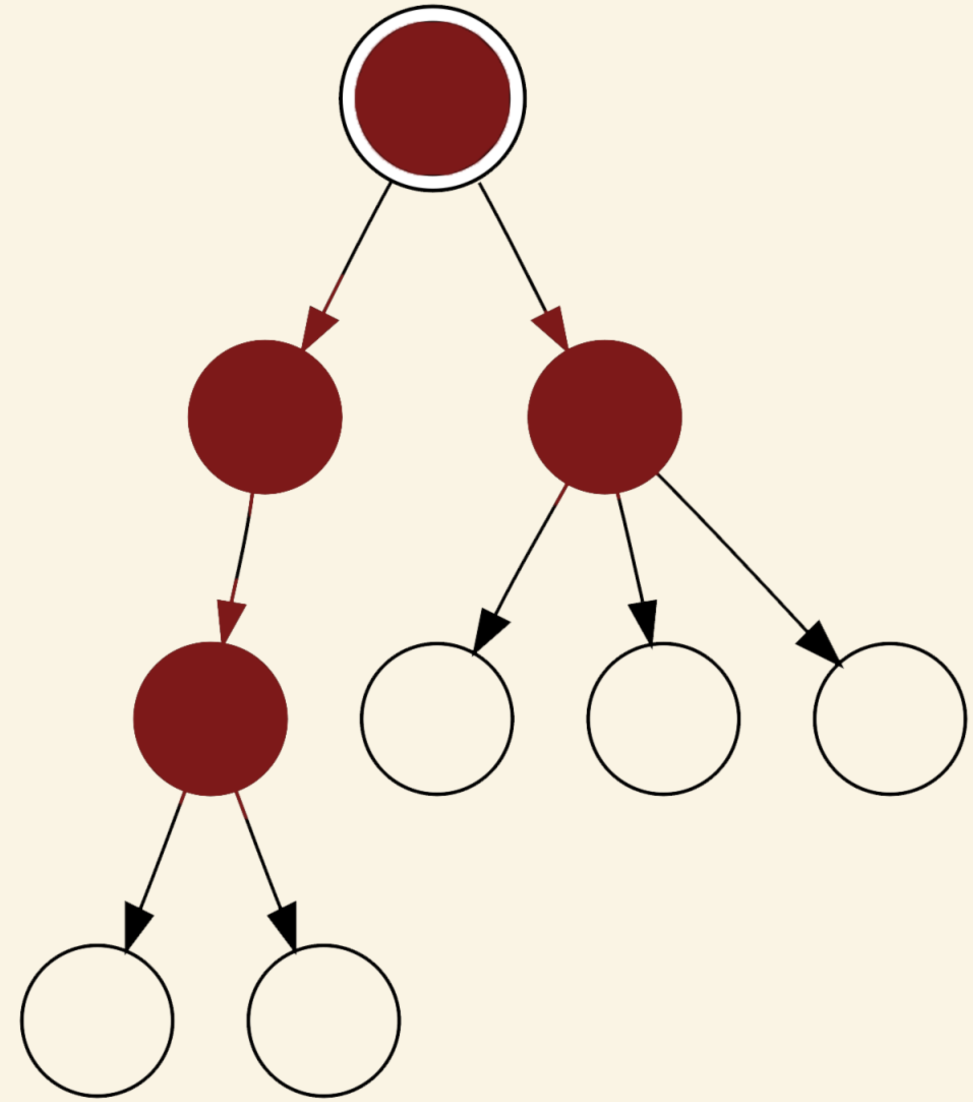


Introduction to Trees

- A **tree** or **general trees** is defined as a **non-empty finite set** of elements called **vertices** or **nodes** having the property that each **node** can have minimum **degree 1** and maximum **degree n**.

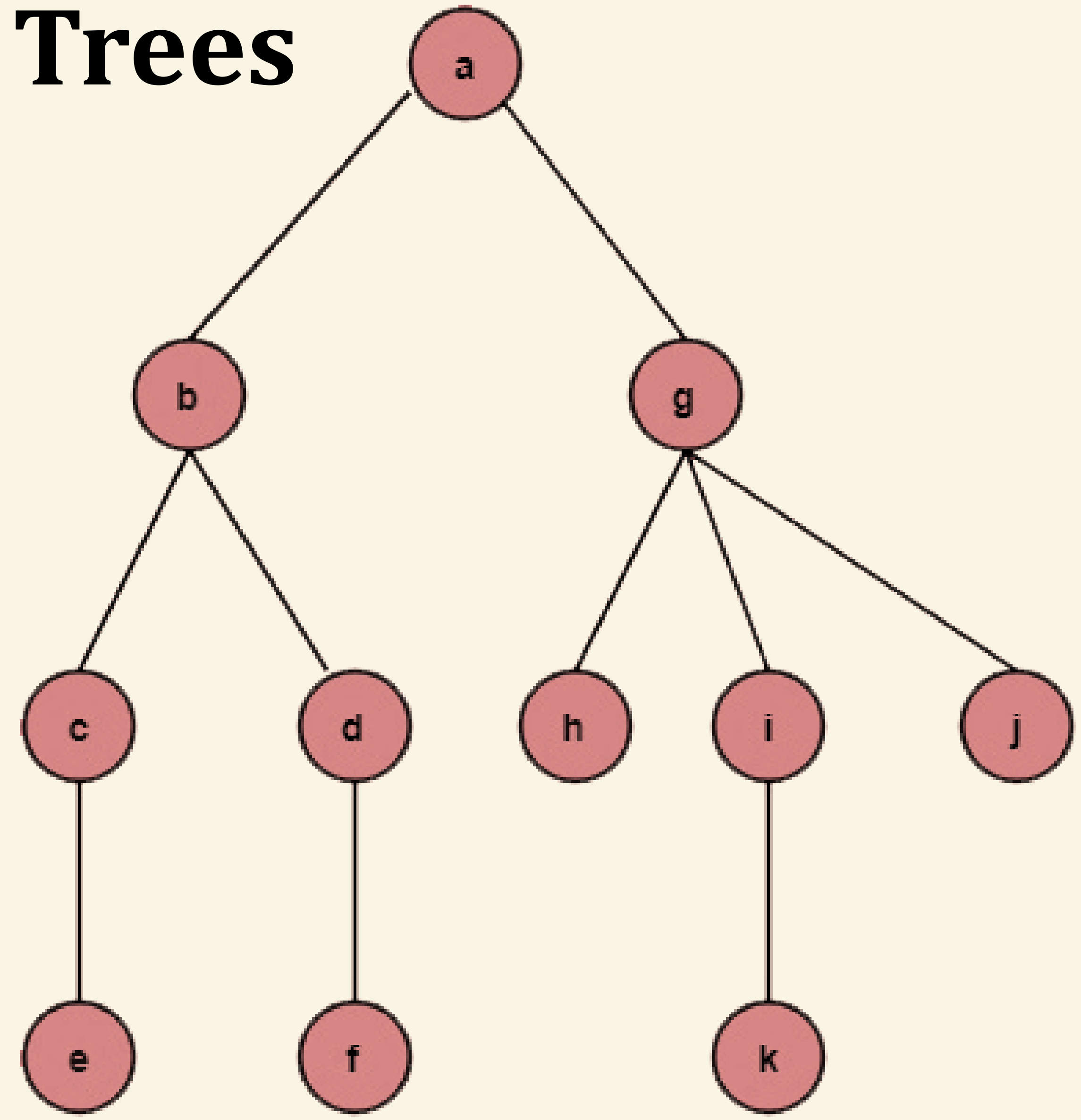
Directed Trees

- A **directed tree** is an **acyclic directed graph**.



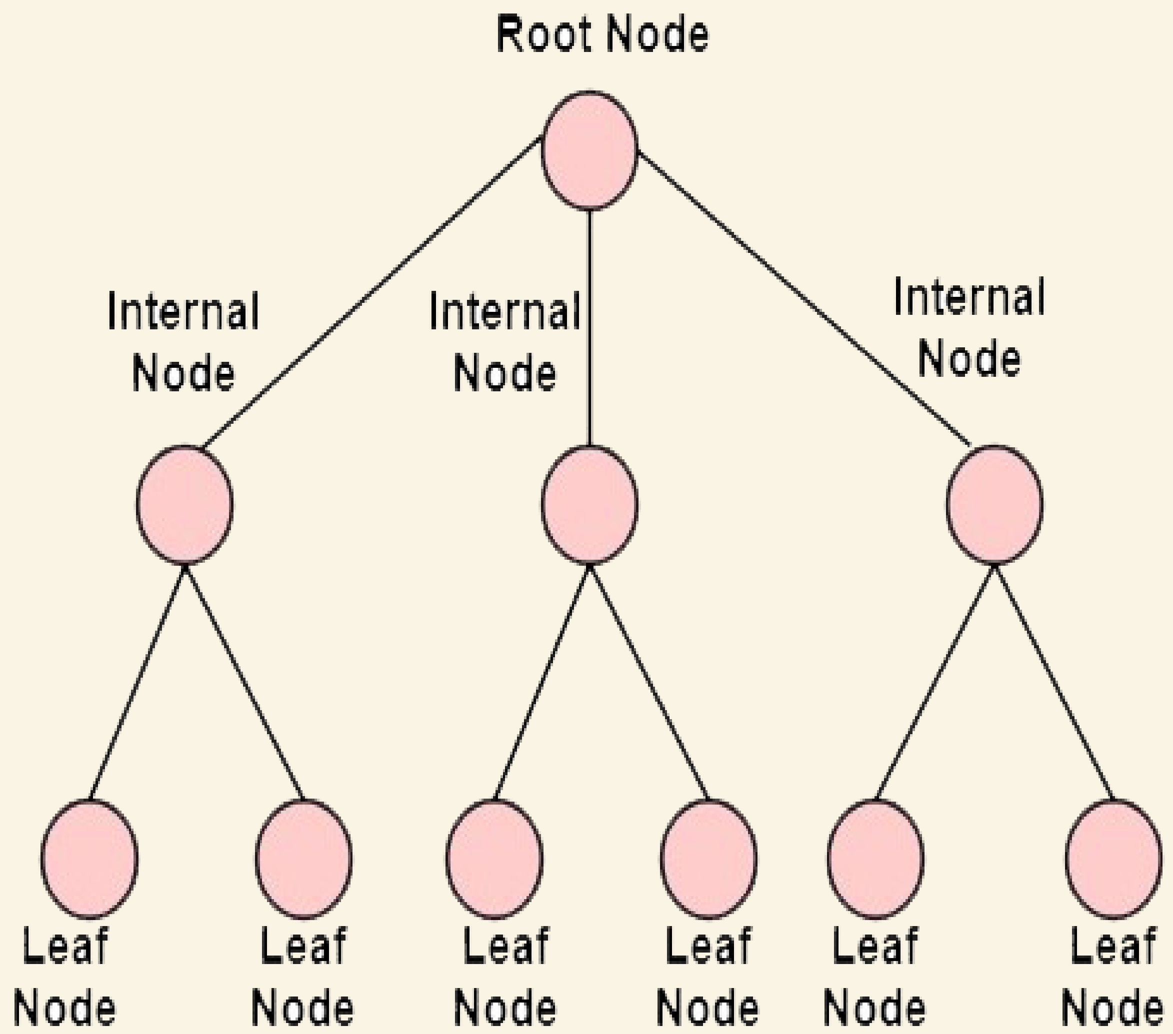
Ordered Trees

- If in a tree at **each level**, an **ordering** is **defined**, then such a **tree** is called an **ordered tree**.



Rooted Trees

- A **rooted tree G** is a connected **acyclic** graph with a special node that is called the **root** of the tree and every edge directly or indirectly **originates** from the **root**.



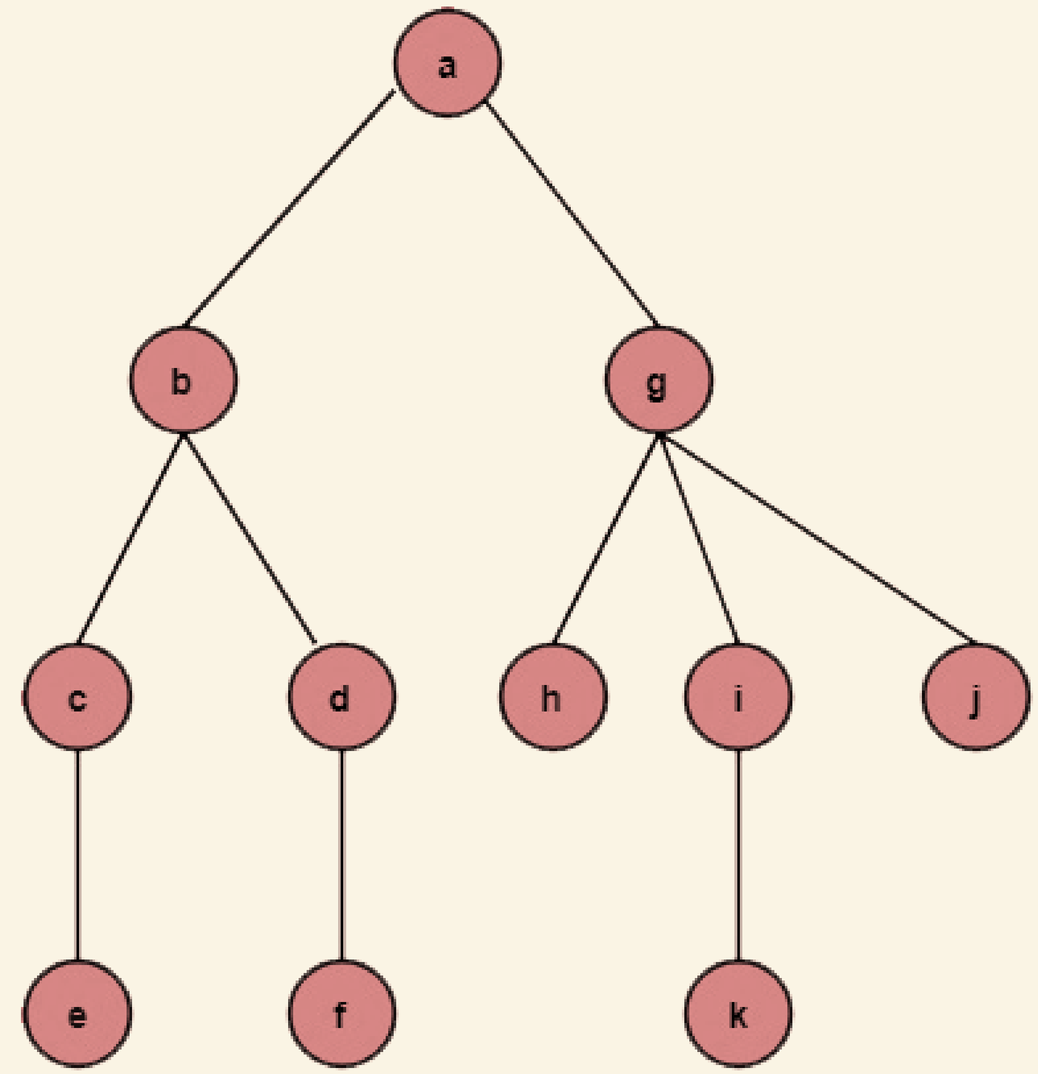
Properties of Trees

1. There is only **one path** between each pair of vertices of a **tree**.
2. If a graph G there is **one and only one path** between each pair of vertices G is a **tree**.
3. A **tree T** with n vertices has **n-1 edges**.
4. A **graph** is a tree if and only if it a **minimal connected**.

Pendant Vertices In A Tree

Pendant Vertices In A Tree

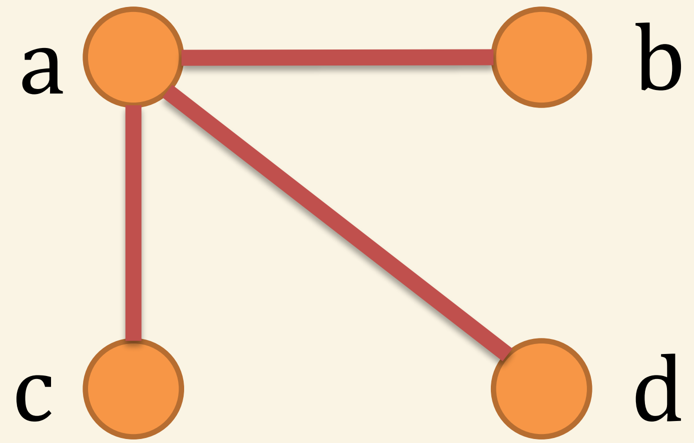
- A **pendant vertex** was defined as a vertex of **degree one**.
- The reason is that in a tree of **n** vertices we have **n-1** edges.



Question

Q1. Prove that any connected **graph G** with **n vertices** and (**n-1**) edges is a tree.

Vertices (n) = 4
Edges (n-1) =3



Question

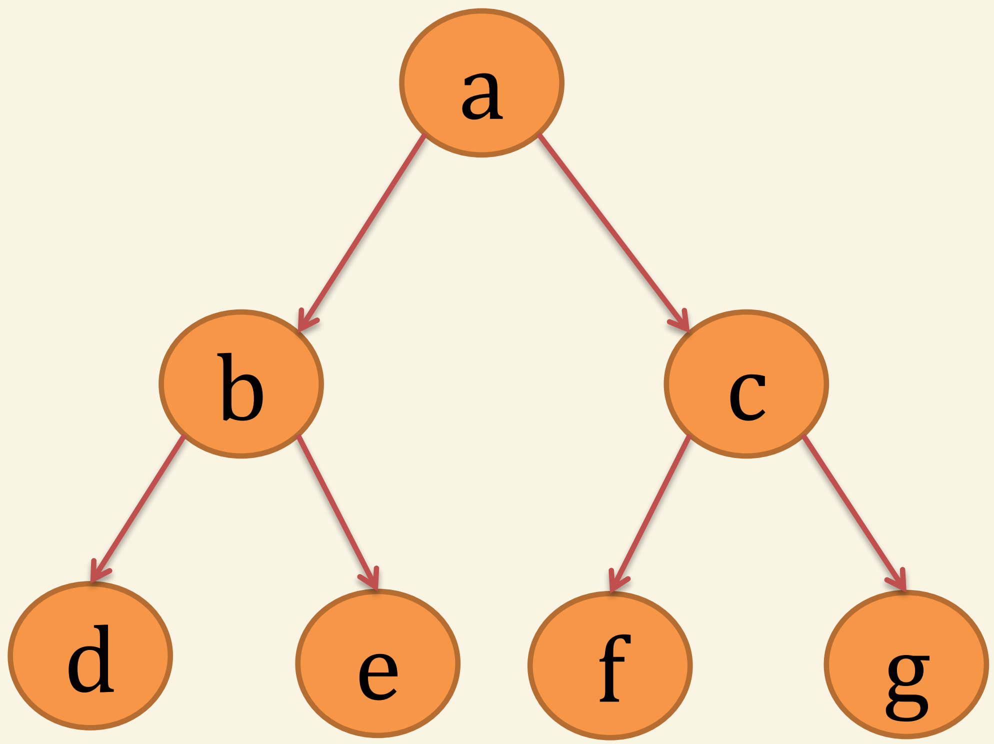
Q2. Prove that any connected **graph G** with **n vertices** and (**n-1**) edges is a tree.

Vertices (n) = 5
Edges (n-1) = 4

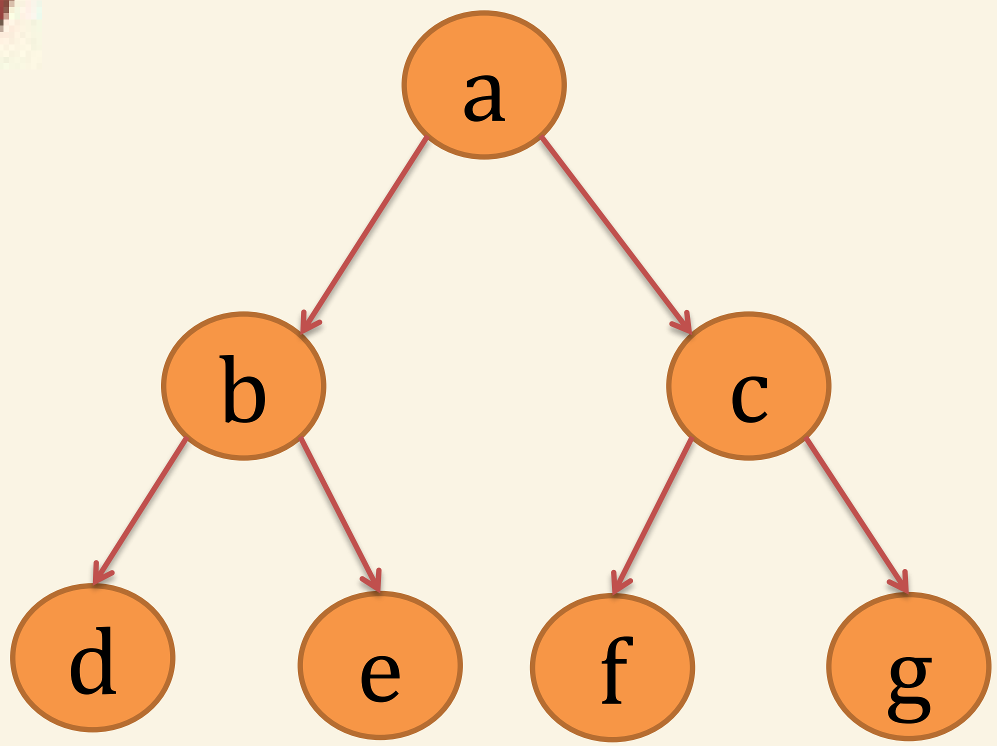
Binary Trees

Binary Trees

- If the **outdegree** of every node is less than **or equal to 2**, in a **directed tree** than the tree is called a binary tree.



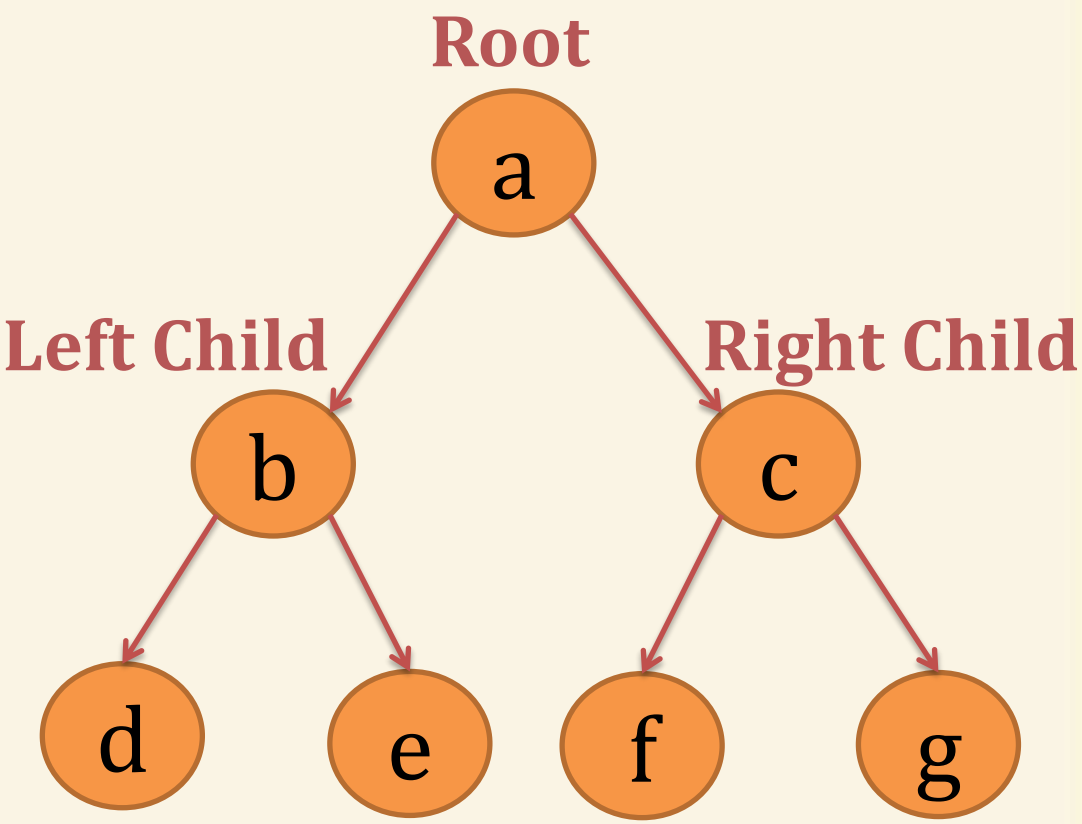
Binary Trees



Vertex	Indegree	Outdegree
a	0	2
b	0	2
c	0	2
d	0	0
e	0	0
f	0	0
g	0	0

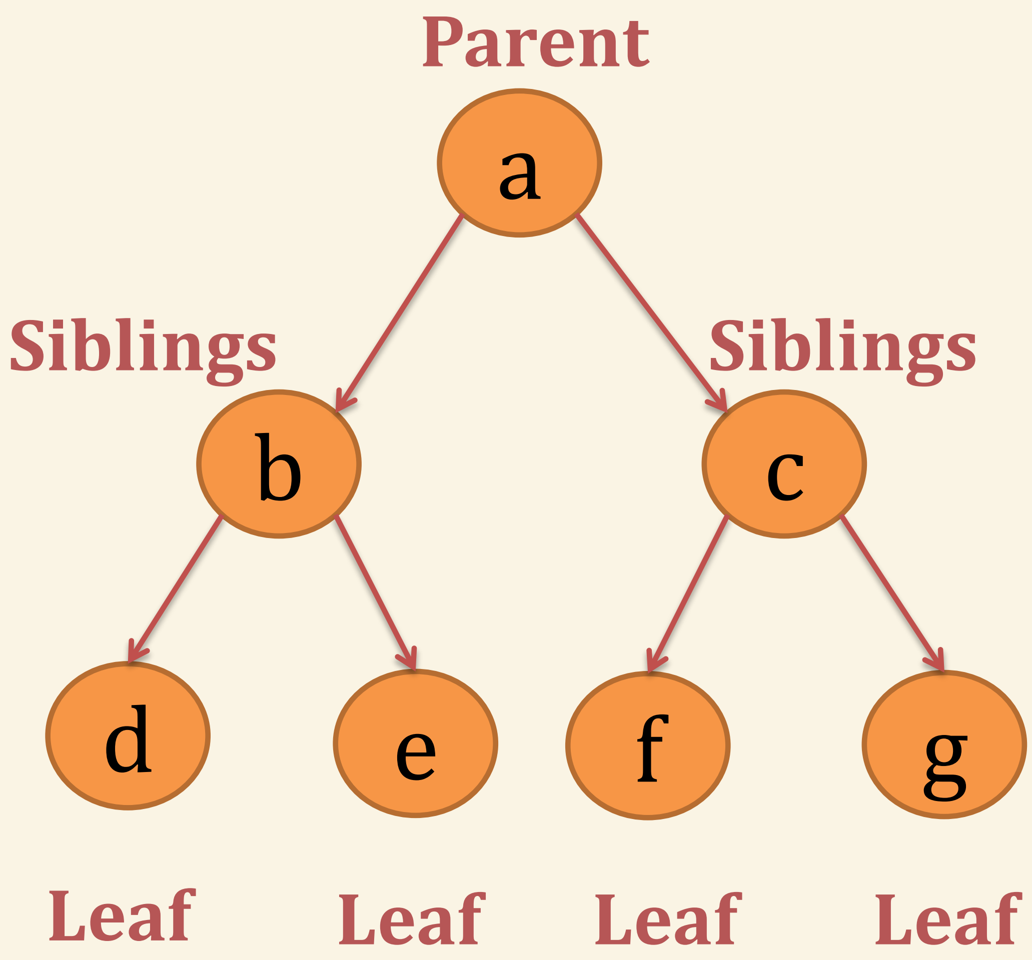
Basic Terminology

- **Root**: A binary tree has a unique node called the root of the tree.
- **Left Child**: The node to the left of the root is called its left child.
- **Right Child**: The node to the right of the root is called its right child.



Basic Terminology

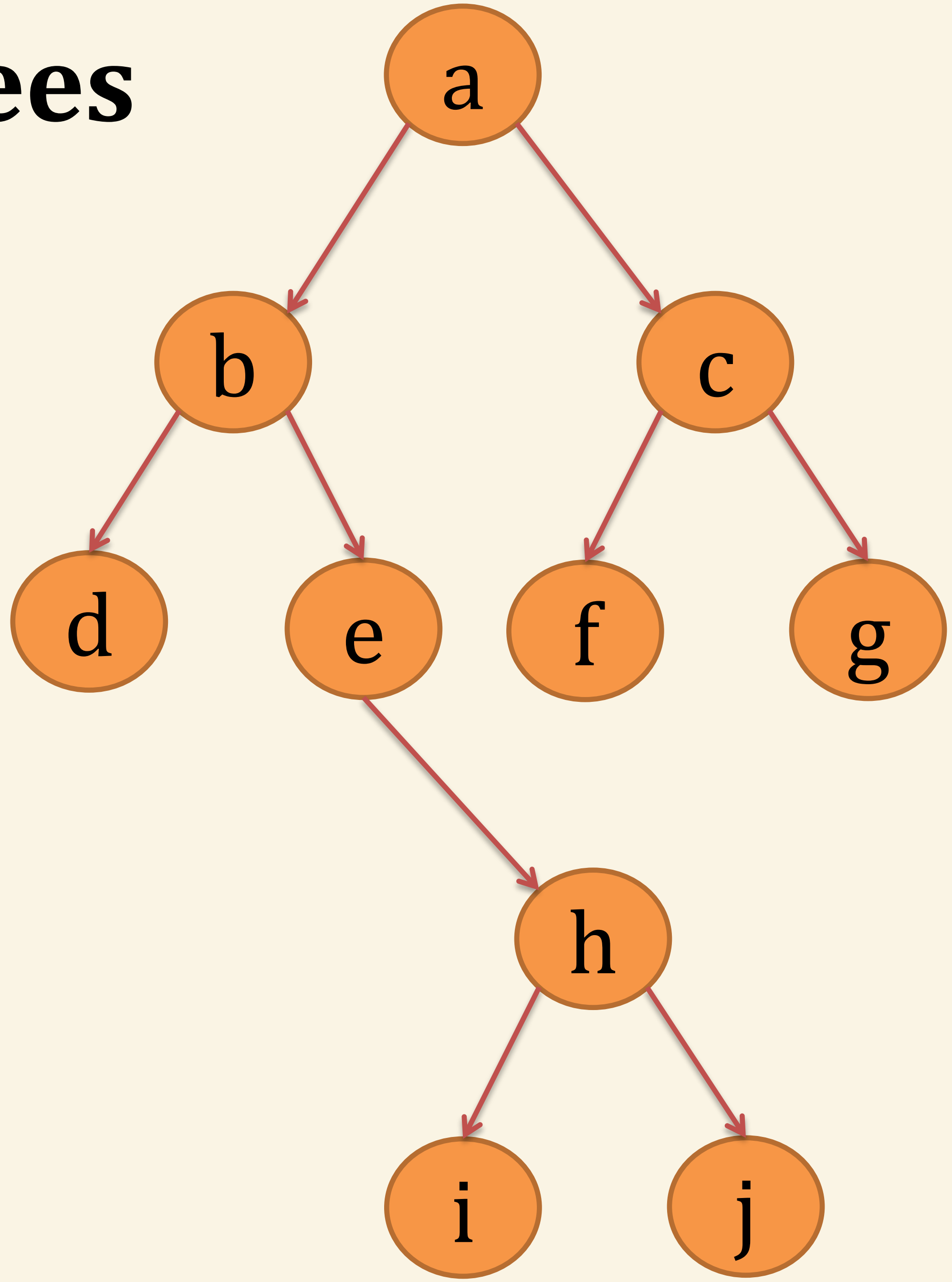
- **Parent**: A node having a left child or right child or both are called the parent of the nodes.
- **Siblings**: Two nodes having the same parent are called siblings.
- **Leaf**: A node with no children is called a leaf.



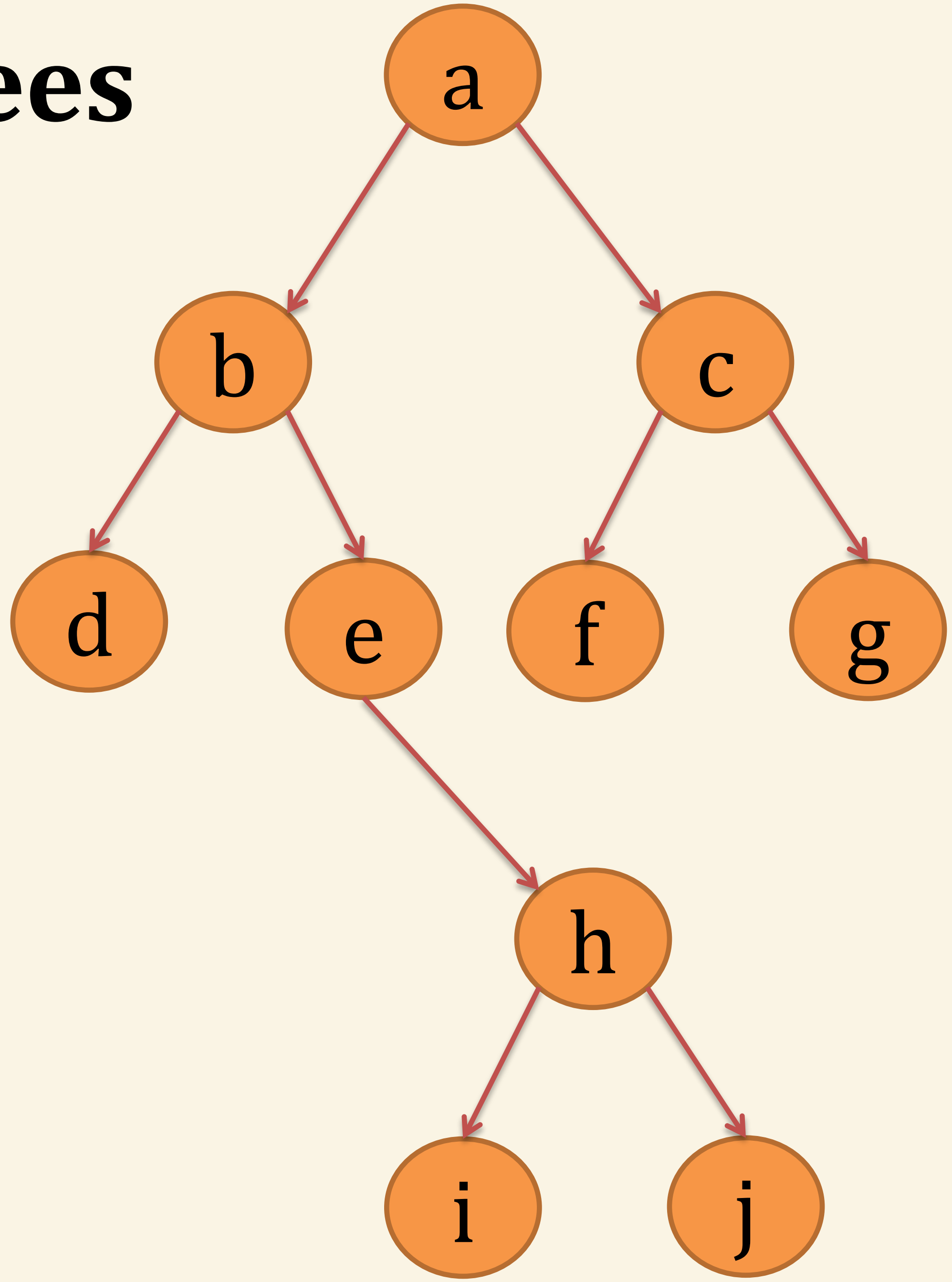
Binary Trees

Example: For the tree as shown in fig:

- Which node is the root?
- Which nodes are leaves?
- Name the parent node of each node.



Binary Trees



Solution:

Q1. Which node is the root?

Ans. The node A is the **root** node.

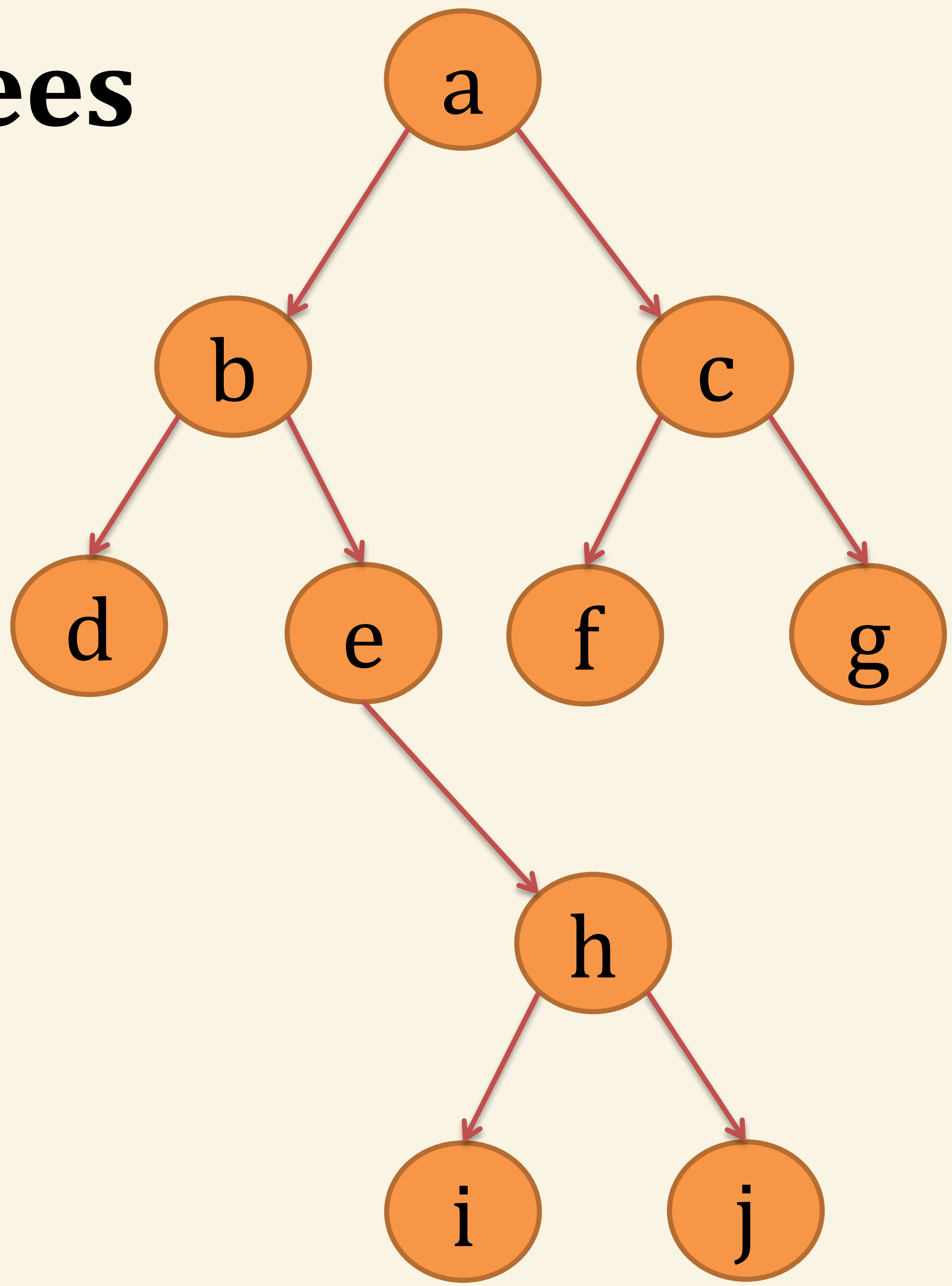
i. The nodes **d, f, g, h, i and j** are **leaves**.

Binary Trees

Solution:

Q2. Which nodes are leaves?

Ans. The nodes **d, f, g, i and j** are **leaves**.



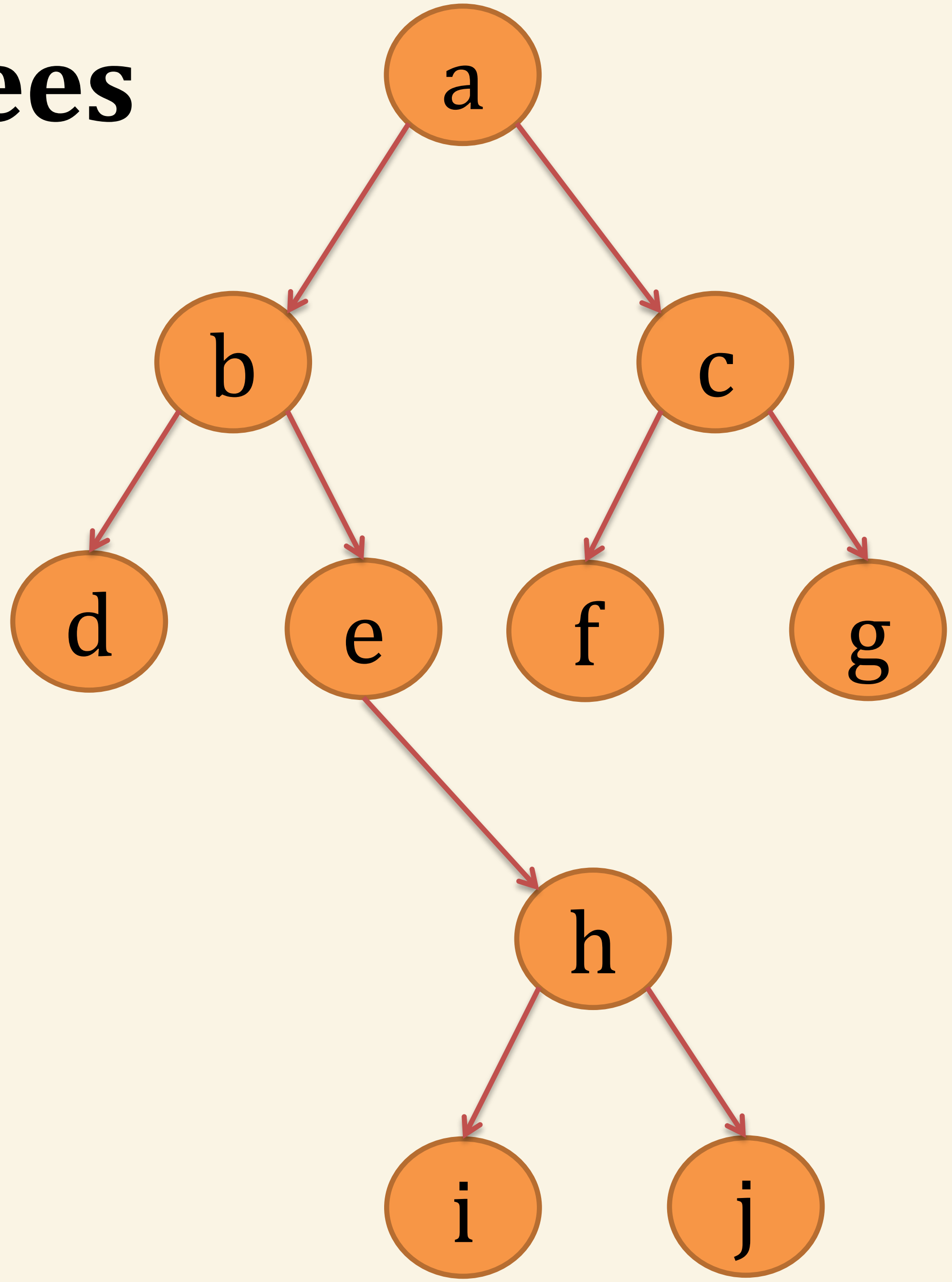
Binary Trees

Solution:

Q3. Name the parent node of each node.

Ans.

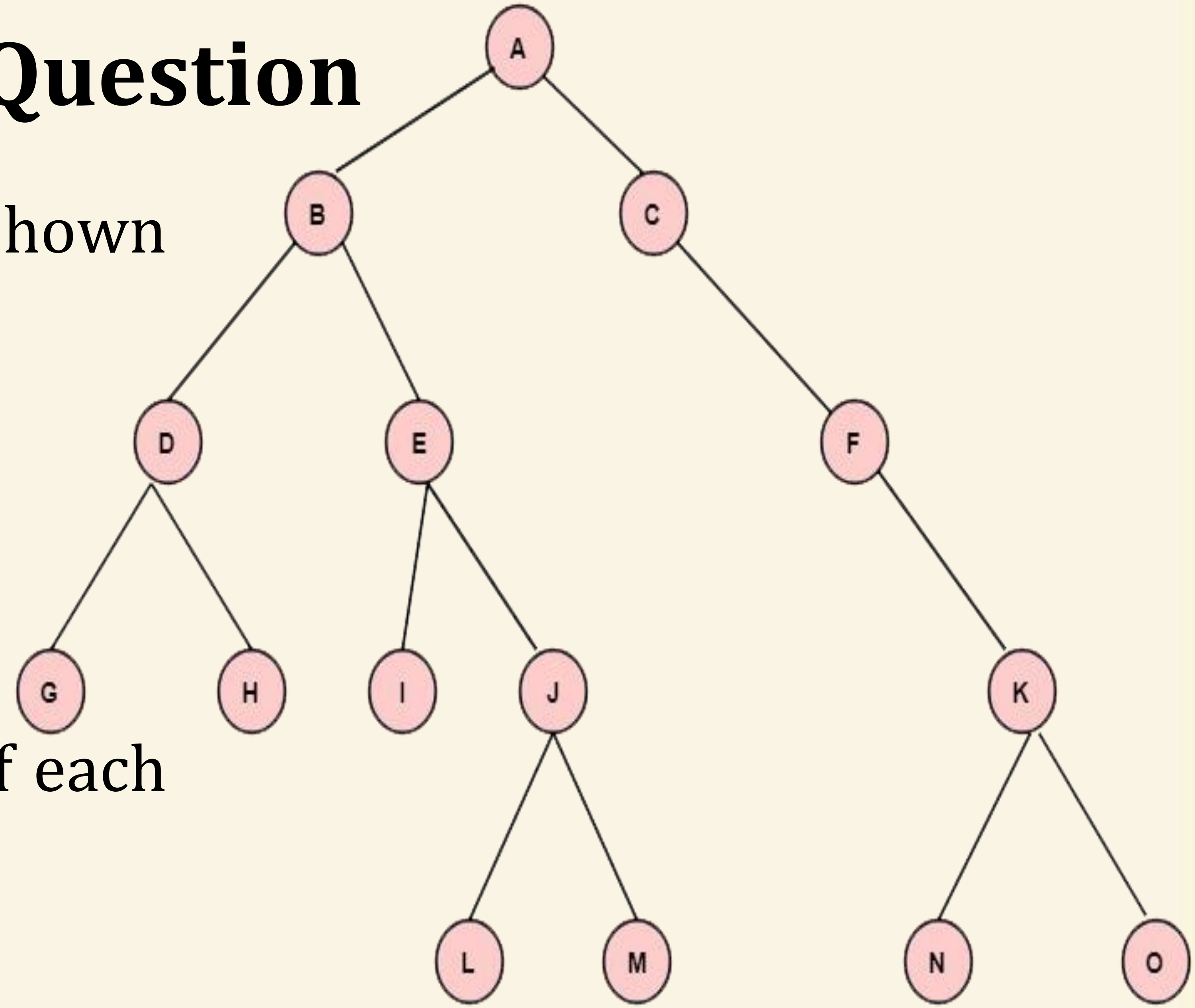
Nodes	Parent
b, c	a
d, e	b
f, g	c
i, j	h
h	e



Question

Example: For the tree as shown in fig:

- Which node is the root?
- Which nodes are leaves?
- Name the parent node of each node.



Thank You