# CS2x1:Data Structures and Algorithms

Koteswararao Kondepu

k.kondepu@iitdh.ac.in

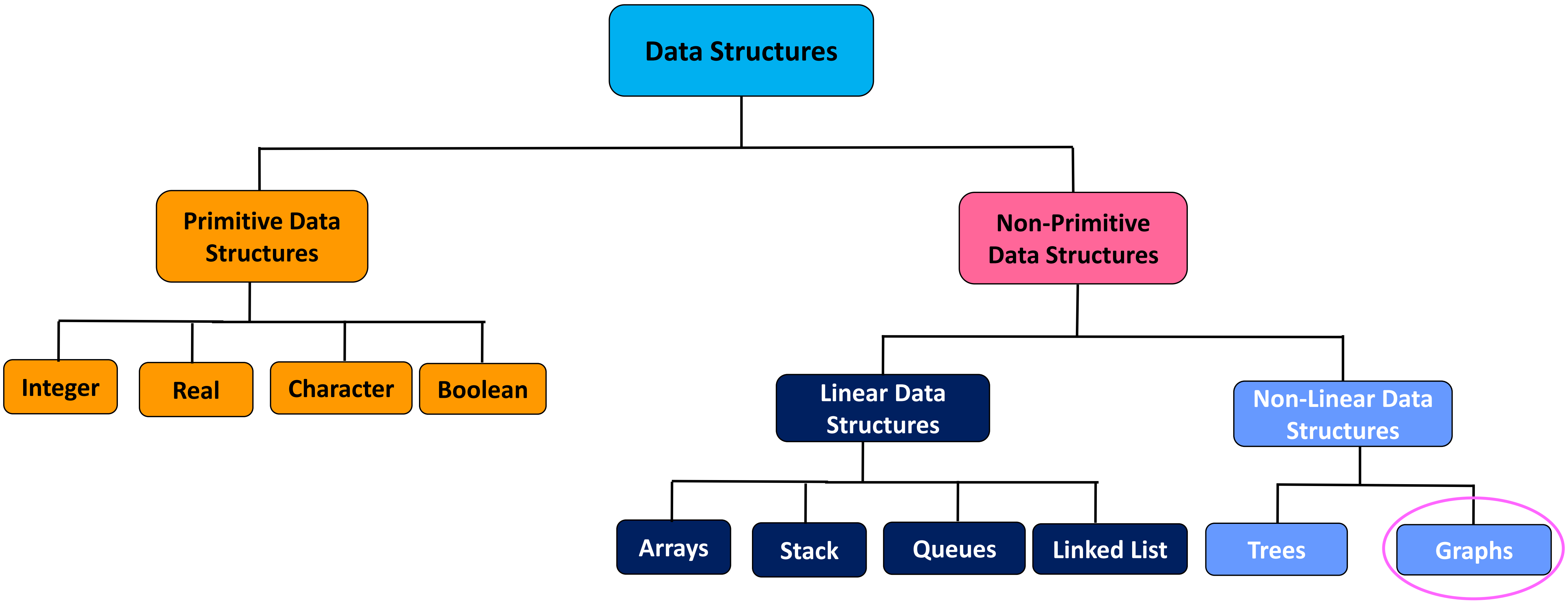
# List of Topics [C201]

- Introduction:
  - *Data structures*
  - *Abstract data types*
  - *Analysis of algorithms.*

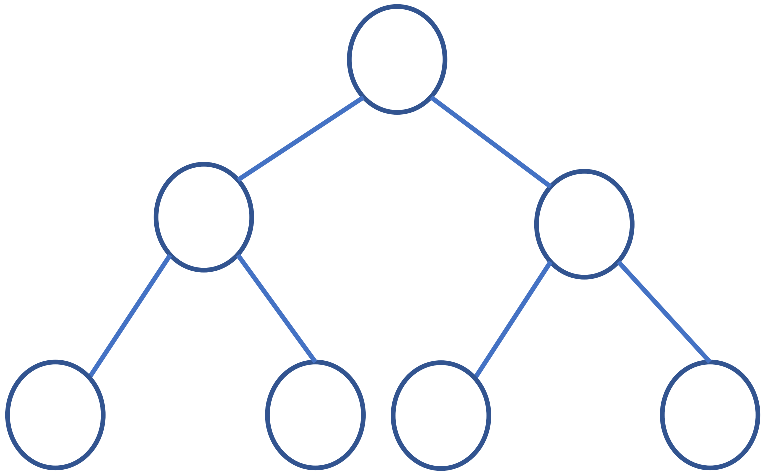
- Creation and manipulation of data structures:
  - *Arrays; Stacks; Queues; Linked lists; Trees; Heaps; Hash tables; Balanced trees [AVL]; Graphs.*

- Algorithms for sorting and searching, depth-first and breadth-first search, shortest paths and minimum spanning tree.

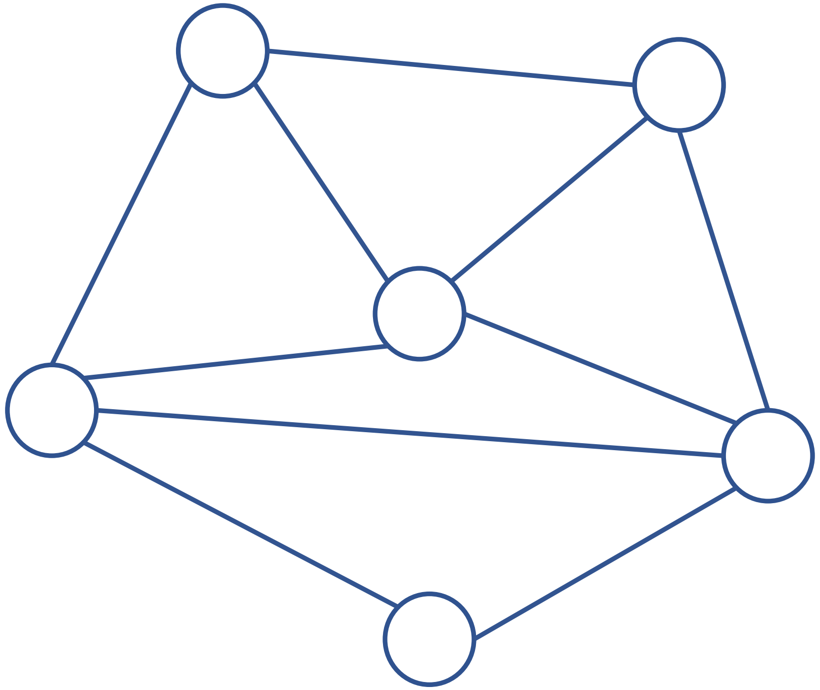
# Classification of Data Structures



# Definition: Graphs



Tree: One-to-Many

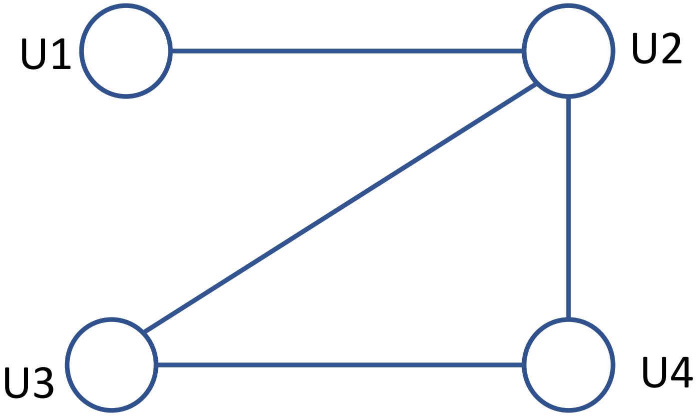


Graph: Many-to-Many

- ✓ A Graph is a non-linear data structure
- ✓ A graph G = (V;E) is defined by *a set of vertices V* , and *a set of edges E* consisting of *ordered or unordered pairs of vertices from V*
- ✓ Vertex is usually represented by a circle with a label A
- ✓ Edge is usually represented by a line or arrow extending from one vertex to another
- ✓ A tree is also a type of graph
- ✓ Graphs also have many application in the real world: road network, rail network, airline network, water network, telecommunication network, etc.

# Terminology: Graphs (1)

- **Undirected graph**: A graph G = (V;E); V → set of all vertices; E→ set of all edges (*set of all unordered pairs* of elements from V
- Example: V ={U1, U2, U3, U4}; E ={(U1, U2), (U2, U4), (U2, U4), (U3, U4)}

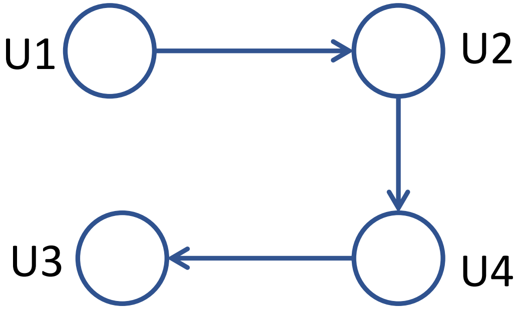



- ✓ *Vertices* →Users: U1, U2, U3 and U4
- ✓ *Edges* → Friendship (graph) among the users

- ✓ Undirected graph → (U1, U2) is equal to (U2, U1)
- ✓ Undirected edge → unordered pair of vertices
- ✓ Undirected graph → termed as "graph"

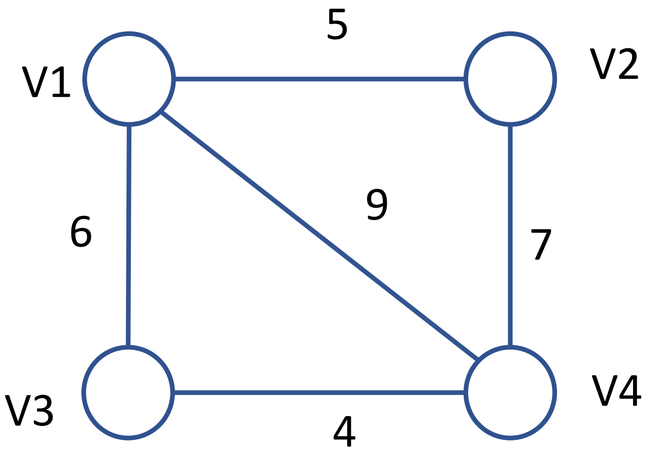
Railway line

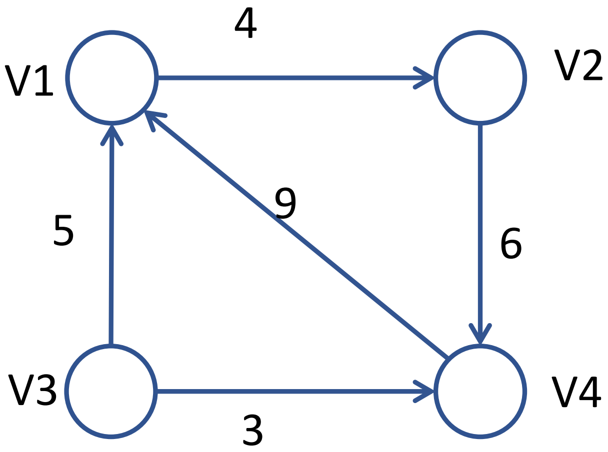
- **Directed graph**: A graph G = (V;E); V → set of all vertices; E→ set of all edges (*set of all ordered pairs* of elements from V
- Example: V ={U1, U2, U3, U4}; E ={(U1, U2), (U2, U4), (U4, U3)}
  - ✓ Directed graph → (U1, U2) is not equal to (U2, U1)
  - ✓ U1 follows U2
  - ✓ Direct edge → ordered pair of vertices
  - ✓ **Directed graph → termed as "digraph"**

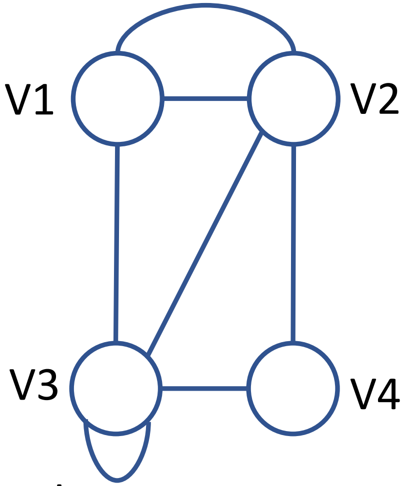



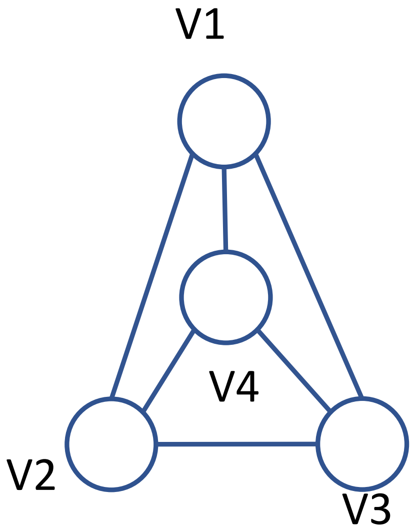
# Terminology: Graphs (2)

- **Weighted graph**: A graph (or digraph) terms as weighted graph, if all the edges in it are labelled with some weights



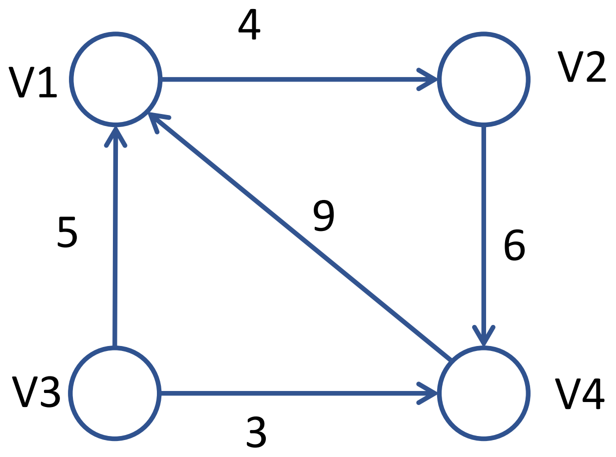




- **Adjacent Vertices**: A vertex $v_i$ is adjacent to $v_j$ , if there is an edge from $v_i$ to $v_j$
- **Self Loop**: If there is an edge whose starting and ending vertices are same → ($v_i$ , $v_i$) is an edge → Self loop
- **Parallel Edges**: If there is more than one edge between the same pair of vertices
- **Simple graph:** A graph if it does not have any self loop or parallel edges → *simple graph*
- **Complete graph:** If each vertex $v_i$ is adjacent to every other vertex $v_j$ → edges from every vertex to all other
- **Acyclic graph**: If there is a path containing *one or more edges* which starts from a vertex $v_i$ and terminates into the same vertex then the path is know *as a cycle* . *If a graph (digraph) does not have cycle then it is called "Acyclic graph"*
- **Isolated vertex:** A vertex is Isolated if there is no edge connected from any other vertex to the vertex
- **Degree of vertex**: The number of edges connected with vertex $v_i$

# Terminology: Graphs (3)

- **Degree of vertex**: The number of edges connected with vertex $v_i$

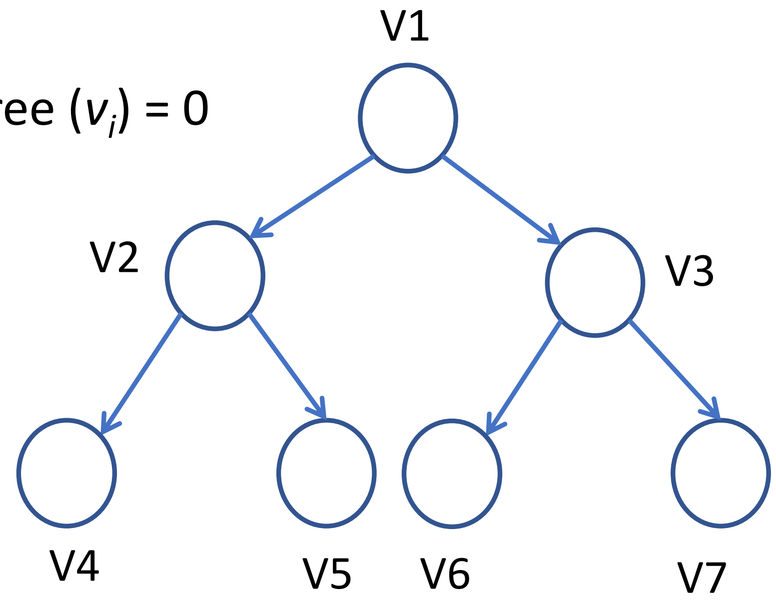



- ✓ *Degree of vertex* → degree ($v_i$)
- ✓ *Indegree ($v_i$)* → number of incoming edges towards $v_i$
- ✓ *Outdegree ($v_i$)* → number of outgoing edges from $v_i$

*Indegree (V1) = 2; Outdegree (V1) = 1*
*Indegree (V2) = 1; Outdegree (V2) = 1*
*Indegree (V3) = 0; Outdegree (V3) = 2*
*Indegree (V4) = 2; Outdegree (V2) = 1*

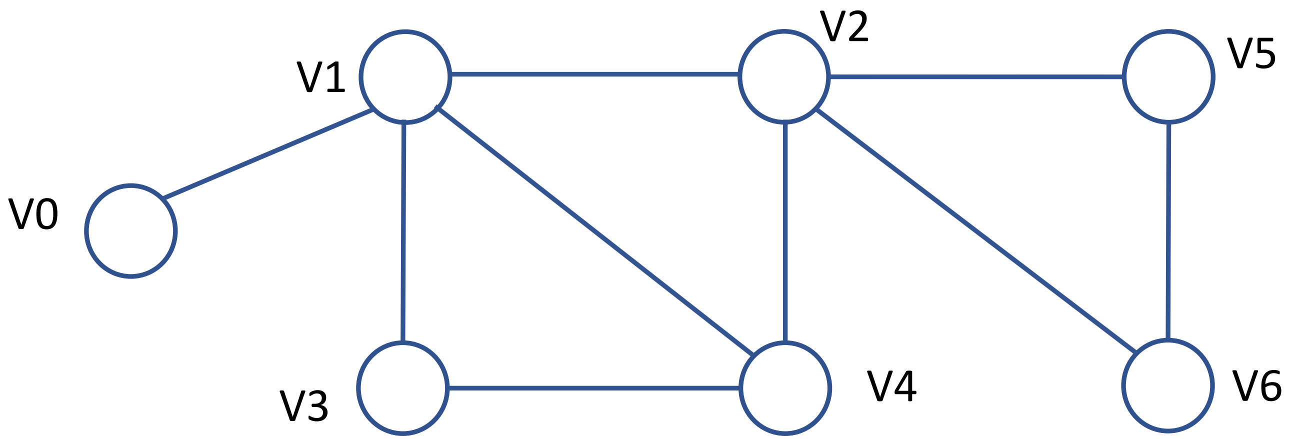
- **Pendant vertex**: A vertex $v_i$ is *pendant* if its indegree ($v_i$) = 1 and outdegree ($v_i$) = 0

- **Connected graph**: In a graph (or digraph), two vertices $v_i$ and $v_j$ are said to be connected if there is a path in G from $v_i$ to $v_j$
  - ✓ A graph is said to be connected if for every pair of distinct vertices $v_i$ and $v_j$, there is a path
  - ✓ A digraph is said to be *strongly connected* if for every pair of distinct vertices, there is a direct path from $v_i$ to $v_j$ and also from $v_j$ to $v_i$

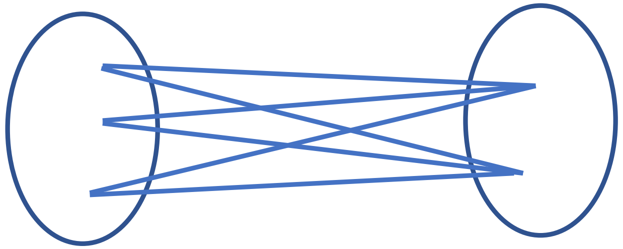


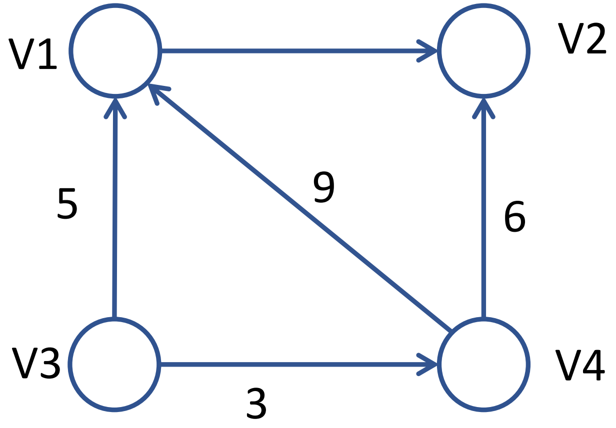
# Terminology: Graphs (4)

- **Path**: A path in a graph is a sequence of edges connecting two vertices



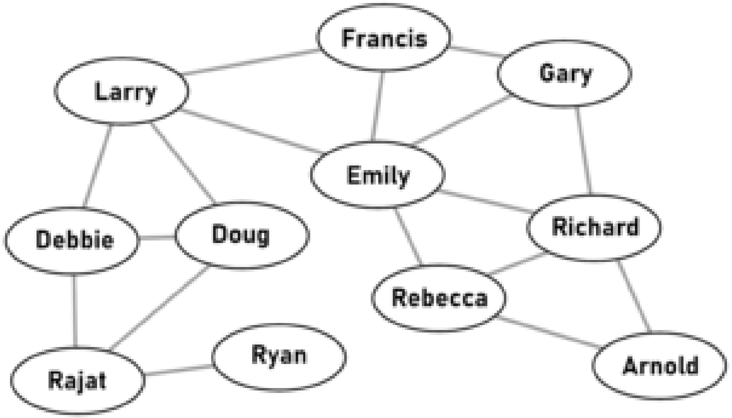

- **Simple Path**: A simple path in a path with no repeated vertices
- **Directed acyclic graph** [DAG]: A direct graph with no cycles
- **Bipartite graph:** A graph whose vertices are divided into two sets such that all the edges connect a vertex in *one set with a vertex in the other set*





# Terminology: Graphs (6)

- **Sparse graph**: A graph with relatively few edge A (< |V| log |V|)
- **Dense graph:** A graph with relatively few of the possible edge are missing is called *dense graph*
- **Six degree of separation:** The notation of six degree of separation presumes the world social network is *a connected graph*




- **Representation of Graphs**
  - ✓ ***Set representation (Adjacency Set)***
  - ✓ ***Linked representation (Adjacency List)***
  - ✓ ***Sequential (Matrix) representation (Adjacency Matrix)***

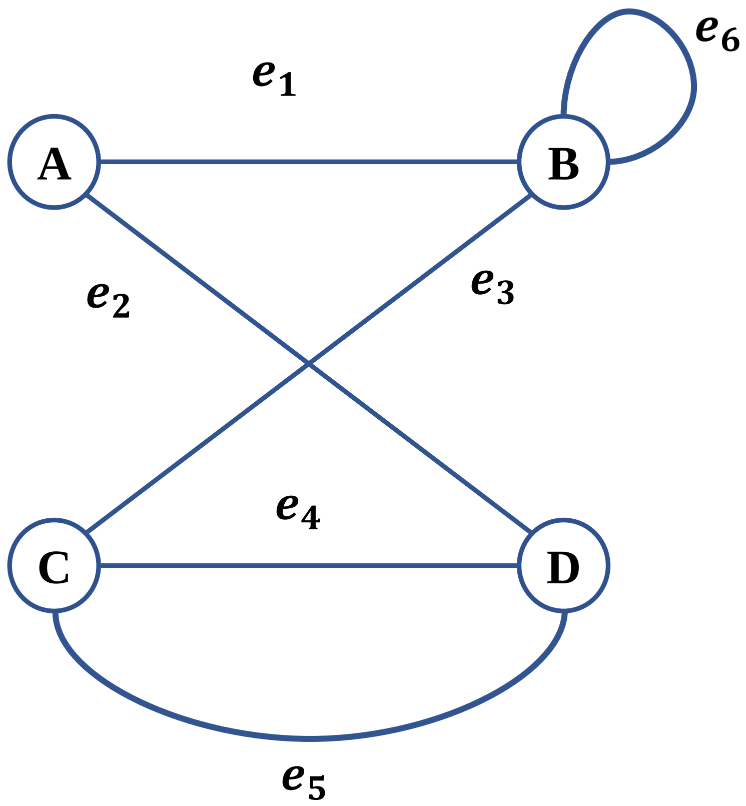
# Exercise: Graphs (1)

Answer the following questions by following Figure!

1. Which of the following graph type is presented?
   a. Undirected  only
   b. Directed and weighted
   c. Undirected and weighted

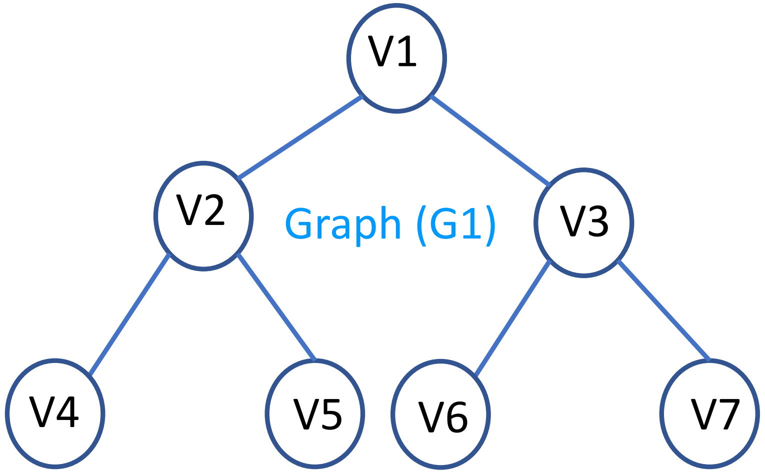
2. Total number of vertices and edges?
   a. 4, 5
   b. 5, 4
   c. 4, 6
   d. 6, 4
3. How many self-loop/s are presented? *1*

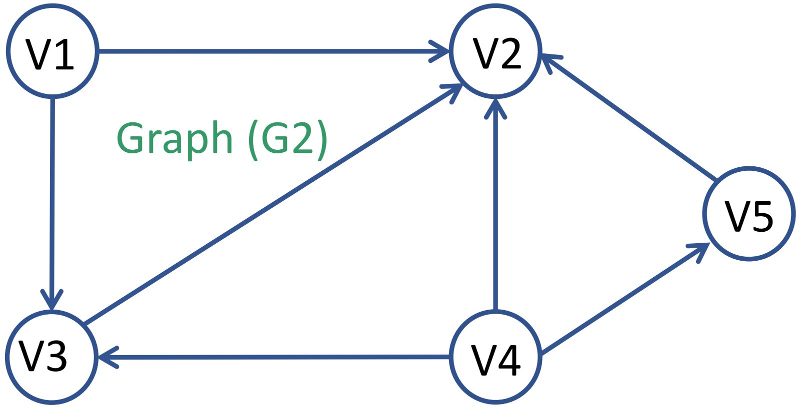
4. Adjacent vertices of  A, B, C, and D respectively
   a. {B, D}, {C, D}, {B, D}, {A, C}
   b. {B, D}, {A,C}, {B, D}, {A, C}

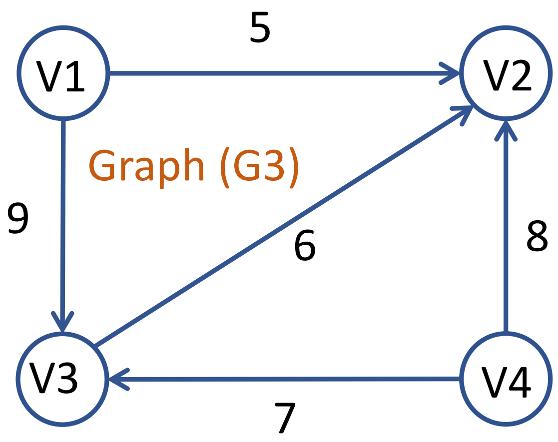


# Representation: Set → Graphs

V → set of vertices; E→ set of edges → V x V (unordered); E → W x V x V for weighted graph




- V(G1) = {V1, V2, V3, V4, V5, V6, V7};
- E(G1) ={(V1, V2), (V1, V3), (V2, V4), (V2, V5), (V3, V6), (V3, V7)}

- V(G2) = {V1, V2, V3, V4, V5};
- E(G2) ={(V1, V2), (V1, V3), (V3, V2), (V4, V2), (V4, V3), (V4, V5), (V5, V2)}

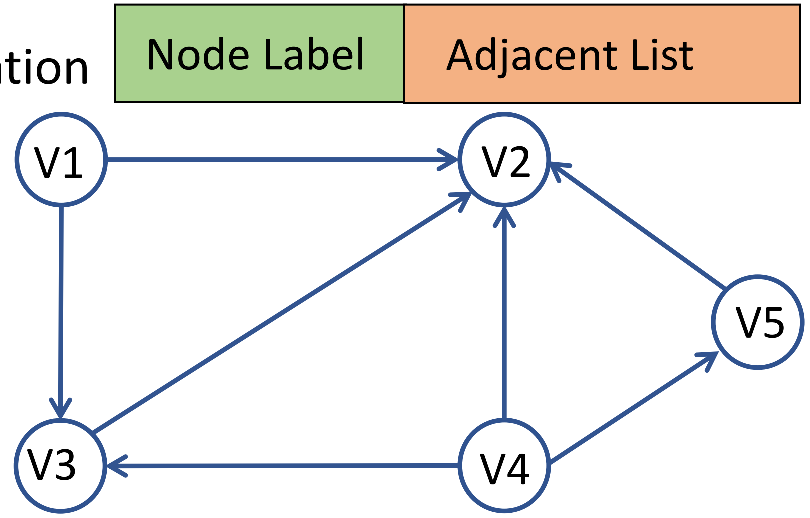
- V(G3) = {V1, V2, V3, V4};
- E(G3) ={(5, V1, V2), (9, V1, V3), (6, V3, V2), (8, V4, V2), (7, V4, V3)}

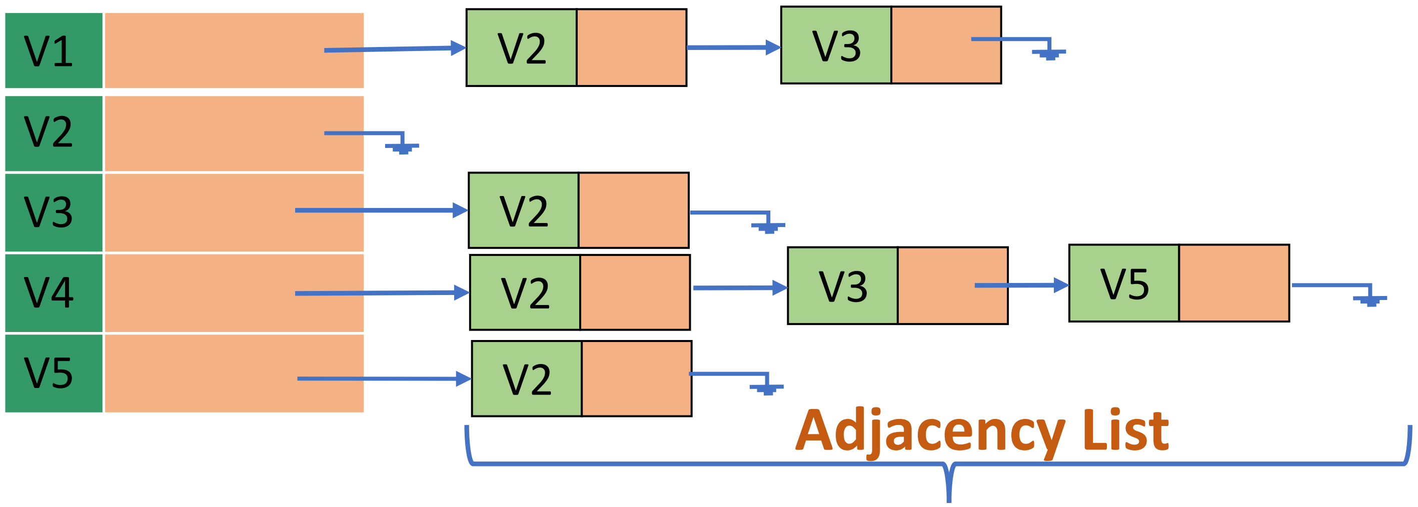
Disadv. → if a graph is multigraph and undirected, this method does not allow to store the parallel edges → As *in sets two identical elements cannot exist; Not useful for manipulation of graph concern*

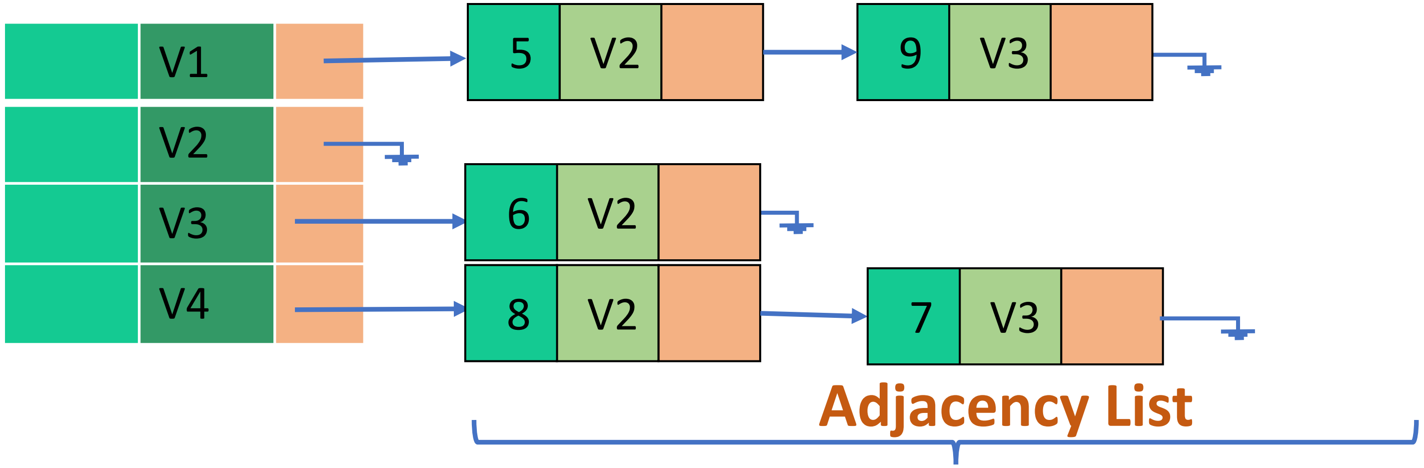
Adv.→ more straightforward representation; most efficient in terms of memory

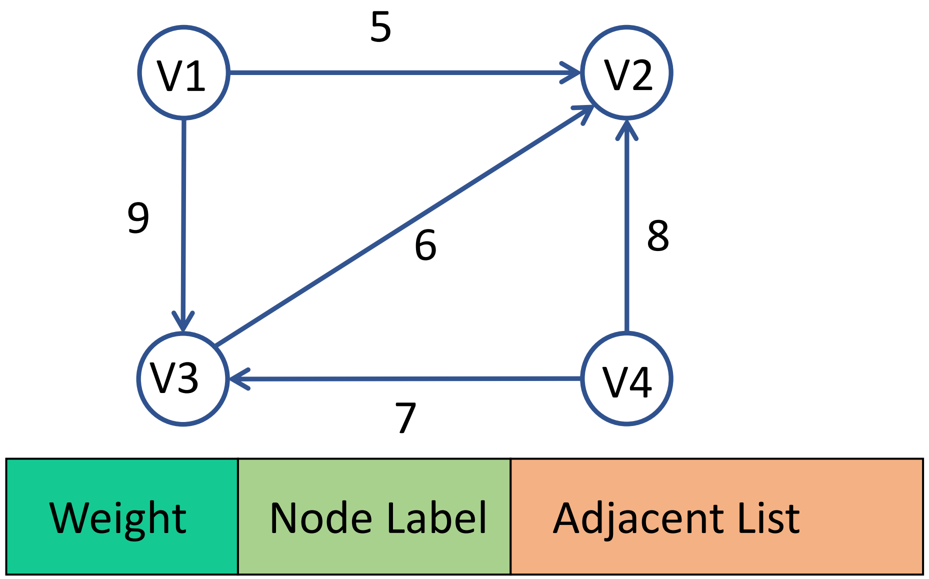
# Representation: Linked → Graphs

❖ Linked representation is another space-saving way of graph representation



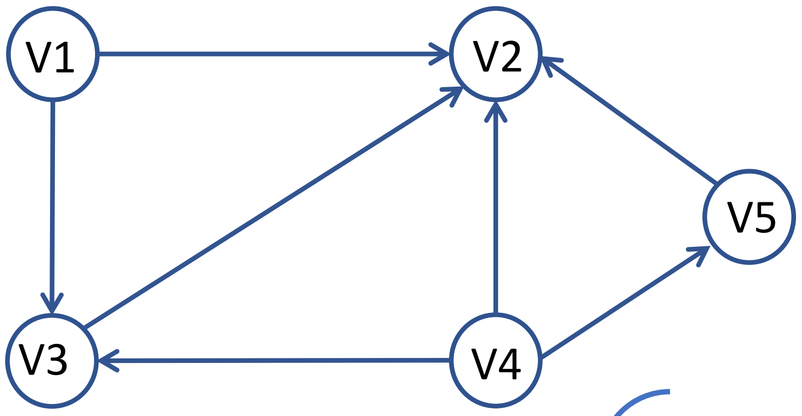
# Representation: Matrix → Graphs

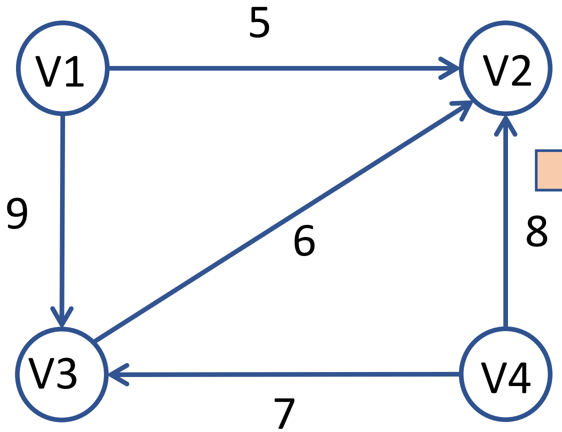
$$
\begin{array}{c|c}
 & \begin{matrix} V1 & V2 & Vi & \cdots & Vn \end{matrix} \\
\begin{matrix} V1 \\ V2 \\ Vi \\ \cdots \\ Vn \end{matrix} & \begin{pmatrix} & & & & \\ & a_{ij} & & & \\ & & & & \\ & & & & \\ & & & & \end{pmatrix}
\end{array}
$$

**Adjacency matrix**

$$
\begin{aligned}
a_{ij} &= 1, if\ there\ is\ an\ edge\ from\ V_i\ to\ V_j \\
&= 0, otherwise
\end{aligned}
$$

$$
\begin{pmatrix}
0 & 1 & 1 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 \\
0 & 1 & 0 & 0 & 0 \\
0 & 1 & 1 & 0 & 1 \\
0 & 1 & 0 & 0 & 0
\end{pmatrix}
$$




$$
\begin{pmatrix}
0 & 5 & 9 & 0 \\
0 & 0 & 0 & 0 \\
0 & 6 & 0 & 0 \\
0 & 8 & 7 & 0
\end{pmatrix}
$$

# Exercise: Graphs (2)

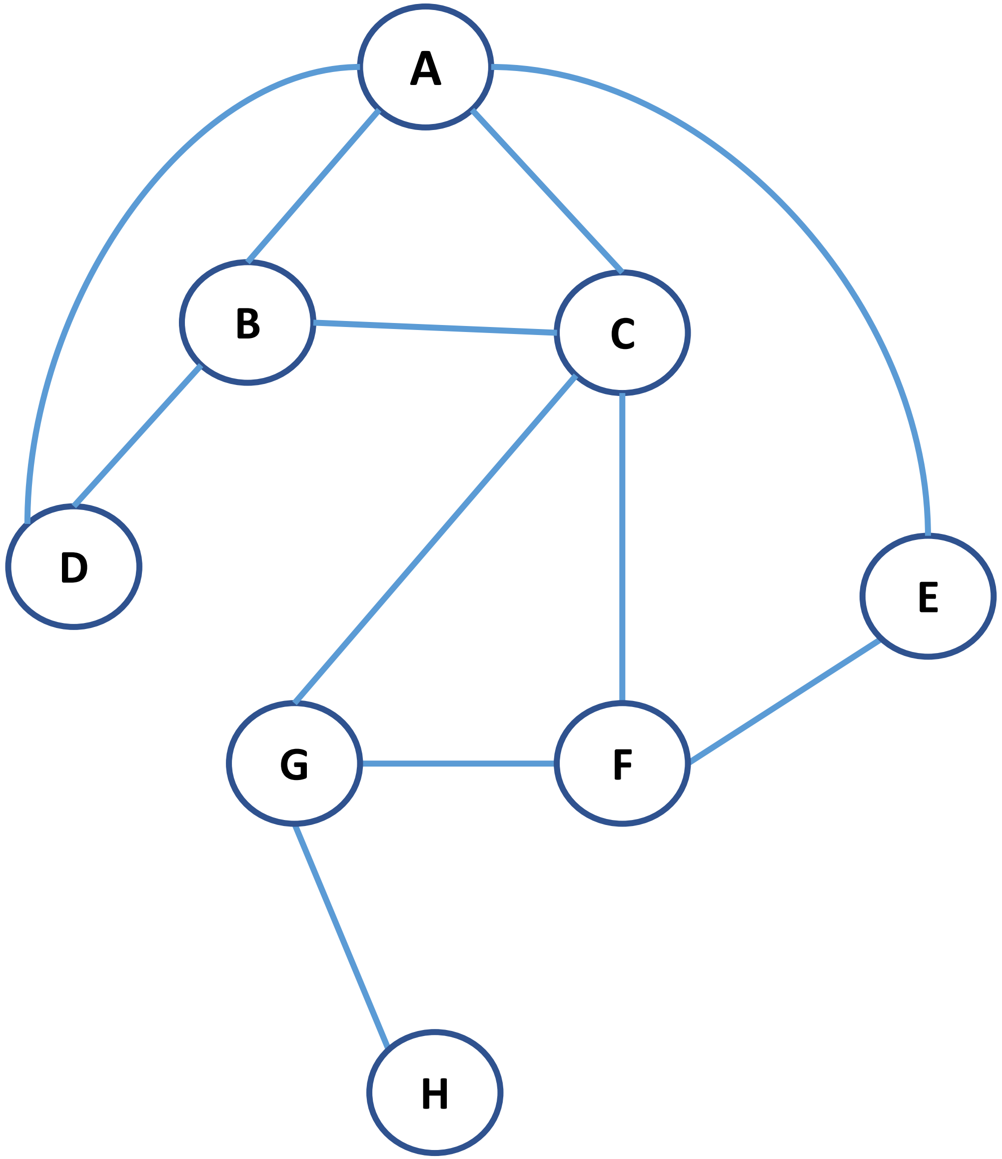
There are 25 telephones in a land. Is it possible to connect them with wires so that each telephone is connected with exactly 7 others?

A. 100
B. 87
C. 70
*D. Not possible*

# Exercise: Graphs (2)

Answer the following questions by following the Figure!

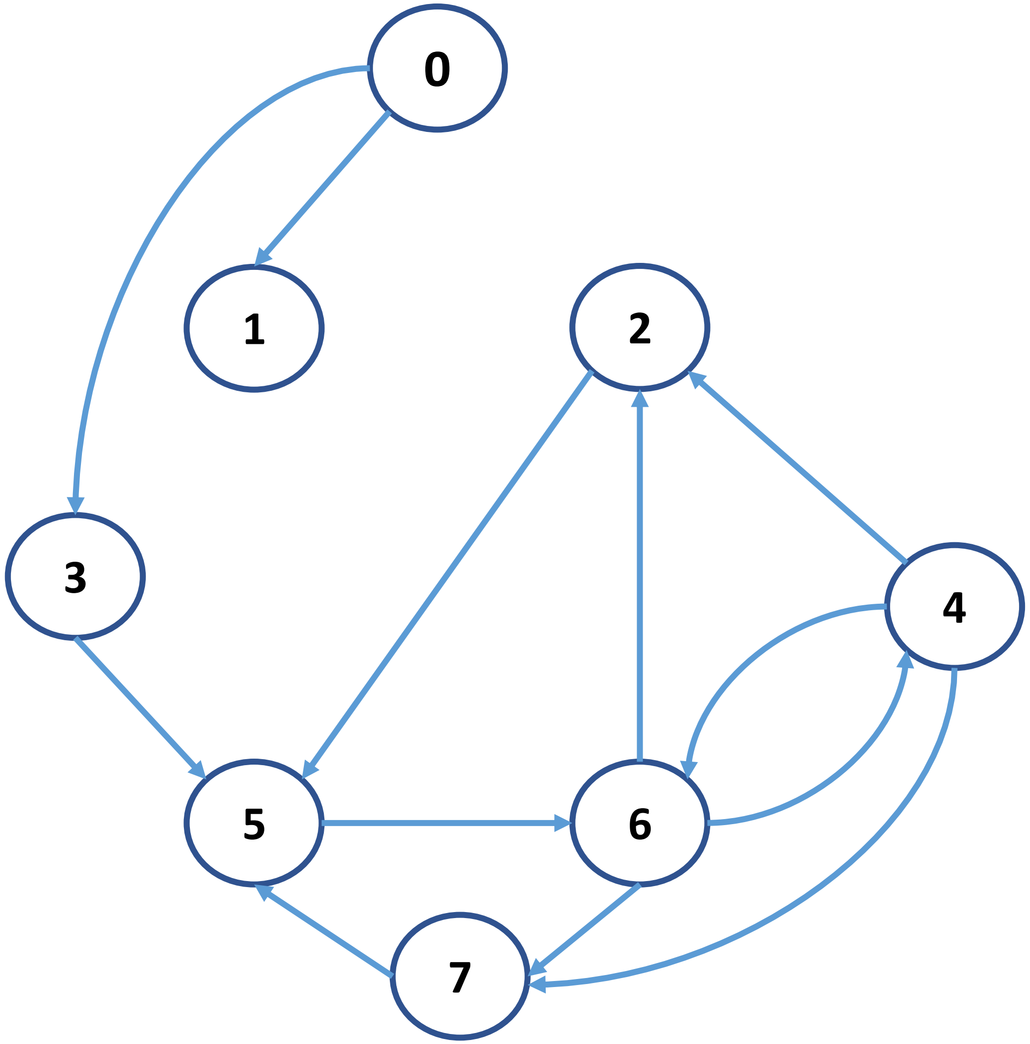
1. Which of the following graph type is presented?
   a. Undirected and dis-connected
   b. Directed and connected
   c. Undirected and connected
2. Total number of vertices and edges?
   a. 10, 8
   b. 11, 8
   c. 8, 10
   d. 8, 11
3. Adjacent vertices of B, C, G, and F respectively
   a. {B, A}, {B, D}, {B, C}, {C, A}, {C, E}, {C, F}, {G, F}, {G, H}, {F, E}
   b. {B, A}, {B, D}, {B, C}, {C, A}, {C, G}, {C, F}, {G, E}, {G, H}, {F, E}
   c. {B, A}, {B, D}, {B, C}, {C, A}, {C, G}, {C, F}, {G, F}, {G, H}, {F, H}
   d. {B, A}, {B, D}, {B, C}, {C, A}, {C, G}, {C, F}, {G, F}, {G, H}, {F, E}



# Exercise: Graphs (3)

Answer the following questions by following the Figure!

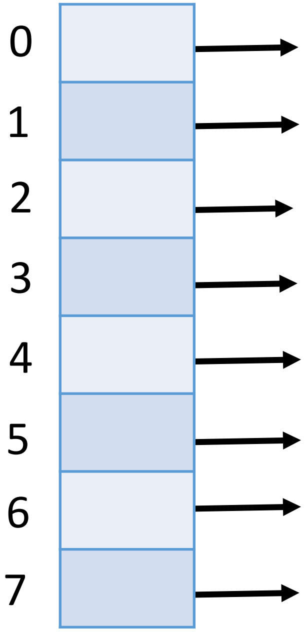
1. Which of the following graph type is presented?
    a. Directed and connected
    b. Directed and dis-connected
2. Total number of vertices and edges?
    a. 12, 8
    b. 10, 8
    c. 8, 12
    d. 8, 10
3. Adjacent vertices of 4, and 6 respectively
    a. {4, 2}, {4,7}, {4, 6}, {6, 5}, {6, 7}, {6, 2}
    b. {4, 2}, {4,7}, {4, 6}, {6, 4}, {6, 5}, {6, 2}
    c. {4, 2}, {4,7}, {4, 5}, {6, 4}, {6, 7}, {6, 2}
    d. {4, 2}, {4,7}, {4, 6}, {6, 4}, {6, 7}, {6, 2}

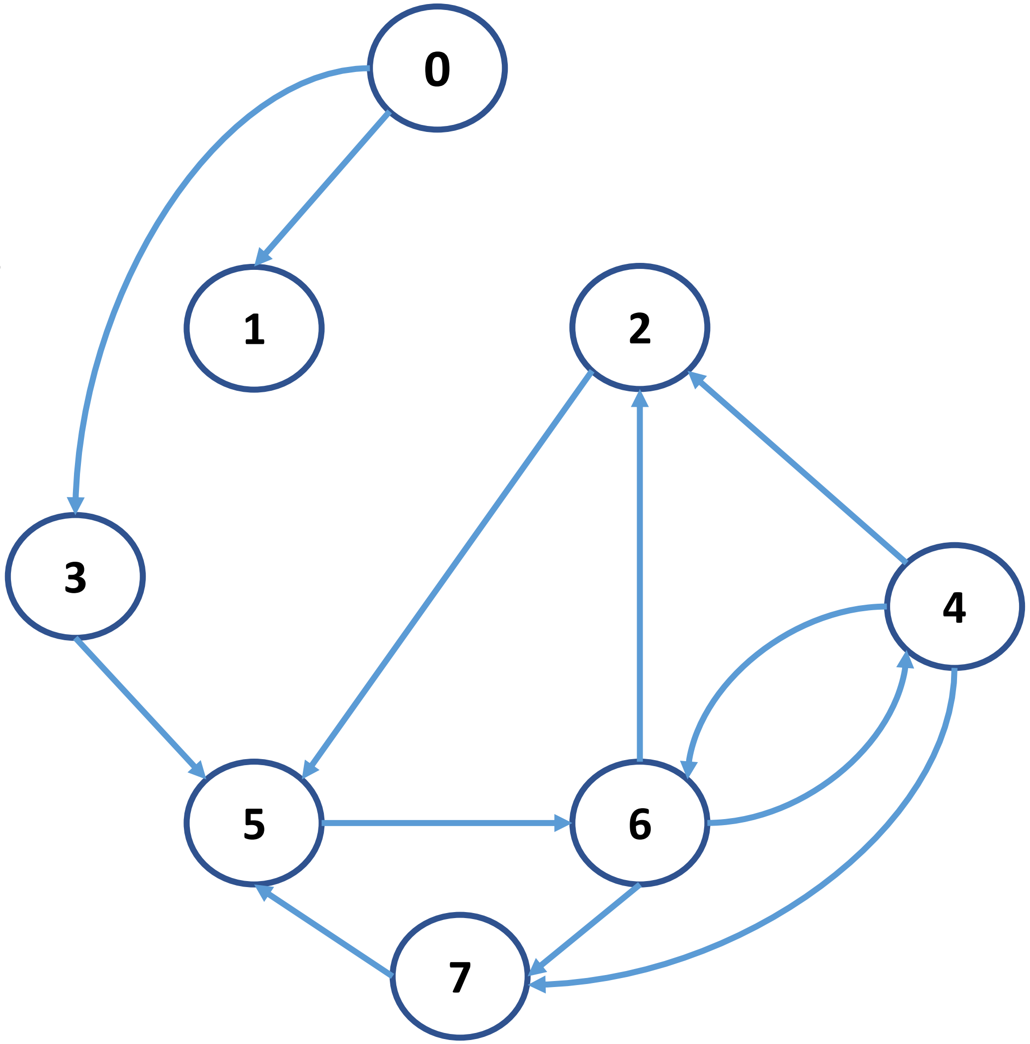


# Exercise: Graphs (4)

Answer the following questions by following the Figure!

1. What is representation of the given graph using Adjacency List?



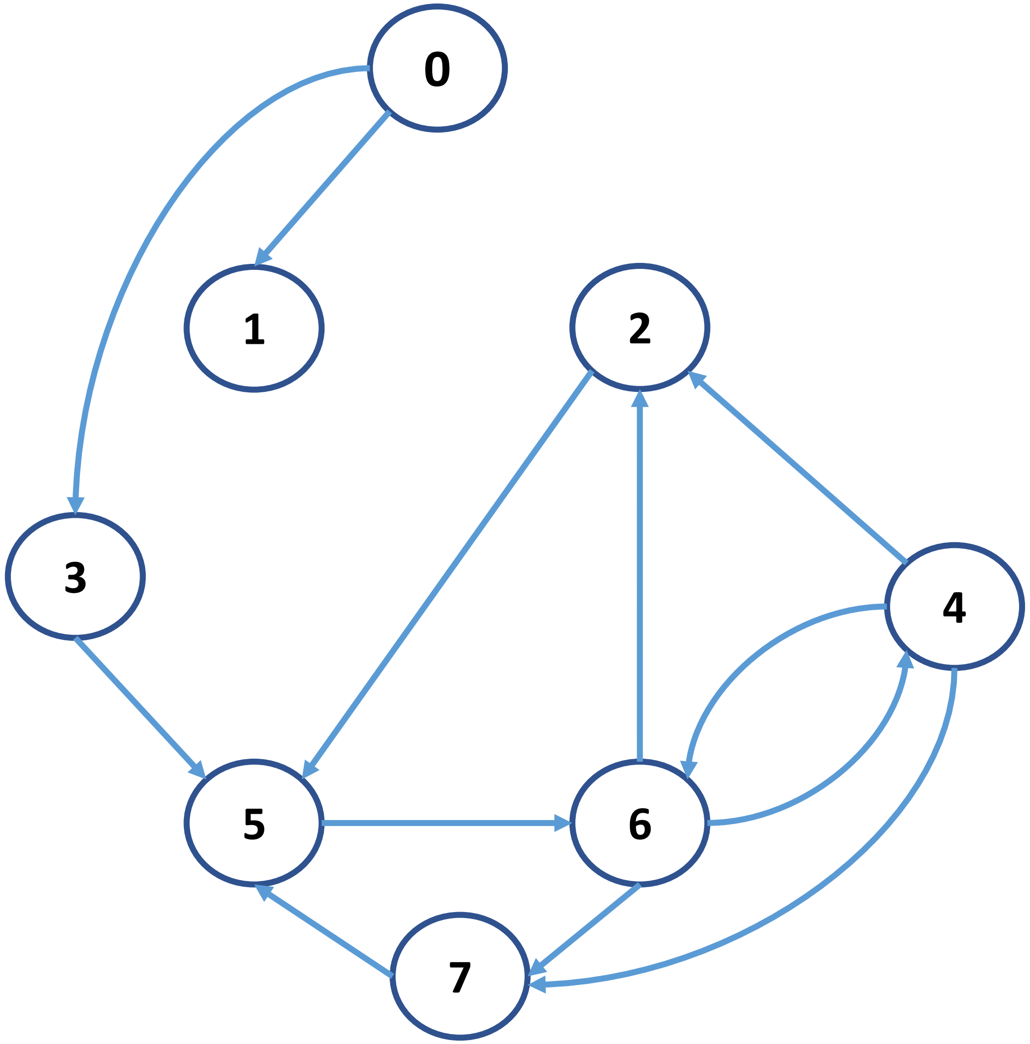



# Exercise: Graphs (5)

Answer the following questions by following the Figure!

What is representation of the given graph using Adjacency Matrix?

| | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 |
|---|---|---|---|---|---|---|---|---|
| 0 | | | | | | | | |
| 1 | | | | | | | | |
| 2 | | | | | | | | |
| 3 | | | | | | | | |
| 4 | | | | | | | | |
| 5 | | | | | | | | |
| 6 | | | | | | | | |
| 7 | | | | | | | | |



# thank you!

email:
k.kondepu@iitdh.ac.in