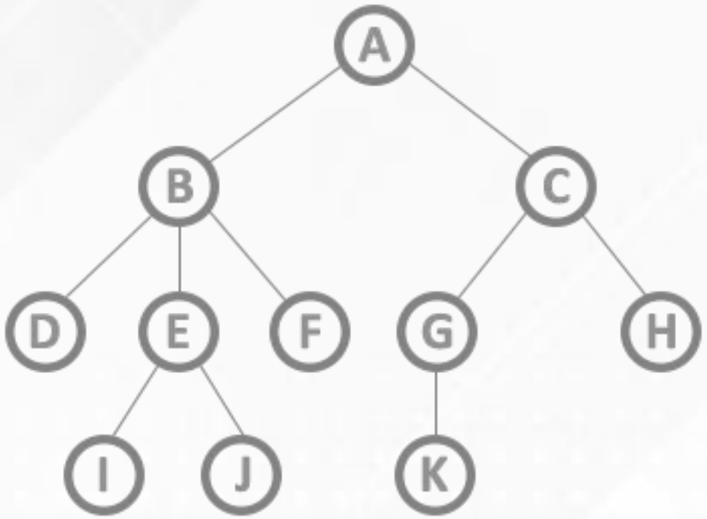
Unit-3

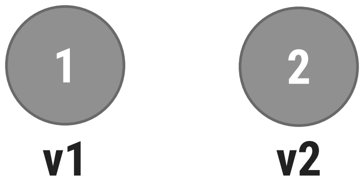
Non-Linear Data Structure (Tree Part-1)



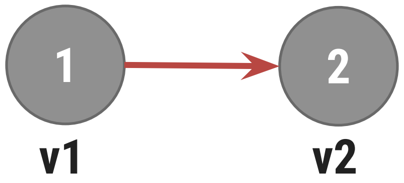
Non Linear Data Structures – Tree & Graph

- Far we have been concerned with Linear List.
- Relationships in such data structures expressed are One Dimensional.
- Now, Introduce Data Structures which are **Non – Linear**, Expressing more complex relationships.
- Non Linear Data Structures – **Graph** & **Tree** (Sub Set of Graph)
- **Tree is a Restricted Graph** – Every Tree is a Graph but Every Graph is not a Tree

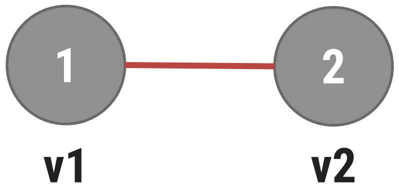
Basic Notations of Graph Theory



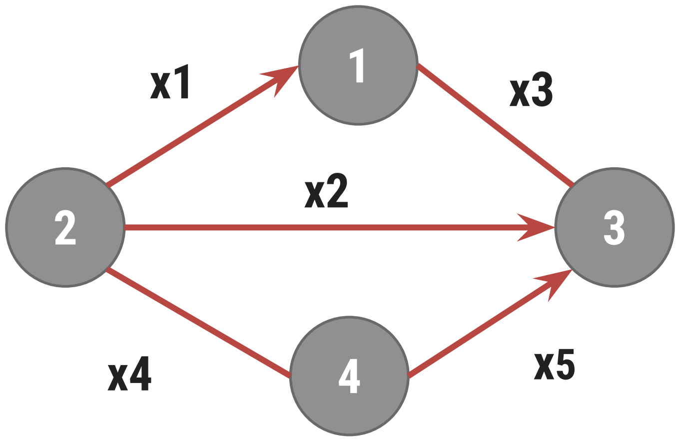
(a)



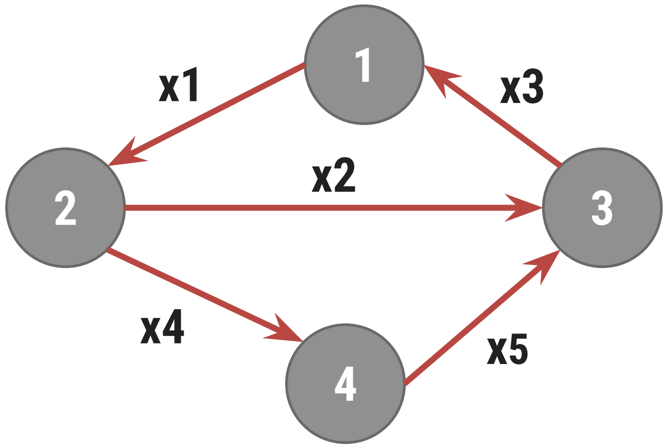
(b)



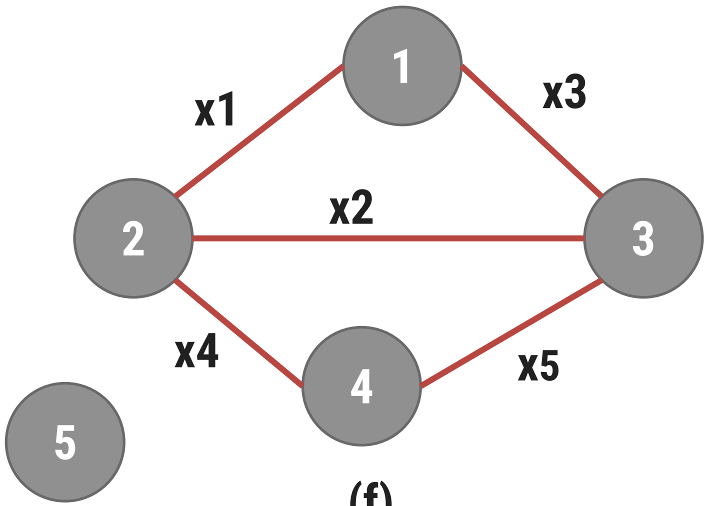
(c)



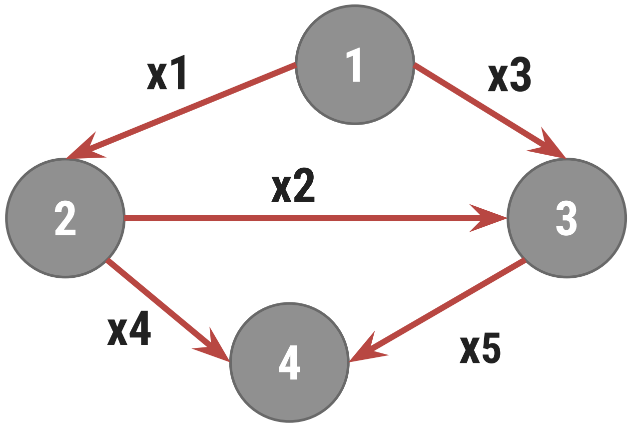
(d)



(e)



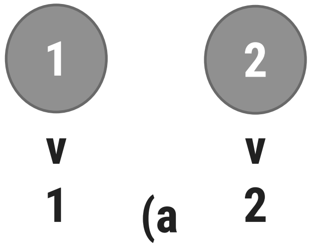
(f)



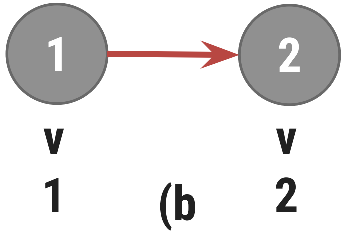
(g)

Basic Notations of Graph Theory

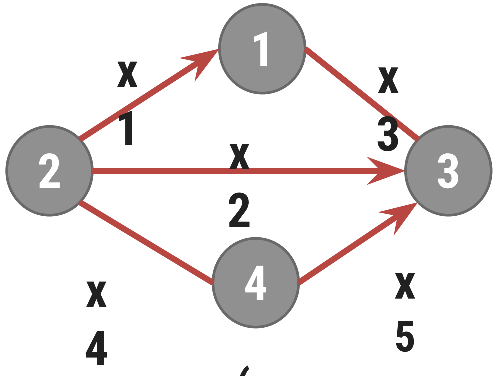
- Consider diagrams shown in above figure
- Every diagrams represent Graphs
- Every diagram consists of a **set of points** which are shown by **dots** or **circles** and are sometimes labelled V_1, V_2, V_3... OR 1,2,3...
- In every diagrams, certain pairs of such **points are connected by lines or arcs**
- Note that every arc start at one point and ends at another point



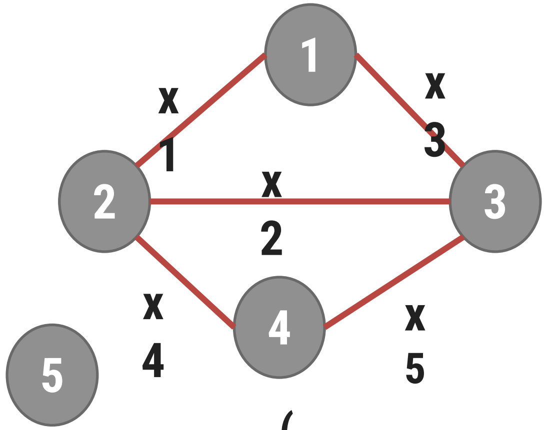
(a)



(b)



(d)



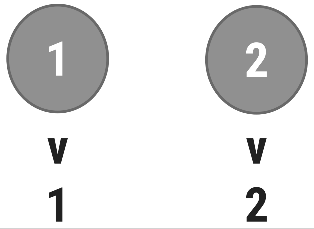
(f)

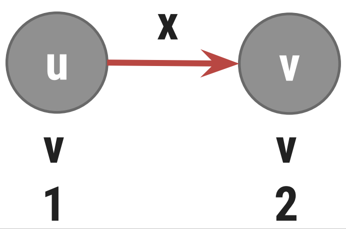
Graph

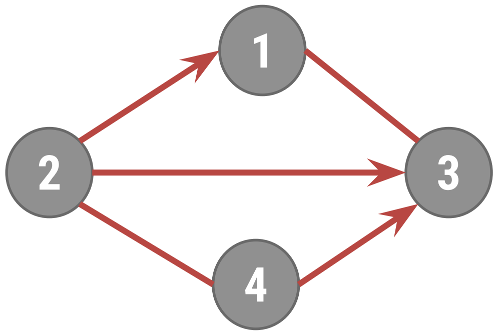
- **A graph G** consist of a **non-empty set V** called the **set of nodes** (points, vertices) of the graph, a **set E** which is the **set of edges** and a **mapping** from the set of edges **E to** a set of **pairs of elements of V**
- It is also convenient to write a graph as **G=(V,E)**
- Notice that definition of graph implies that to every edge of a graph G, we can associate a pair of nodes of the graph. If an edge X Є E is thus associated with a pair of nodes (u,v) where u, v Є V then we says that edge x connect u and v

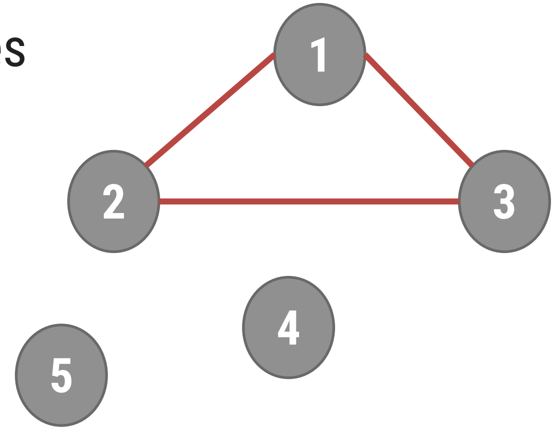
Adjacent Nodes

- Any two nodes which are connected by an edge in a graph are called adjacent nodes





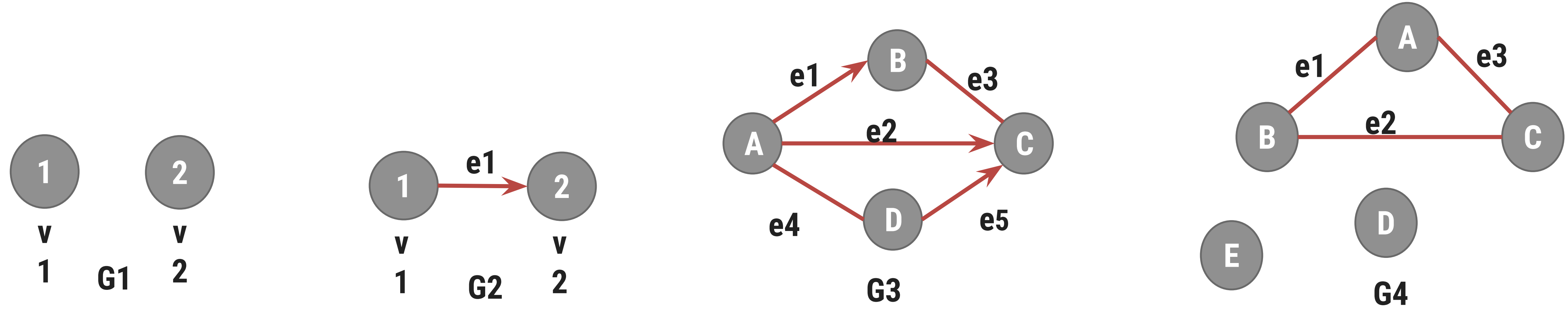




Basic Notations of Graph Theory

Graph

- G1=(V,E) , V = { v1,v2 } , E = { Φ }
- G2=(V,E) , V = { v1,v2 } , E = { e1 } , e1 = <v1,v2>
- G3=(V,E) , V = { A,B,C,D } , E = { <A,B> , <A,C> , (B,C) , <A,D> , <D,C> }
- G4=(V,E) , V = { A,B,C,D,E } , E = { (A,B) , (B,C) , (A,C) } or E = {e1,e2,e3}

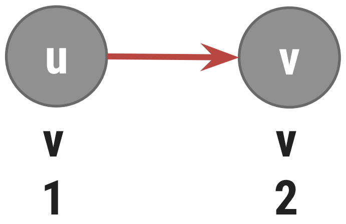


Graph – Concepts & Definitions

- **Directed & Undirected Edge**
  - In a graph **G=(V,E)** an **edge** which is **directed** from one end to another end is called a **directed edge**, while the edge which has no specific direction is called **undirected edge**
- **Directed graph (Digraph)**
  - A graph in which **every edge is directed** is called directed graph or digraph e.g. **b,e & g** are directed graphs
- **Undirected graph**
  - A graph in which **every edge is undirected** is called undirected graph e.g. **c & f** are undirected graphs
- **Mixed Graph**
  - If **some** of the **edges** are **directed** and **some are undirected** in graph then the graph is called mixed graph e.g. **d** is mixed graph

Graph – Concepts & Definitions

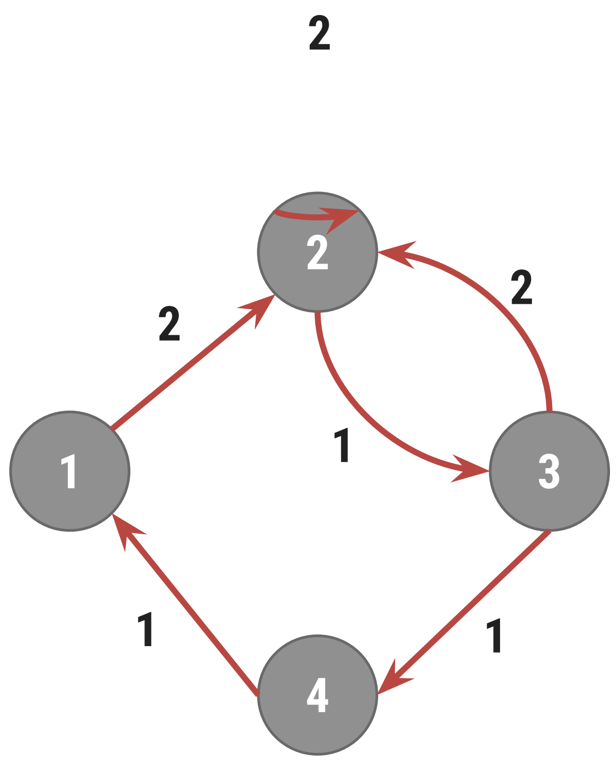
- In a graph **G=(V,E)** an **edge** which is **directed** from one end to another end, associated with pair of node (u,v)



- **Intial Node**
  - Edge initiating or originating from node **u**
- **Terminal Node**
  - Edge terminating or ending at node **v**
- Edge is said to be ***incident*** of the node u and v.

Graph – Concepts & Definitions

- **Loop (Sling)**
  - An **edge** of a graph **which joins a node to itself** is called a loop (sling).
  - The ***direction of a loop is of no significance*** so it can be considered either a directed or an undirected.
- **Distinct Edges**
  - In case of directed edges, **two possible edges** between any pair of nodes which **are opposite in direction** are considered **Distinct**.
- **Parallel Edges**
  - In some directed as well as undirected graphs, we may have **certain pairs of nodes joined by more than one edges**, such edges are called **Parallel** edges.



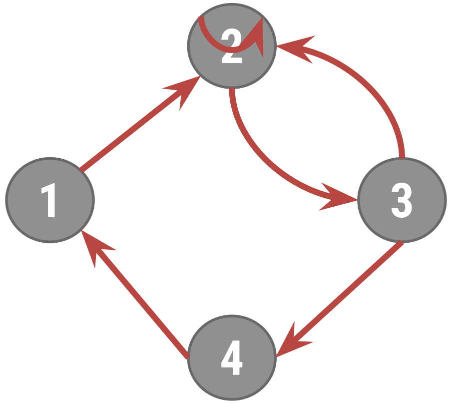
Graph – Concepts & Definitions

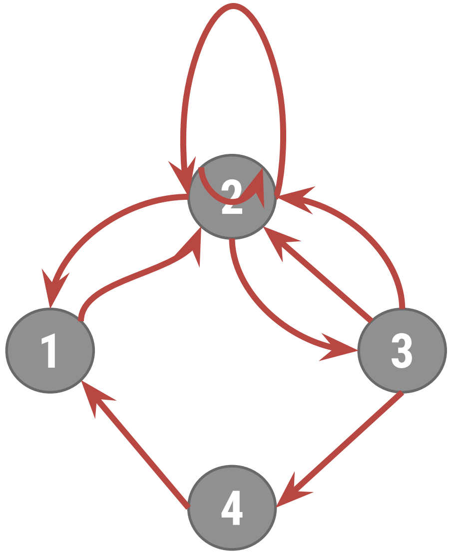
Multigraph

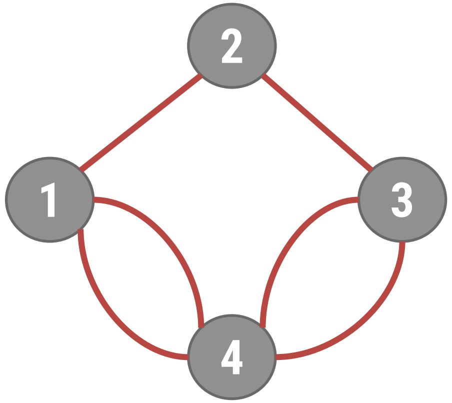
- Any **graph** which **contains** some **parallel edges** is called **multigraph**
- If there is no more then one edge between a pair of nodes then such a graph is called **Simple graph**

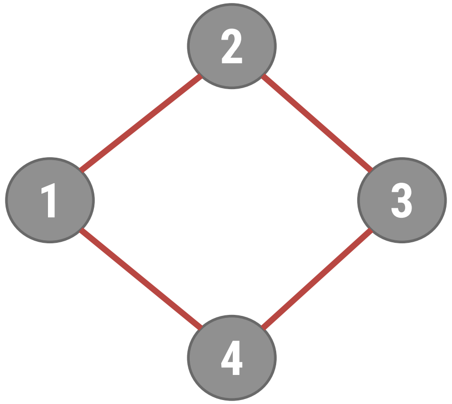
Weighted Graph

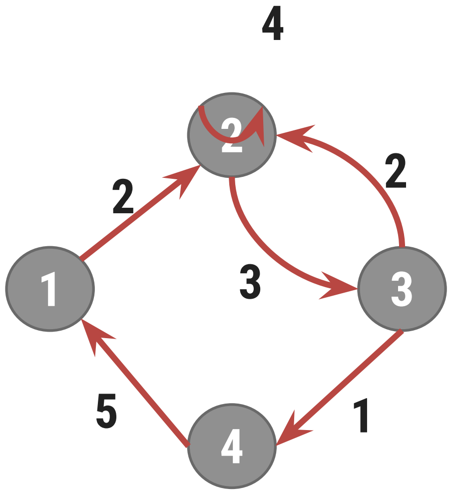
- A graph in which **weights are assigned to every edge** is called weighted graph











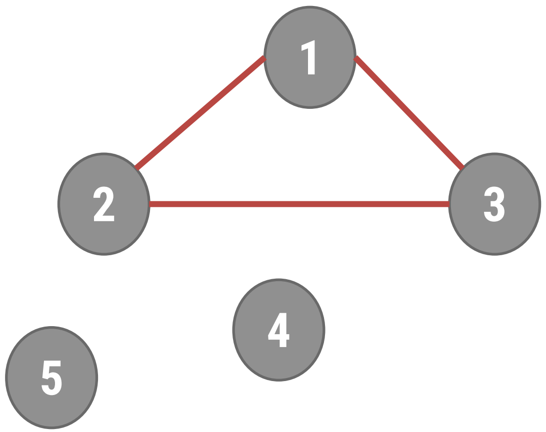
Graph – Concepts & Definitions

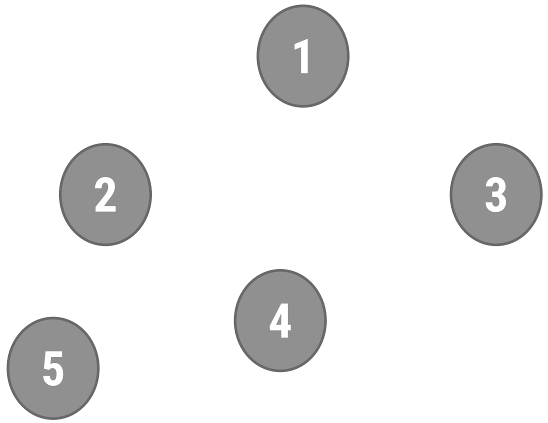
Isolated Node

- In a graph a **node** which is **not adjacent to any other node** is called isolated node

Null Graph

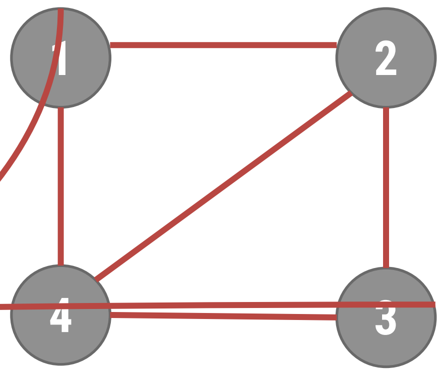
- A graph **containing only isolated nodes** are called null graph. In other words set of edges in null graph is empty

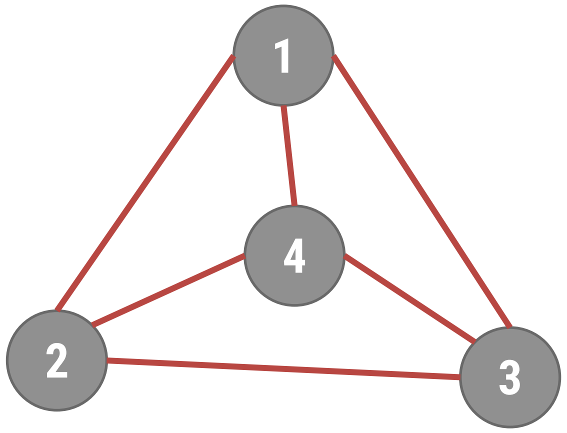




Graph – Concepts & Definitions

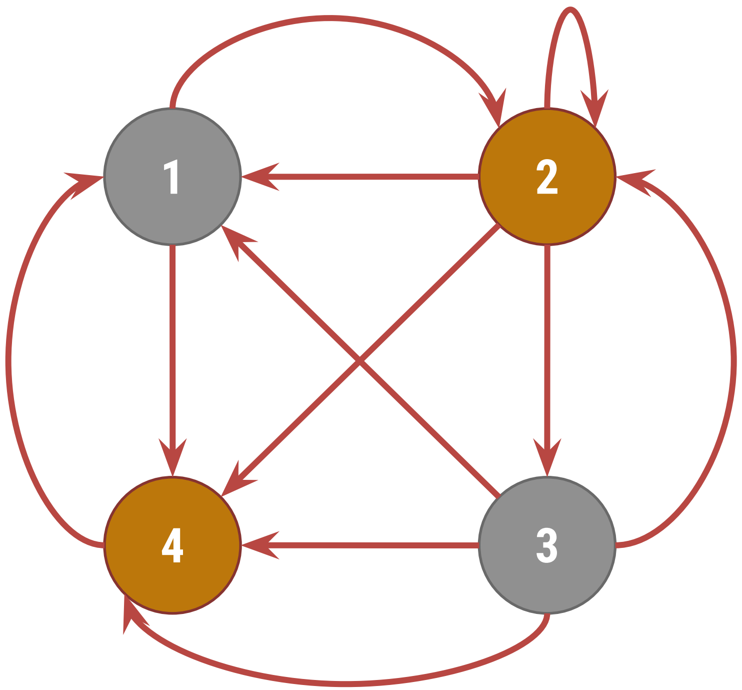
- For a given **graph** there is **no unique diagram** which represents the graph.
- We can obtain a variety of diagrams by locating the nodes in an arbitrary numbers.
- Following both diagrams represents same Graph.





- **Indegree of Node**
  - The **no of edges** which have **V as their terminal node** is call as indegree of node V.
- **Outdegree of Node**
  - The **no of edges** which have **V as their initial node** is call as outdegree of node V.
- **Total degree of Node**
  - Sum of indegree and outdegree of node V is called its Total Degree or Degree of vertex.

Path of the Graph



Some of the path from 2 to 4

P1 = ((2,4))

P2 = ((2,3), (3,4))

P3 = ((2,1), (1,4))

P4 = ((2,3), (3,1), (1,4))

P5 = ((2,3), (3,2), (2,4))

P6 = ((2,2), (2,4))

- Let G=(V, E) be a simple digraph such that the terminal node of any edge in the sequence is the initial node of the edge, if any appearing next in the sequence defined as **path of the graph**.
- **Length of Path**
  - The number of edges appearing in the sequence of the path is called length of path.

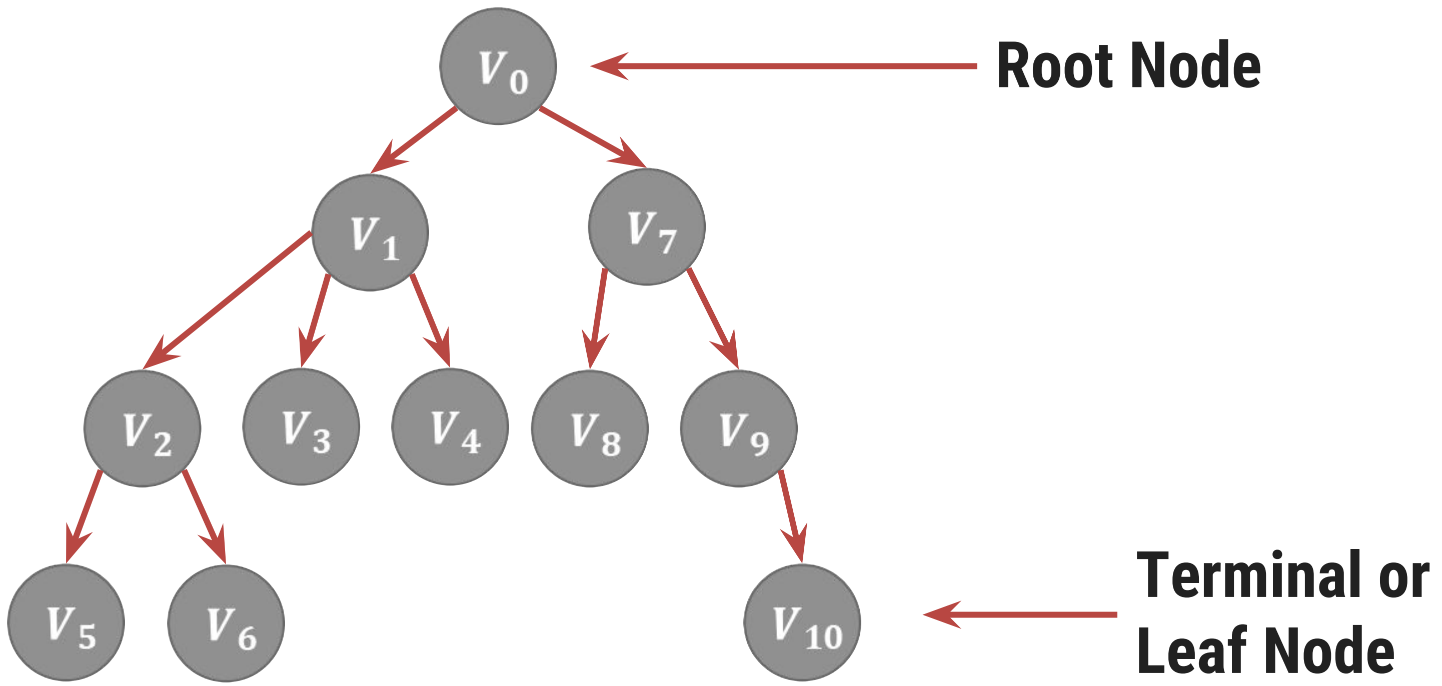
Graph – Concepts & Definitions

- **Simple Path (Edge Simple)**
  - A **path** in a diagraph in which **the edges are distinct** is called simple path or edge simple
  - Path P5, P6 are Simple Paths
- **Elementary Path (Node Simple)**
  - A **path** in which **all the nodes through which it traverses** are **distinct** is called elementary path
  - Path P1, P2, P3 & P4 are elementary Path
  - Path P5, P6 are Simple but not Elementary
- **Cycle (Circuit)**
  - A **path** which **originates and ends in the same node** is called cycle (circuit)
  - E.g. C1 = ((2,2)), C2 = ((1,2),(2,1)), C3 = ((2,3), (3,1), (1,2)
- **Acyclic Diagraph**
  - A simple **diagraph which does not have any cycle** is called Acyclic Diagraph.

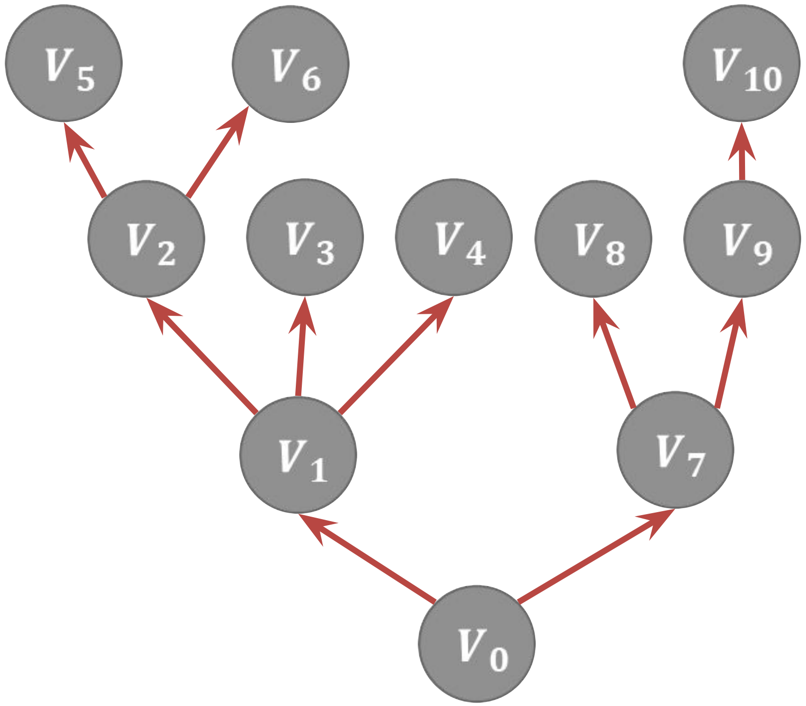
Tree– Concepts & Definitions

Directed Tree

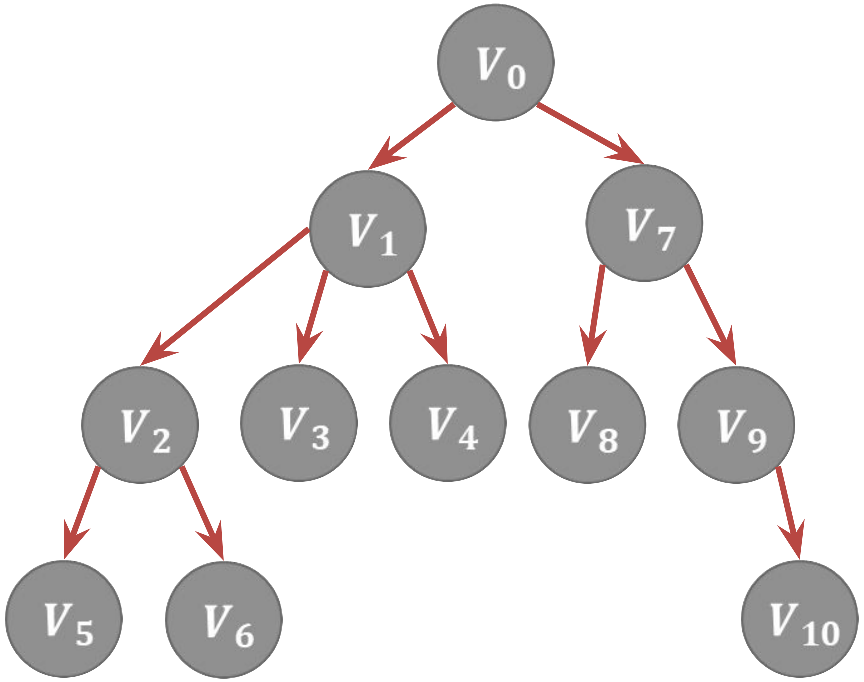
- A directed tree is an acyclic digraph which has one node called its root with in degree 0, while all other nodes have in degree 1.
- Every directed tree must have at least one node.
- An isolated node is also a directed tree.



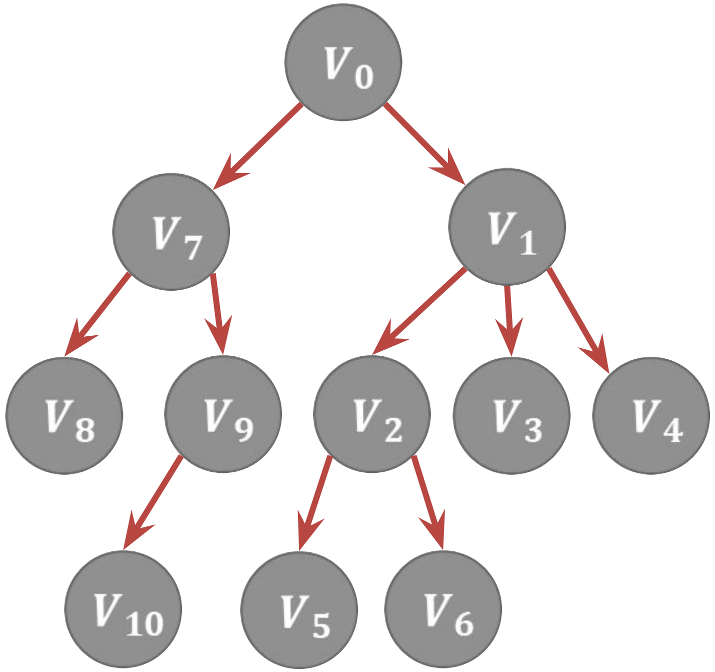
Tree– Concepts & Definitions



(a)



(b)



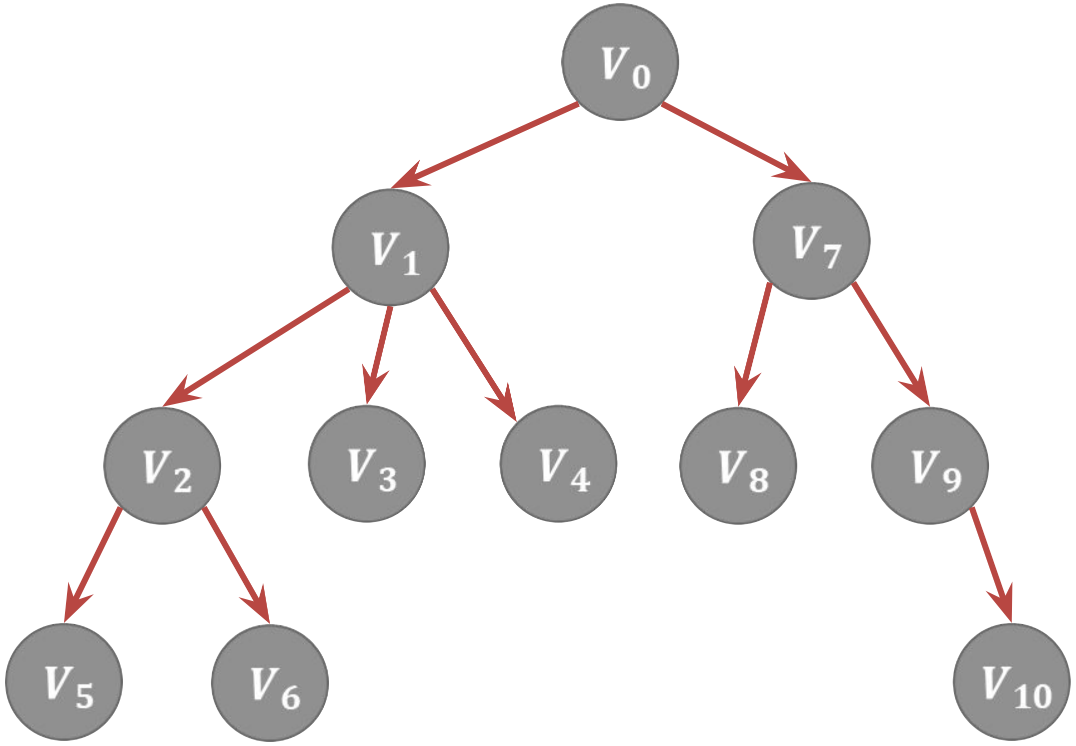
(c)

Tree– Concepts & Definitions

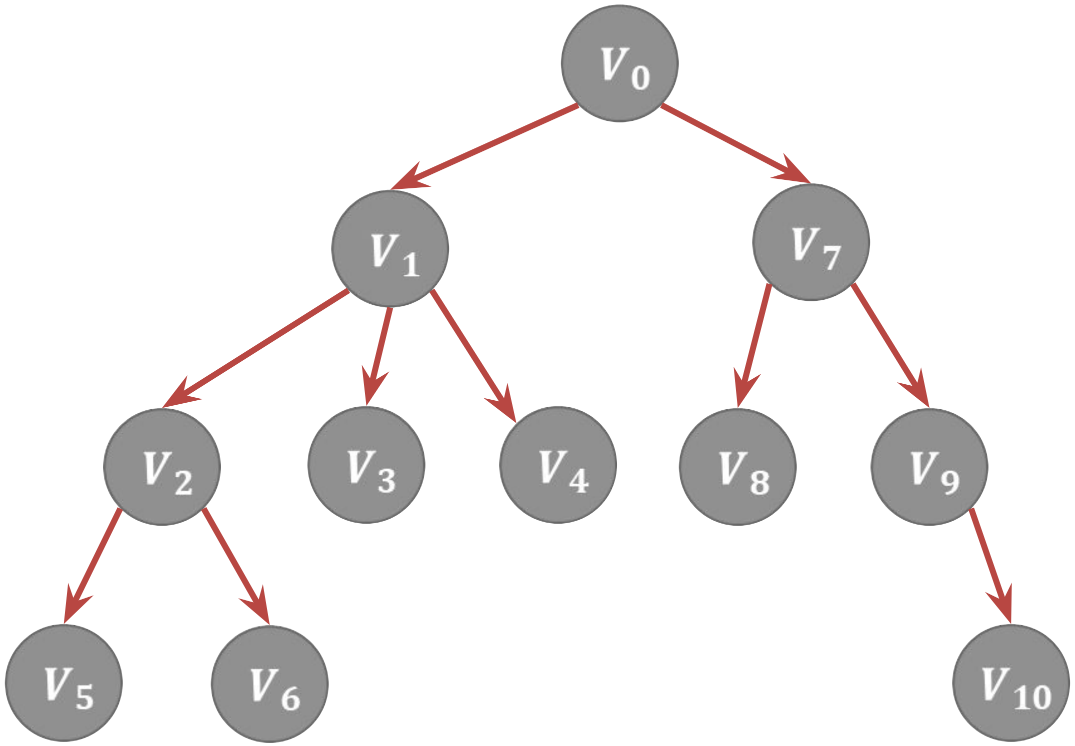
- **Terminal Node (Leaf Node)**
  - In a directed tree, any **node** which **has out degree 0** is called terminal node or leaf node.
- **Level of Node**
  - The level of any node is the length of its path from the root.
- **Ordered Tree**
  - In a directed tree an ordering of the nodes at each level is prescribed then such a tree is called ordered tree.
  - The diagrams (b) and (c) represents same directed tree but different ordered tree.
- **Forest**
  - If we delete the root and its edges connecting the nodes at level 1, we obtain a set of disjoint tree. A set of disjoint tree is a forest.

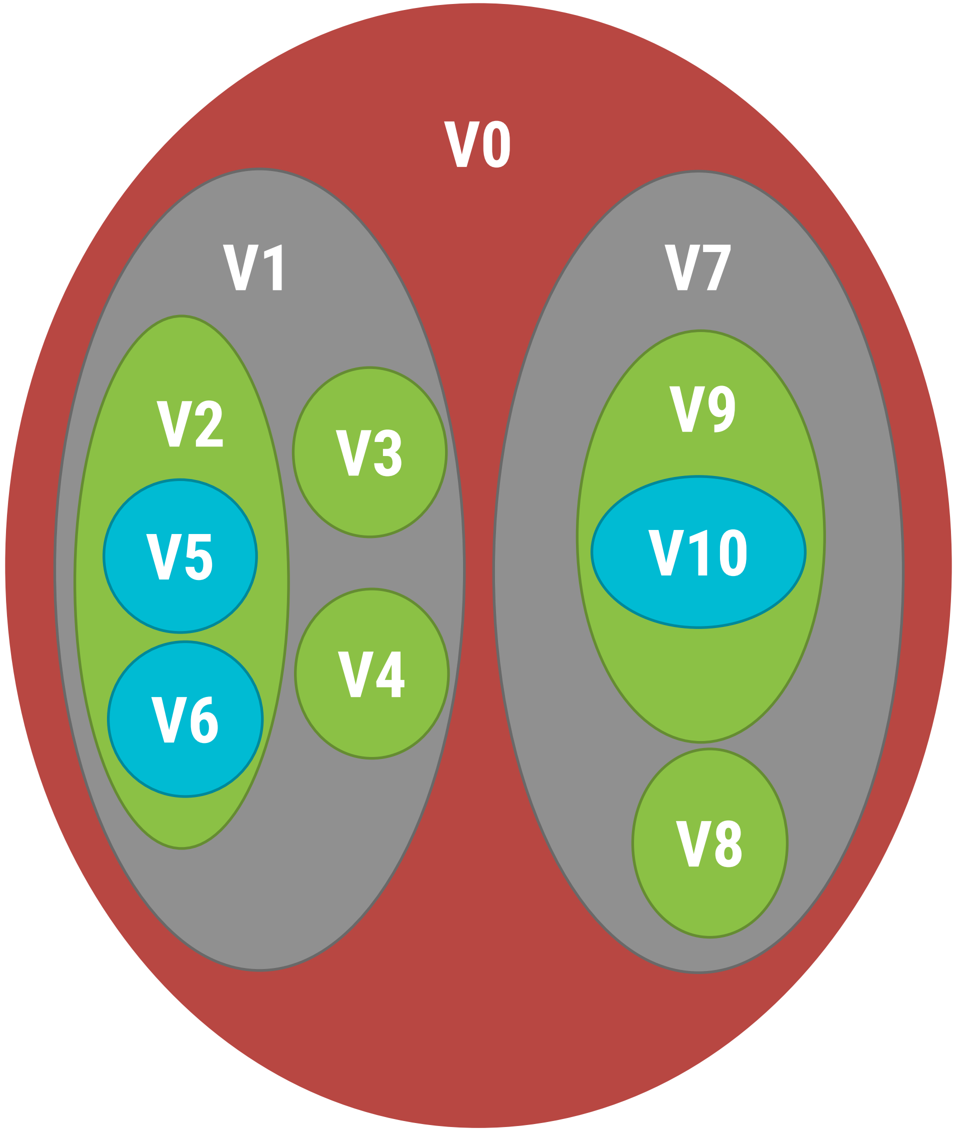
Representation of Directed Tree

- Other way to represent directed tree are
  - Venn Diagram
  - Nesting of Parenthesis
  - Like table content of Book
  - Level Format

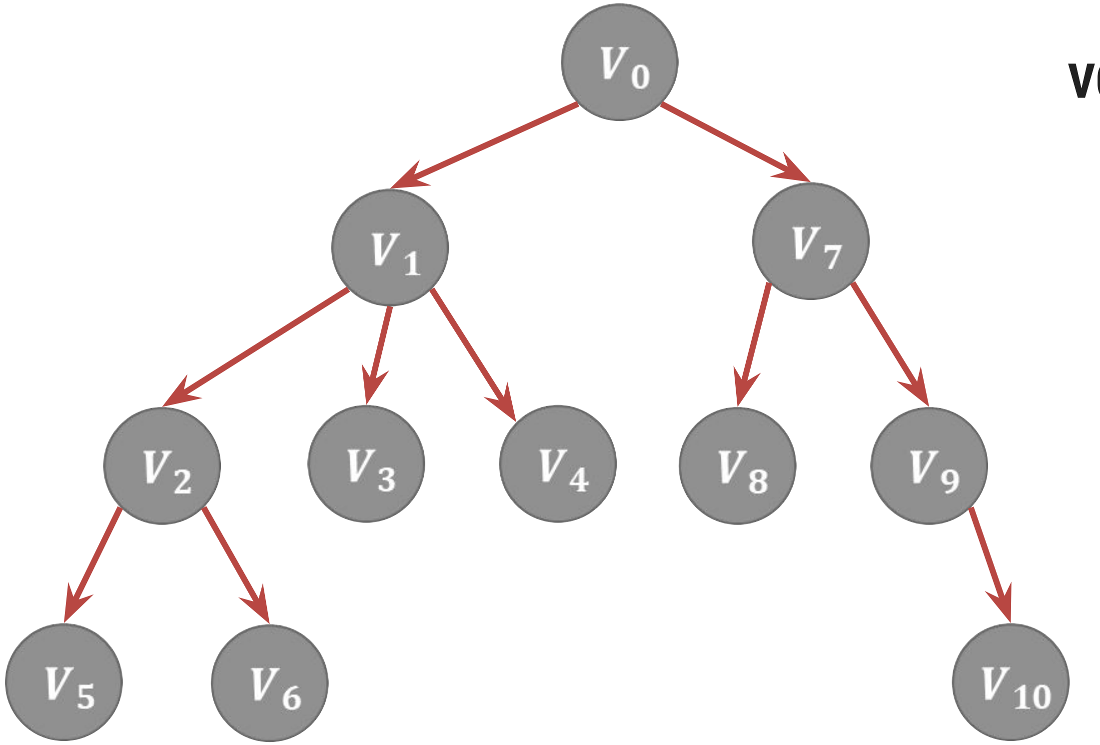


Venn Diagram

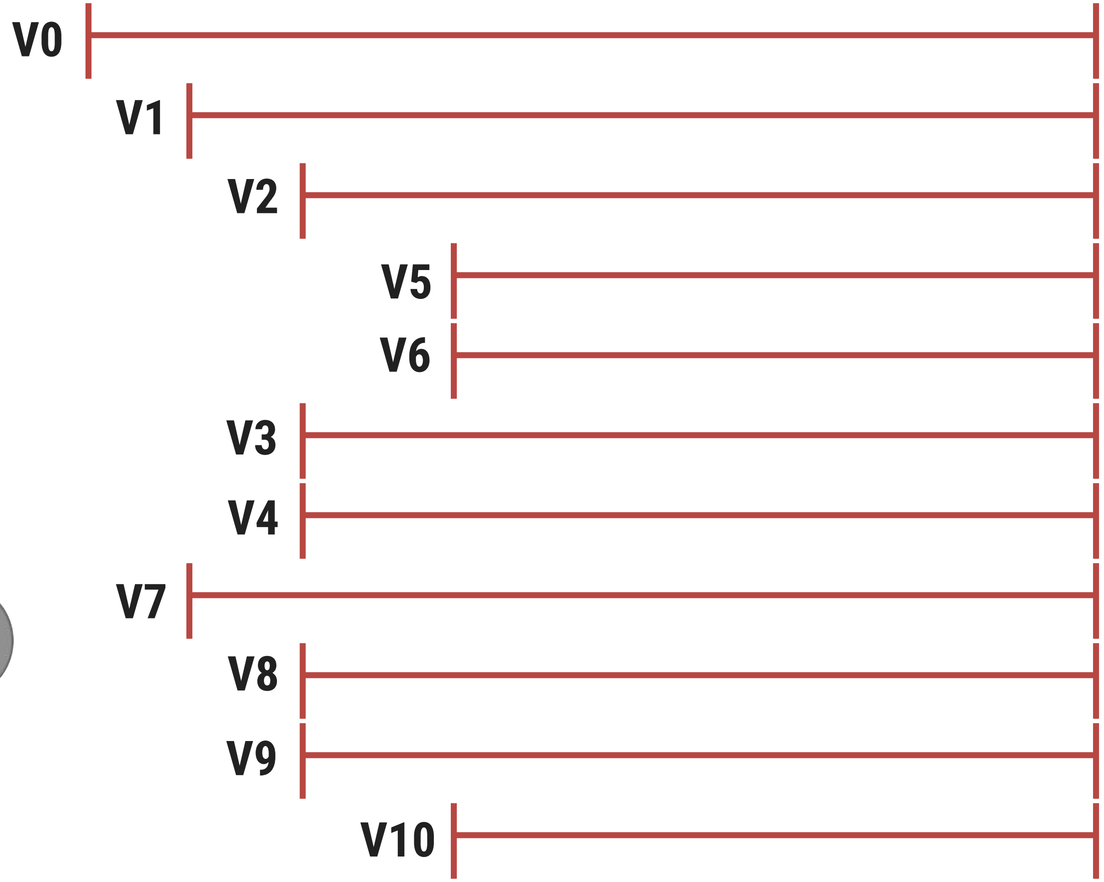




Nesting of Parenthesis



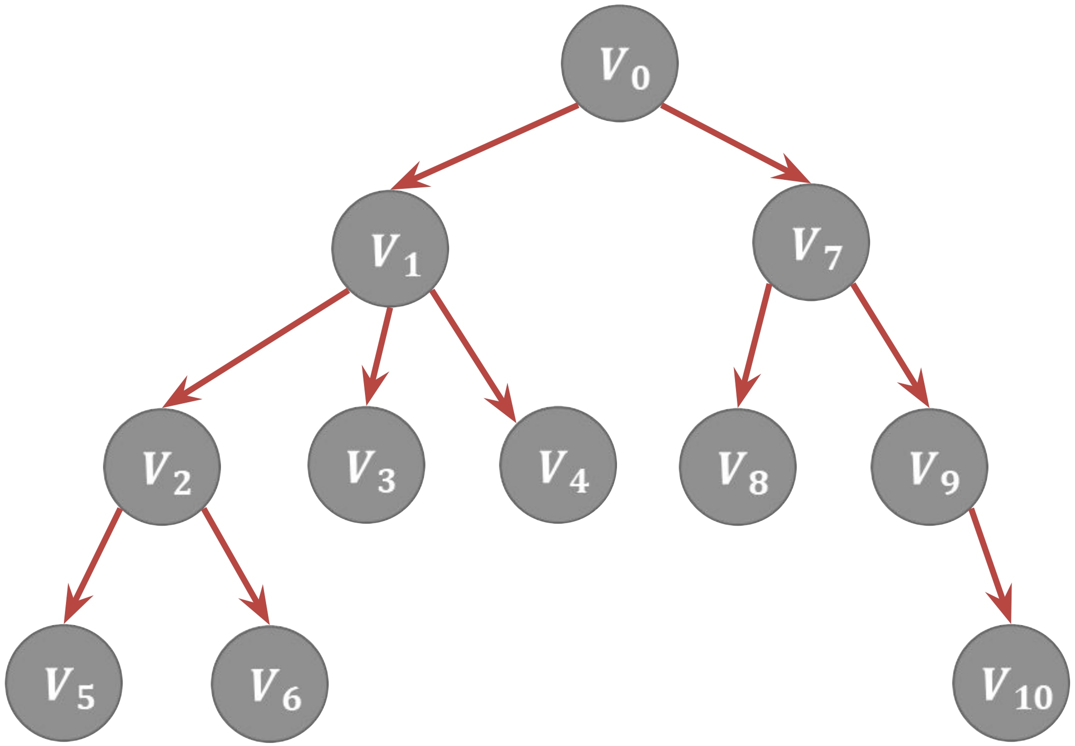
Like a table Content of Book



$(V_0\ (V_1\ (V_2\ (V_5)\ (V_6)\ )\ (V_3)\ (V_4)\ )\ (V_7\ (V_8)\ (V_9\ (V_{10})\ )\ )\ )$

Nesting of Parenthesis

Level Format



1 V_0

2 V_1

3 V_2

4 V_5

4 V_6

3 V_3

3 V_4

2 V_7

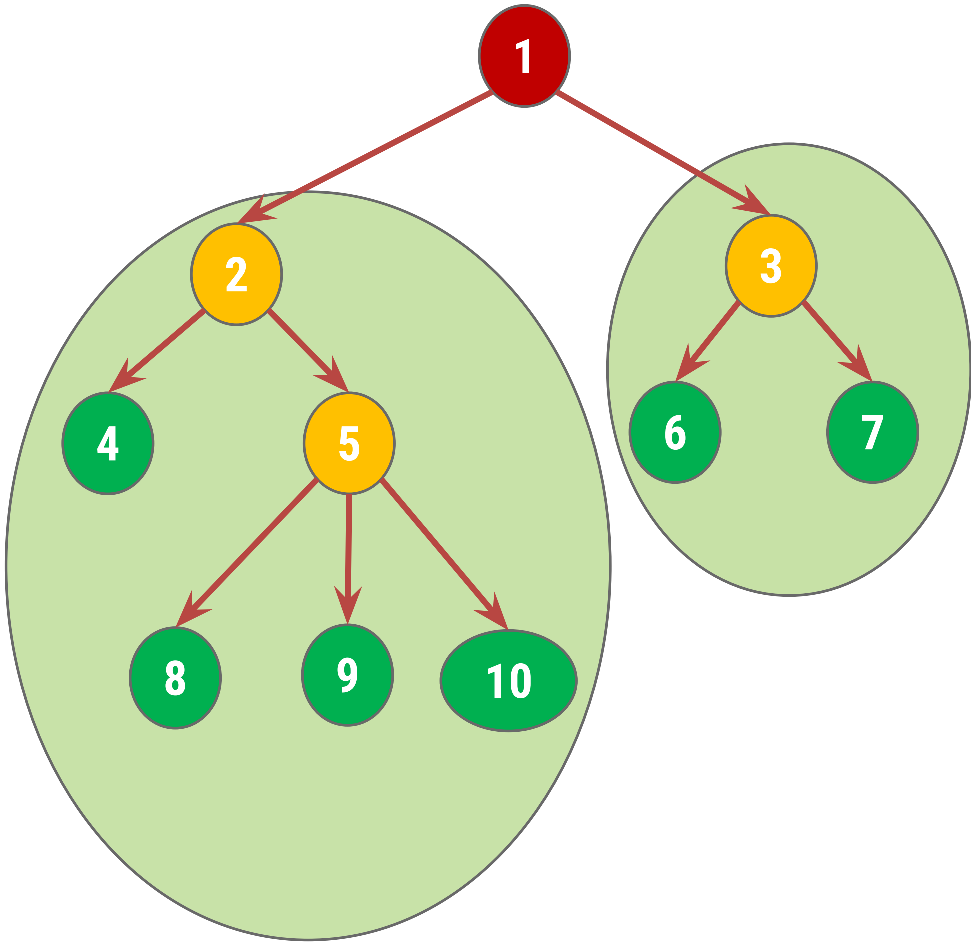
3 V_8

3 V_9

4 V_{10}

Tree– Concepts

- Root Node [1]
  - In-degree : **0** Out-degree : **0 , 1 , ……. , m**
- Intermediate Node [2, 3, 5]
  - In-degree : **1** Out-degree : **0 , 1 , ………, m**
- Leaf Node [4, 6, 7, 8, 9, 10]
  - In-degree : **1** Out-degree : **0**
- Tree : [1]
  - Left Sub Tree : [2,4,5,8,9,10]
  - Right Sub Tree : [3,6,7]



Tree– Concepts & Definitions

- The node that is reachable from a node is called **descendant** of a node.
- The nodes which **are reachable from a node through a single edge** are called the **children of node**.

Tree– Concepts & Definitions

- **M-ary Tree**
  - If in a directed tree the **out degree of every node** is **less than or equal to m** then tree is called an m-ary tree.
- **Full or Complete M-ary Tree**
  - If **the out degree of each and every node** is **exactly equal to m or 0** and their **number of nodes at level i is m(i-1)** then the tree is called a full or complete m-ary tree.
- **Positional M-ary Tree**
  - If we consider m-ary trees in which the m children of any node are assumed to have m distinct positions, if such positions are taken into account, then tree is called positional m-ary tree.

Tree– Concepts & Definitions

- **Height of the tree**
  - The height of a tree is the length of the path from the root to the deepest node in the tree.
- **Binary Tree**
  - If in a directed tree the **out degree of every node** is **less than or equal to 2** then tree is called binary tree.
- **Strictly Binary Tree**
  - A strictly binary tree (sometimes proper binary tree or 2-tree or full binary tree) is a tree in **which every node other than the leaves has two children.**
- **Complete Binary Tree**
  - If the **out degree of each and every node is exactly equal to 2 or 0** and **their number of nodes at level i is 2(i-1)** then the tree is called a full or complete binary tree.

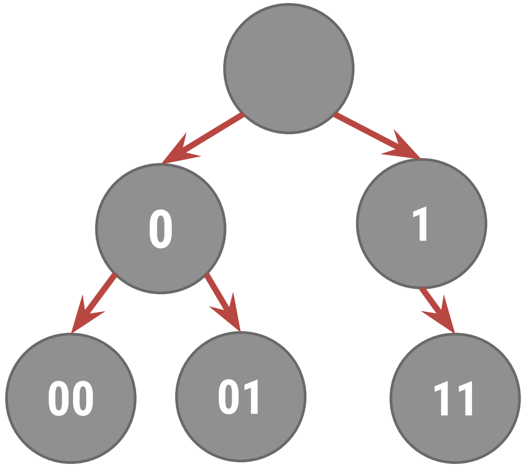
Tree– Concepts & Definitions

Sibling

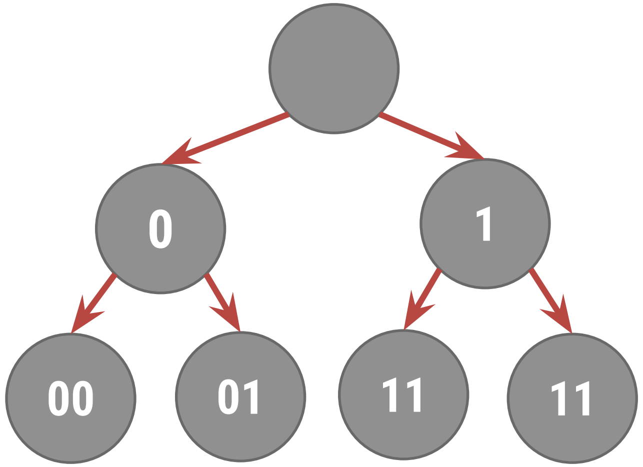
- Siblings are nodes that share the same parent node

Positional m-ary Tree

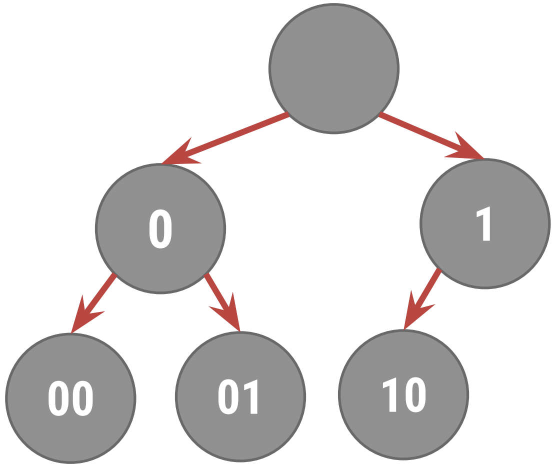
- If we consider m-ary trees in which the **m children of any node** are assumed **to have m distinct positions**, if such positions are taken into account, then tree is called positional m-ary tree



(a) Binary tree



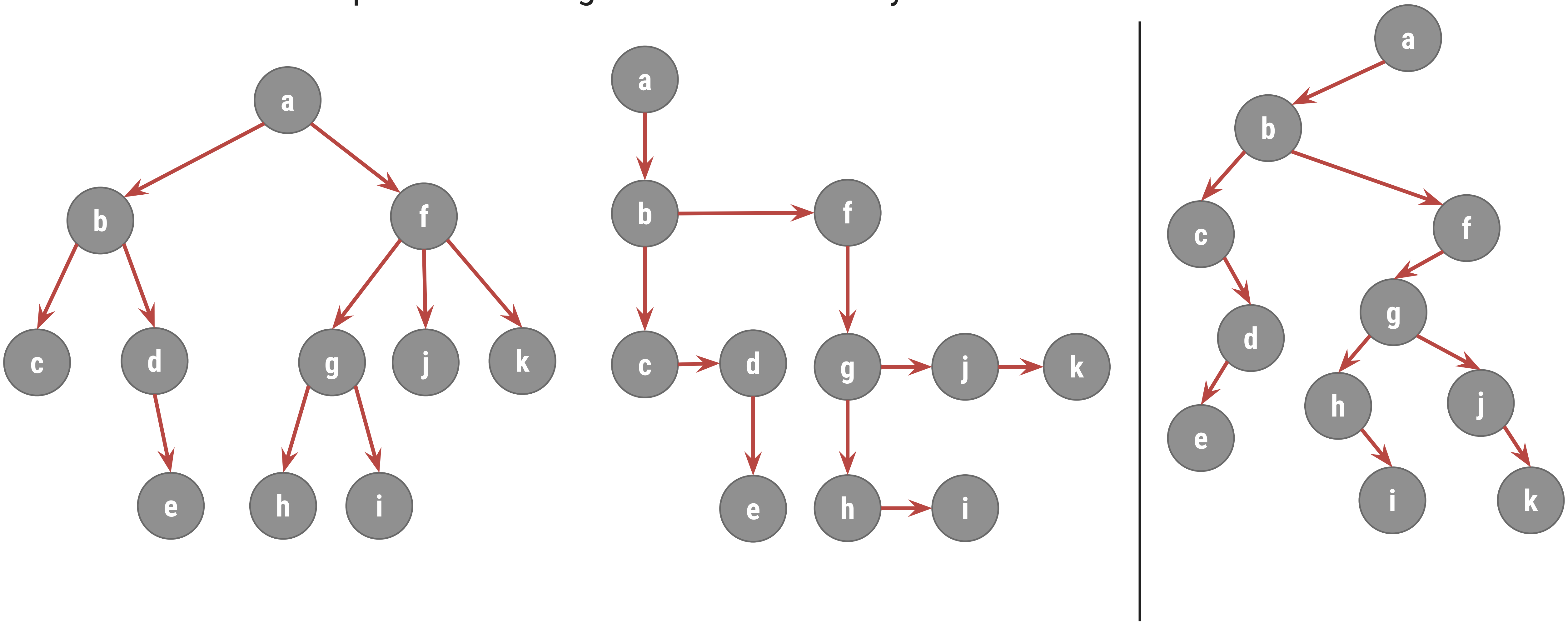
(b) Complete binary tree



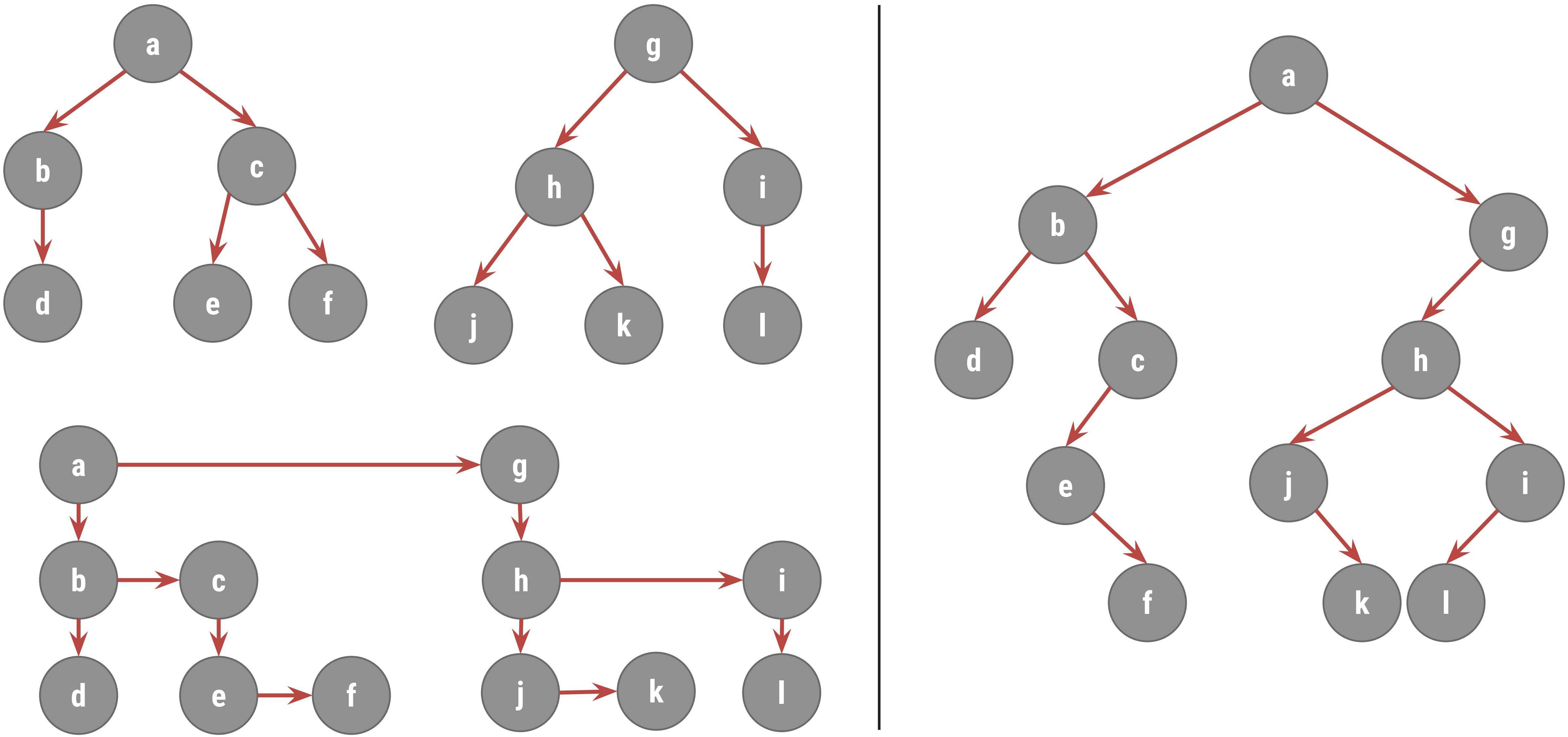
(c)

Convert any tree to Binary Tree

- Every Tree can be Uniquely represented by binary tree
- Let’s have an example to convert given tree into binary tree



Convert Forest to Binary Tree



Thank You