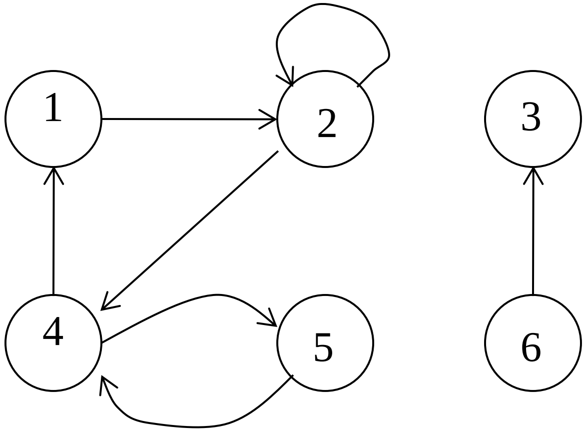
Introduction to Algorithms Lecture 10

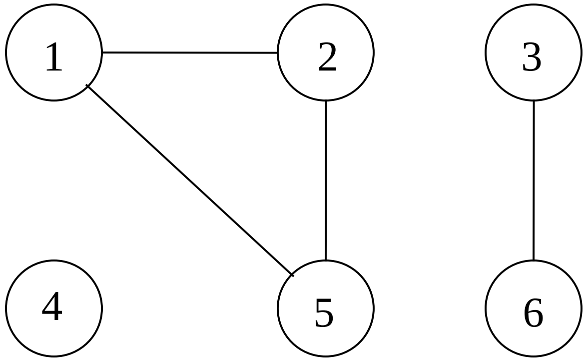
By: Darmen Kariboz

Introduction to Graph Theory:
Basic Concepts

What is a Graph?

- Graph ~ network
- Informally a *graph* is a set of nodes joined by a set of lines or arrows.





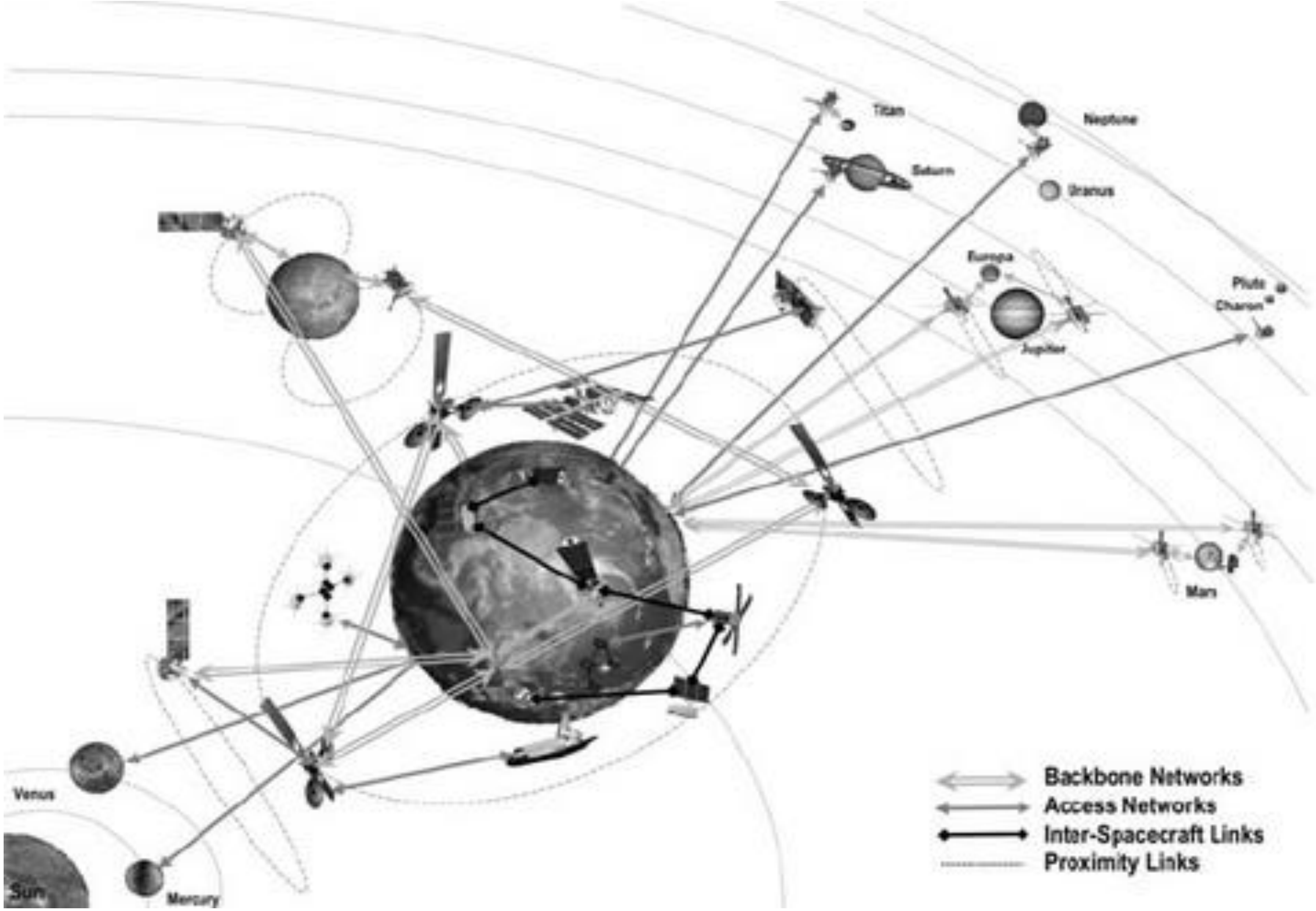
Graph-based representations

- Representing a problem as a graph can provide a different point of view
- Representing a problem as a graph can make a problem much simpler
  - More accurately, it can provide the appropriate tools for solving the problem

Friendship Network

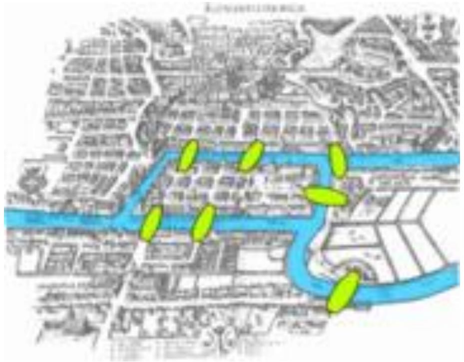
Beshparmak, manty,ehh...

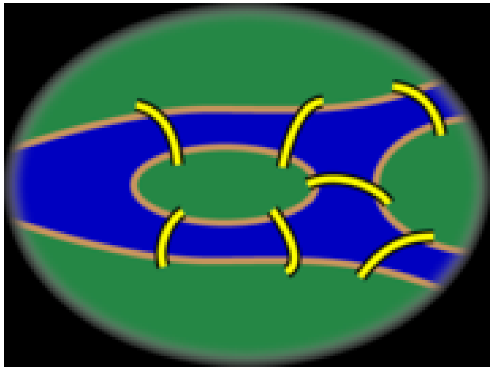
Internet

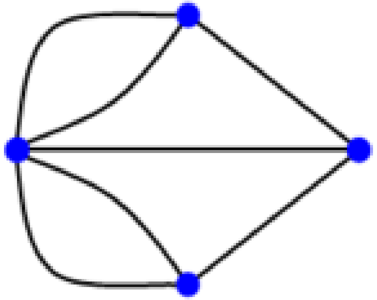

Titan
Neptune
Saturn
Uranus
Europa
Pluto
Charon
Jupiter
Mars
Venus
Sun
Mercury
Backbone Networks
Access Networks
Inter-Spacecraft Links
Proximity Links

Graph Theory - History

Leonhard Euler's paper on "*Seven Bridges of Königsberg*", published in 1736.



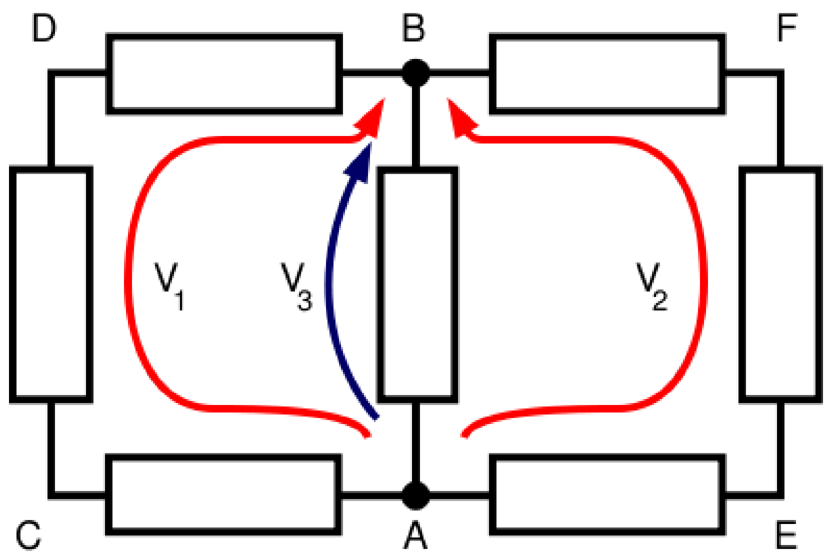




Graph Theory - History

Trees in Electric Circuits

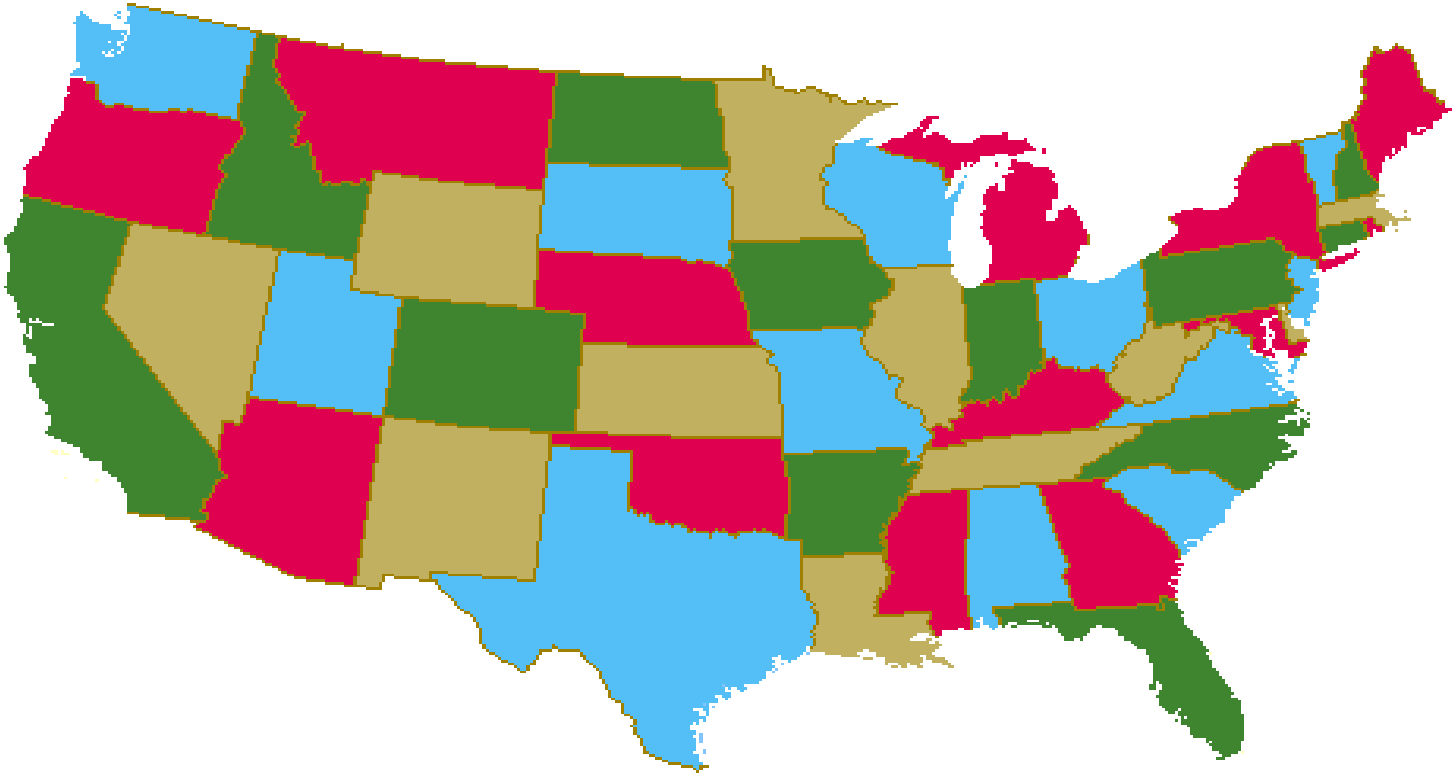
Gustav Kirchhoff



$$V_1 = V_2 = V_3$$

Graph Theory - History

Four Colors of Maps



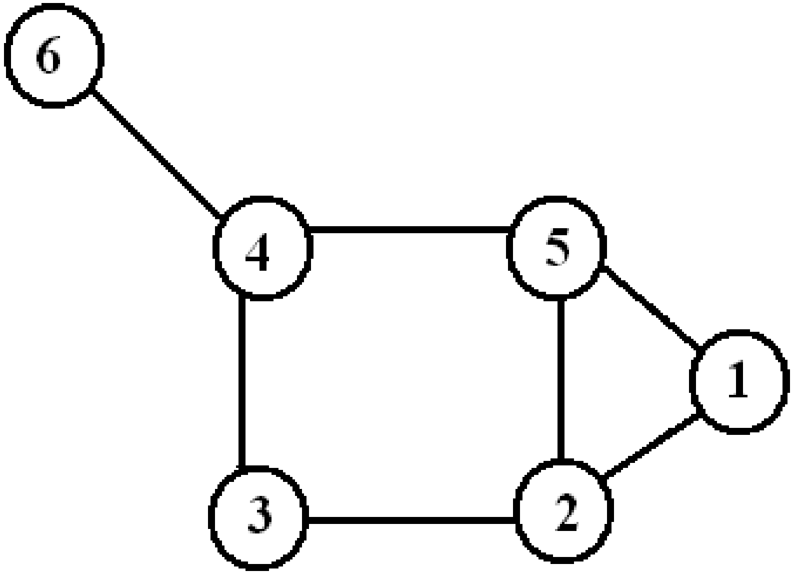
Definition: Graph

- G is an ordered triple G:=(V, E, f)
- V is a set of nodes, points, or vertices.
- E is a set, whose elements are known as edges or lines.
- f is a function
  - maps each element of E
  - to an unordered pair of vertices in V.

Definitions

- Vertex
  - Basic Element
  - Drawn as a *node* or a *dot*.
  - **Vertex set** of *G* is usually denoted by *V*(*G*), or *V*
- Edge
  - A set of two elements
  - Drawn as a line connecting two vertices, called end vertices, or endpoints.
  - The **edge** set of G is usually denoted by E(G), or E.

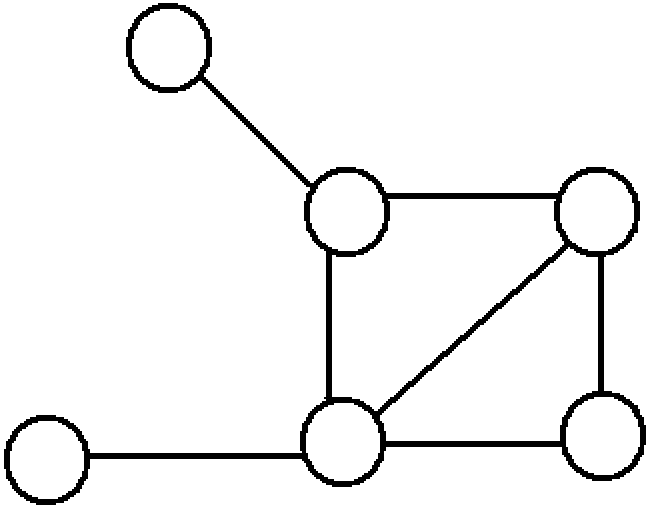
Example

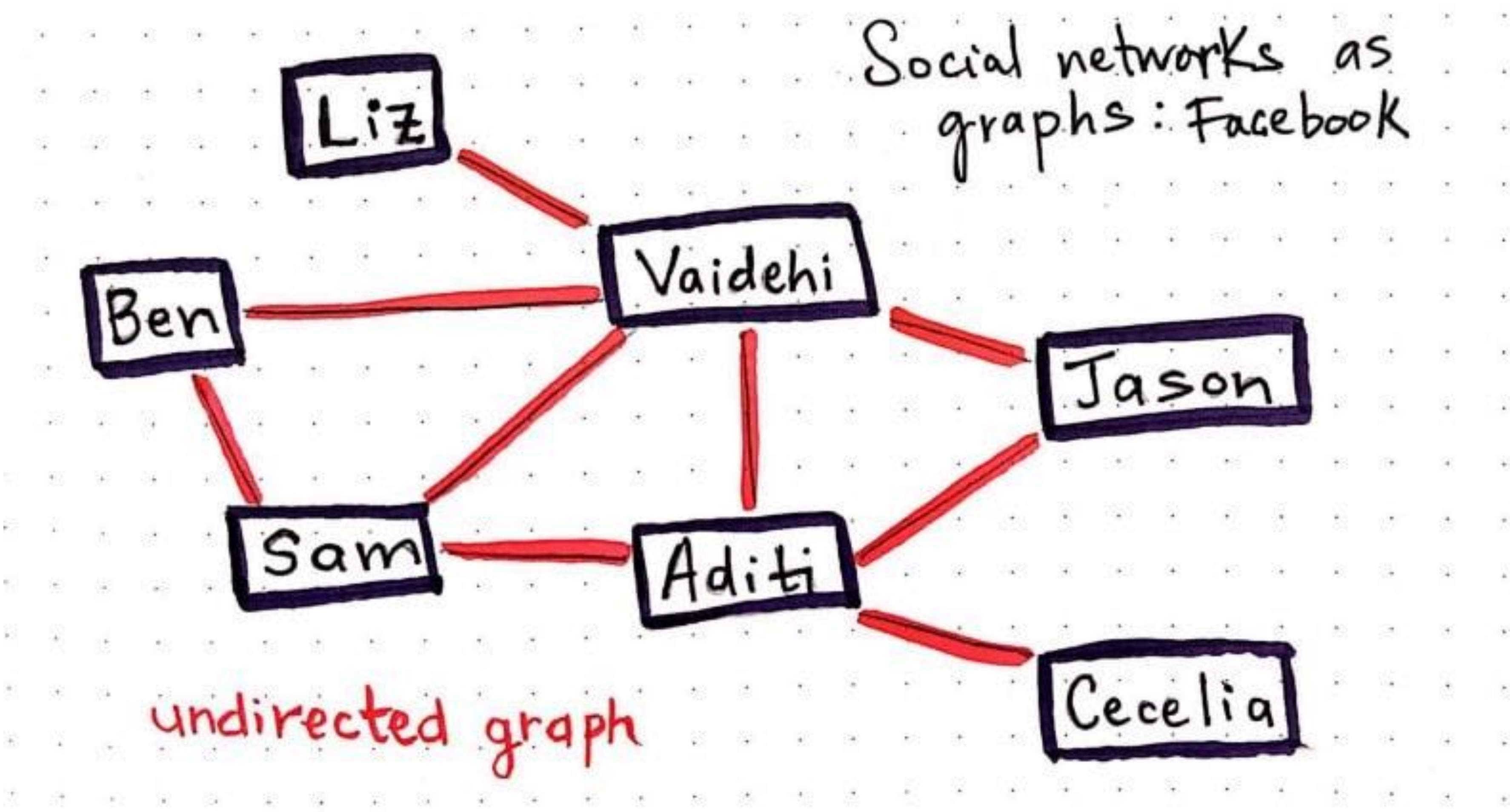


- V:={1,2,3,4,5,6}
- E:={{1,2},{1,5},{2,3},{2,5},{3,4},{4,5},{4,6}}

Simple Graphs

Simple graphs are graphs without multiple edges or self-loops.

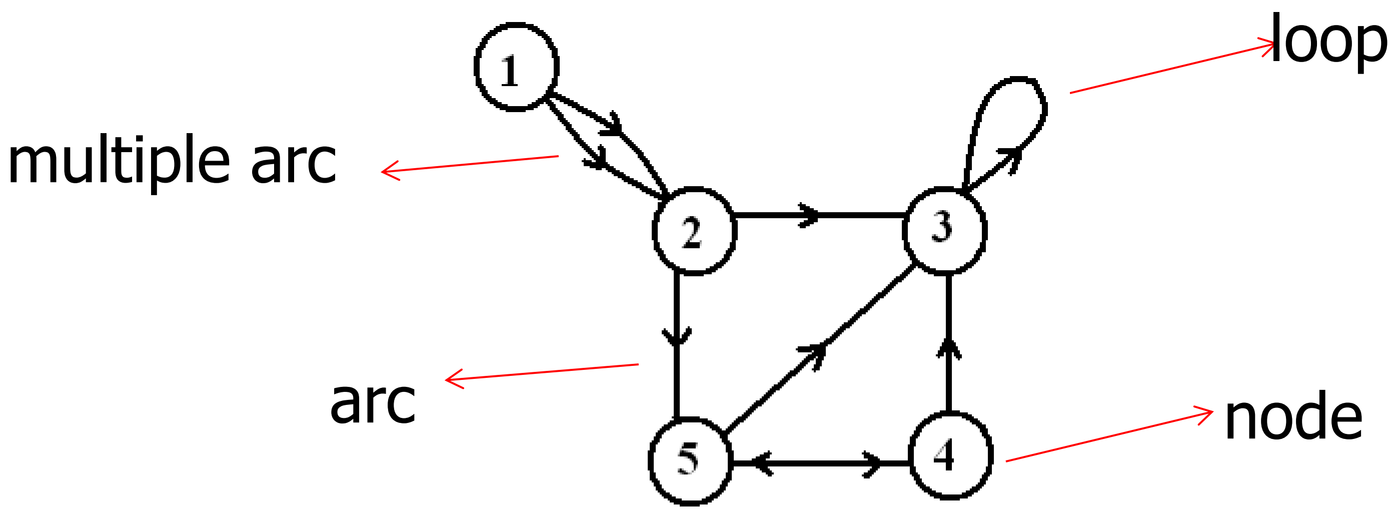


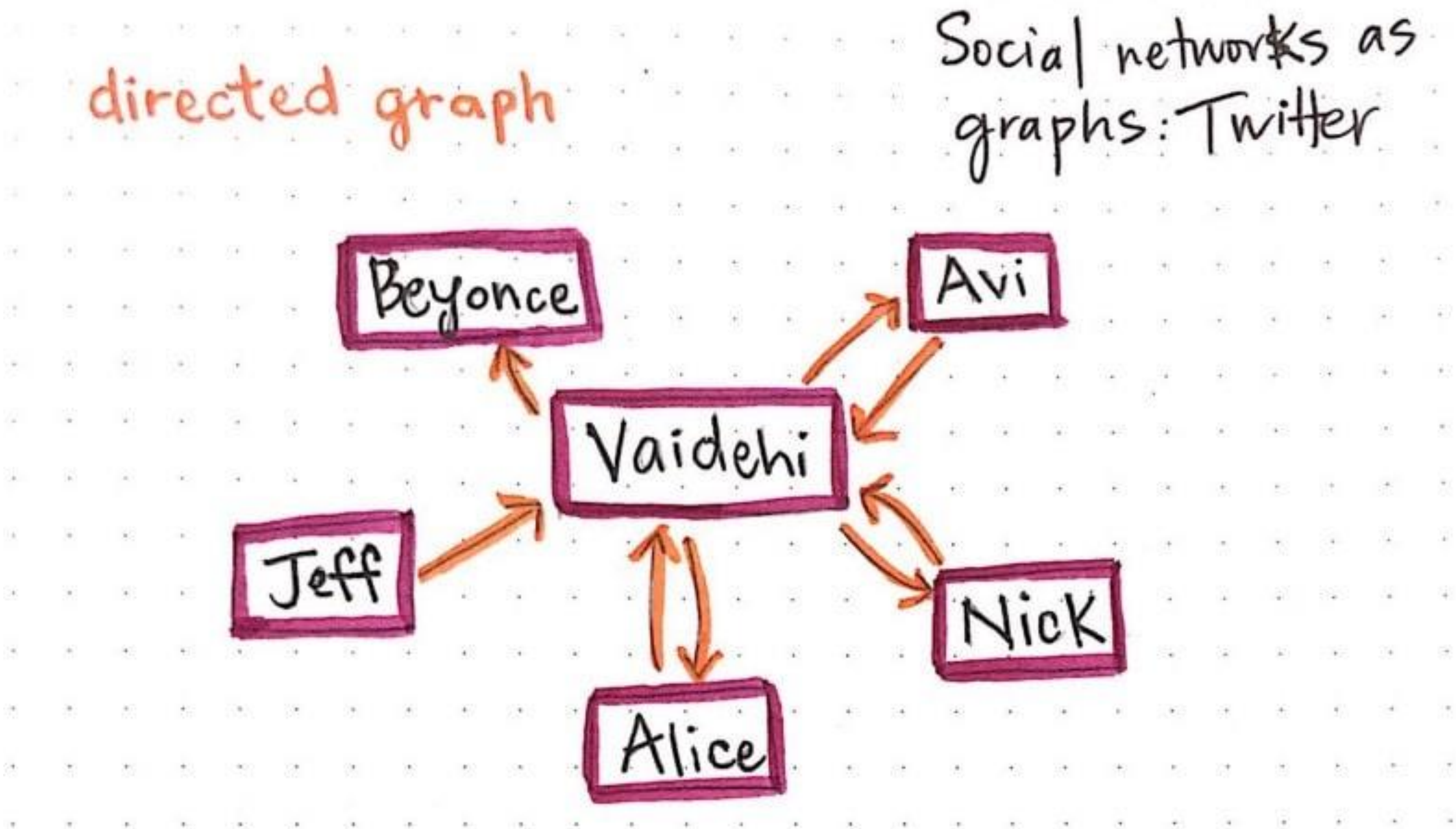


Social networks as graphs: Facebook
Liz
Vaidehi
Ben
Jason
Sam
Aditi
Cecelia
undirected graph

Directed Graph (digraph)

- Edges have directions
- An edge is an *ordered* pair of nodes

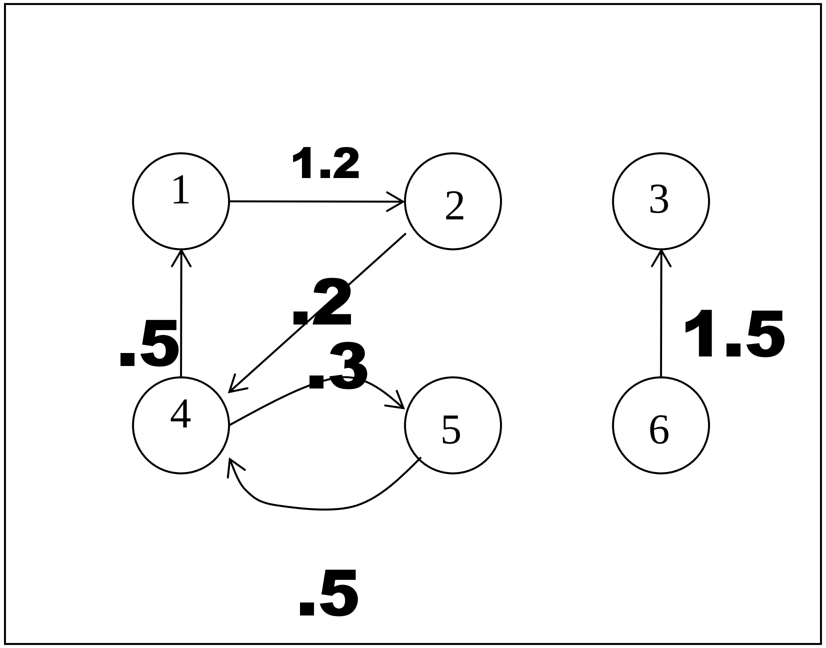


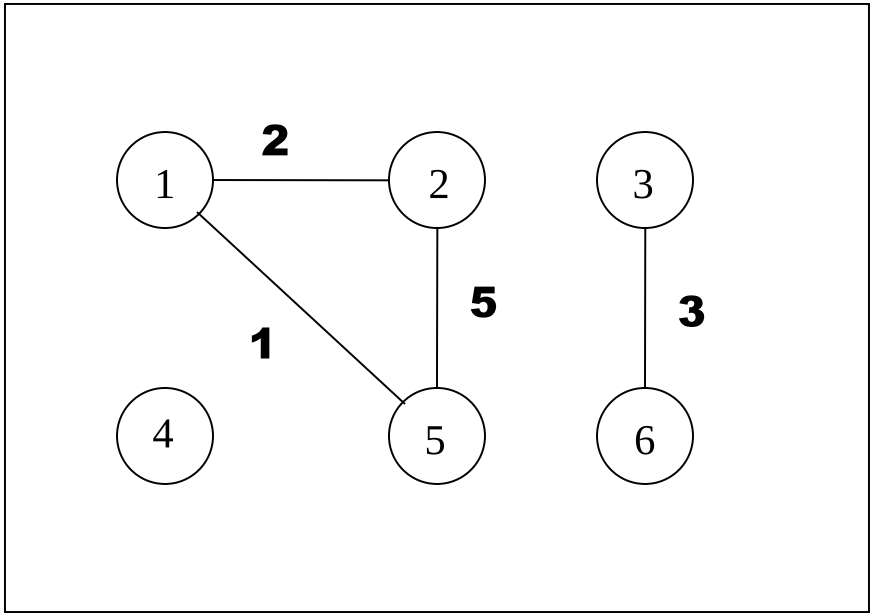


Social networks as graphs: Twitter
directed graph
Beyonce
Avi
Vaidehi
Jeff
Nick
Alice

Weighted graphs

- is a graph for which each edge has an associated ***weight***, usually given by a ***weight function*** $w: E \rightarrow$ **R**.





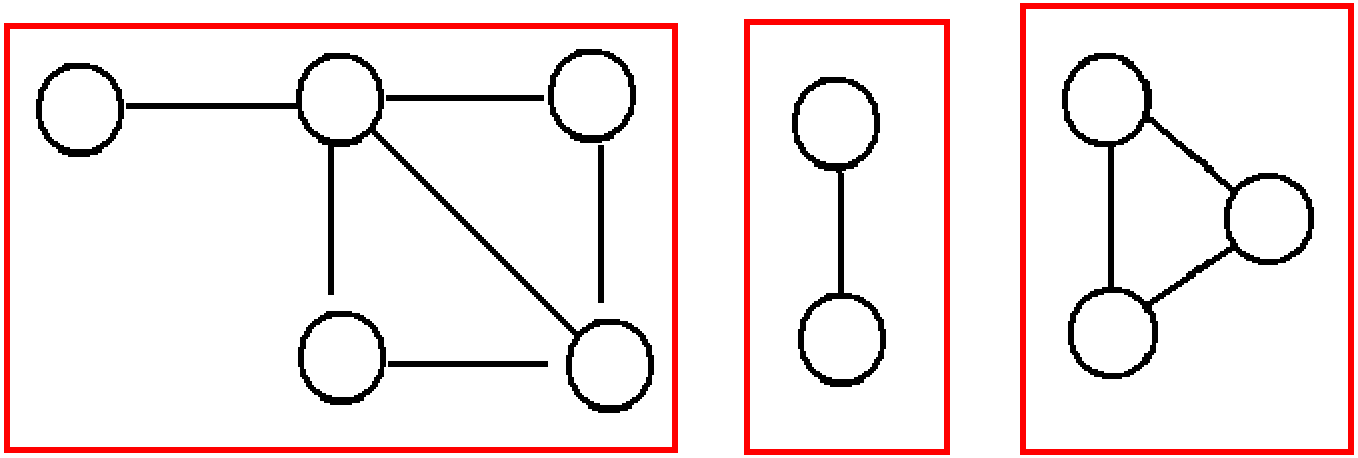
Connectivity

- a graph is ***connected*** if
- you can get from any node to any other by following a sequence of edges OR
- any two nodes are connected by a path.

- A directed graph is ***strongly connected*** if there is a directed path from any node to any other node.

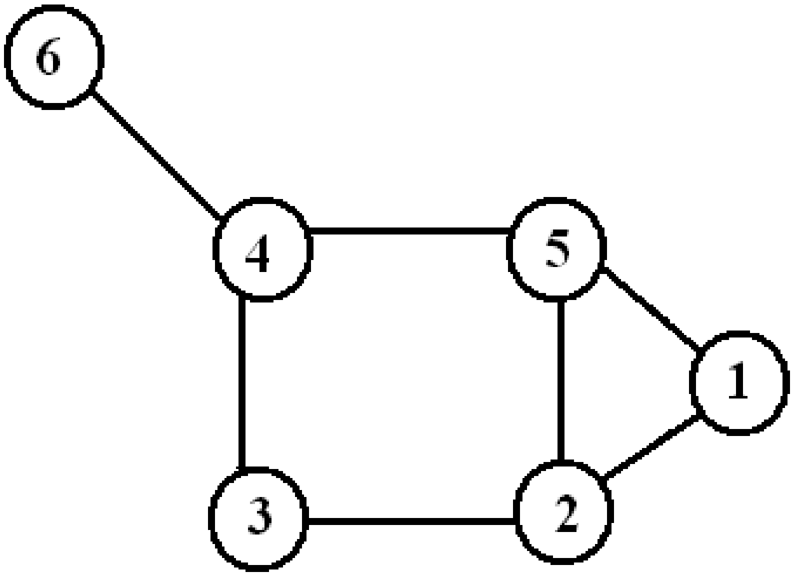
Component

- Every disconnected graph can be split up into a number of connected ***components***.



Degree

- Number of edges incident on a node

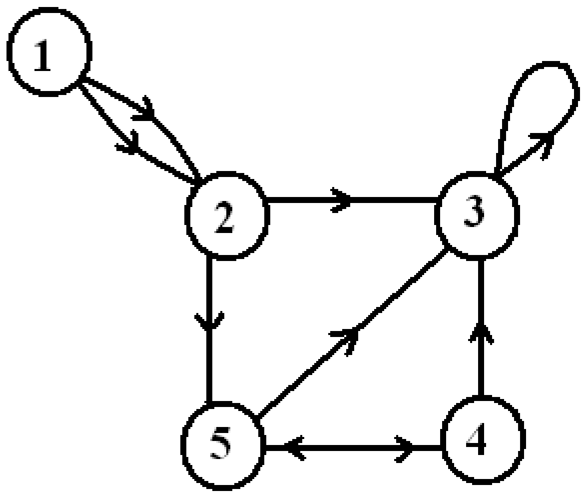


The degree of 5 is 3

Degree (Directed Graphs)

- In-degree: Number of edges entering
- Out-degree: Number of edges leaving

- Degree = indeg + outdeg



outdeg(1)=2
indeg(1)=0

outdeg(2)=2
indeg(2)=2

outdeg(3)=1
indeg(3)=4

Walks

A ***walk of length k*** in a graph is a succession of k (not necessarily different) edges of the form

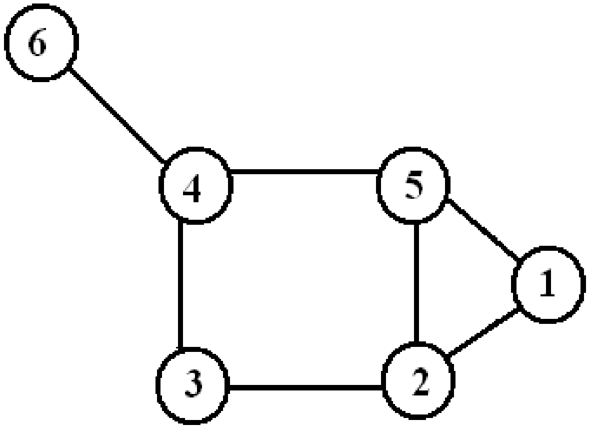
uv,vw,wx,…,yz.

This walk is denote by uvwx…xz, and is referred to as a ***walk between u and z***.

A walk is ***closed*** if u=z.

Path

- A *path* is a walk in which all the edges and all the nodes are different.

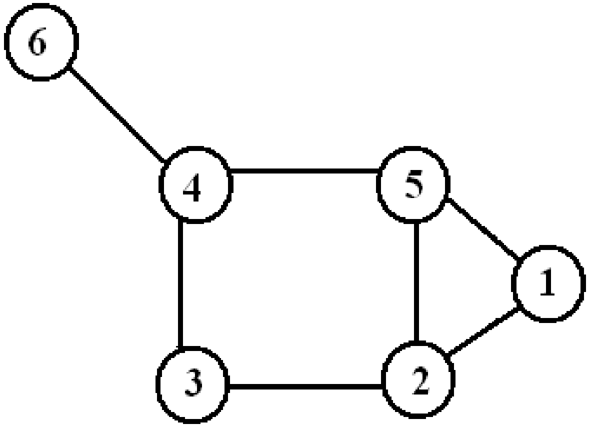


Walks and Paths

1,2,5,2,3,4	1,2,5,2,3,2,1	1,2,3,4,6
walk of length 5	CW of length 6	path of length 4

Cycle

- A ***cycle*** is a closed walk in which all the edges are different.

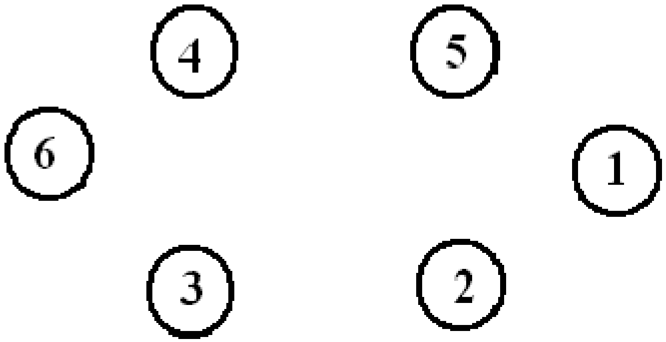


1,2,5,1 2,3,4,5,2

3-cycle 4-cycle

Special Types of Graphs

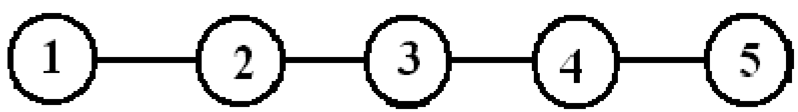
- Empty Graph / Edgeless graph
  - No edge

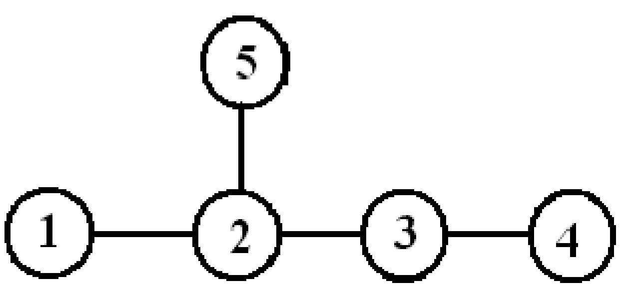


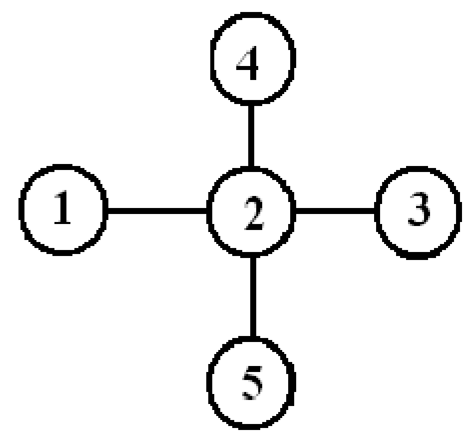
- Null graph
  - No nodes
  - Obviously no edge

Trees

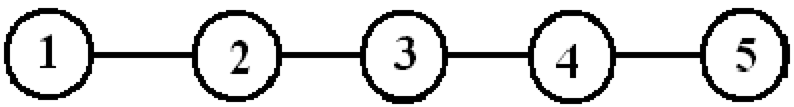
- Connected Graph without cycles
- Two nodes have *exactly* one path between them



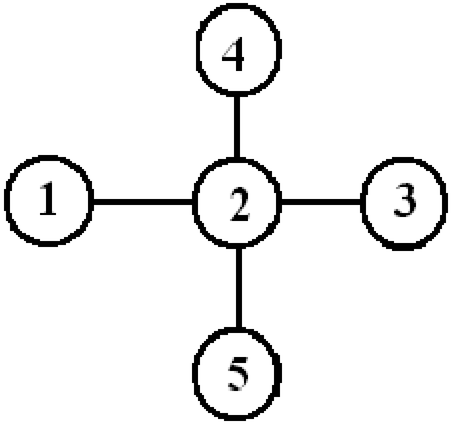




Special Trees



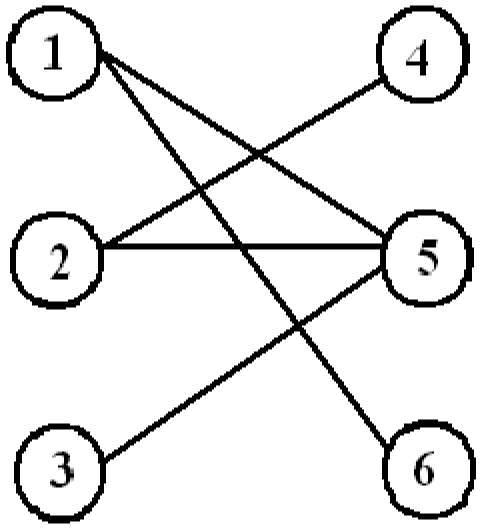
Paths



Stars

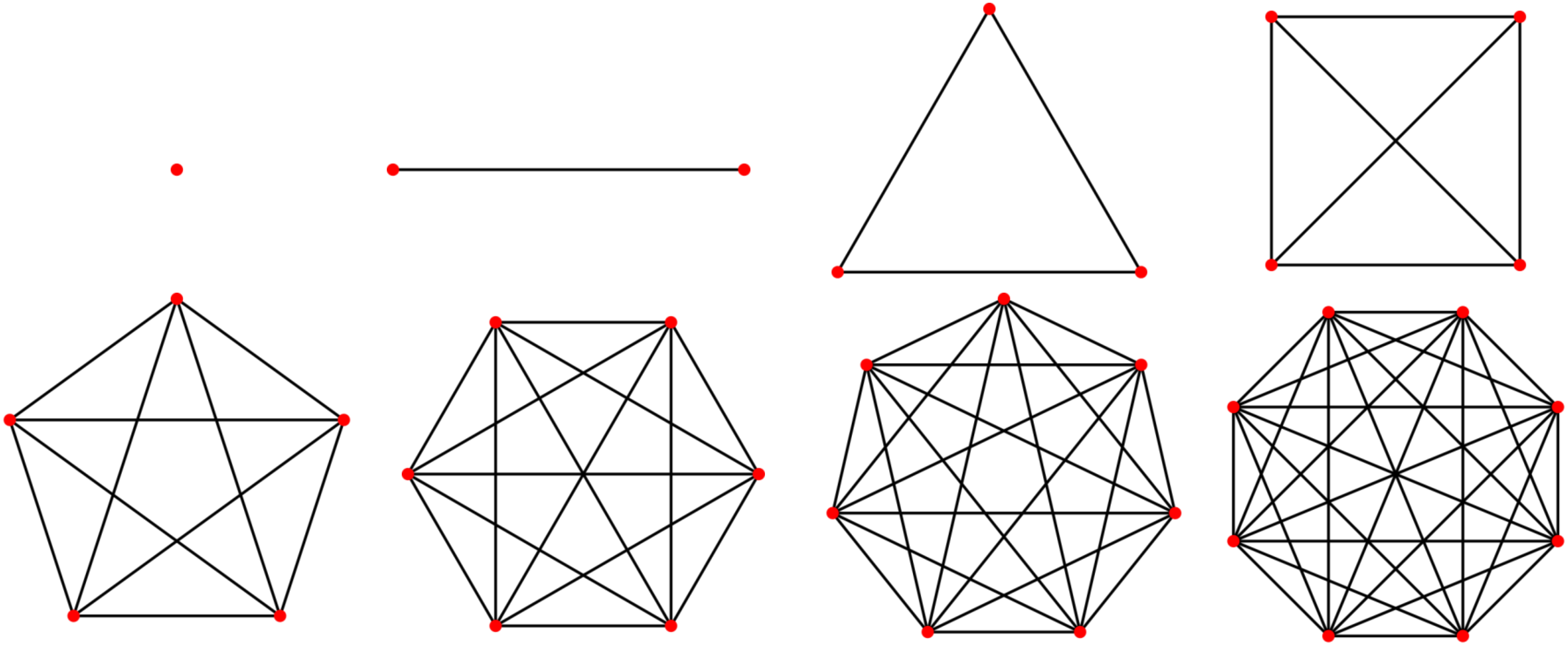
Bipartite graph

- V can be partitioned into 2 sets V_1 and V_2 such that $(u,v) \in E$ implies
  - either $u \in V_1$ and $v \in V_2$
  - OR $v \in V_1$ and $u \in V_2$.



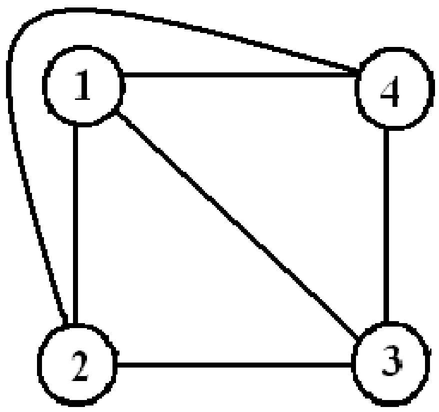
Complete Graph

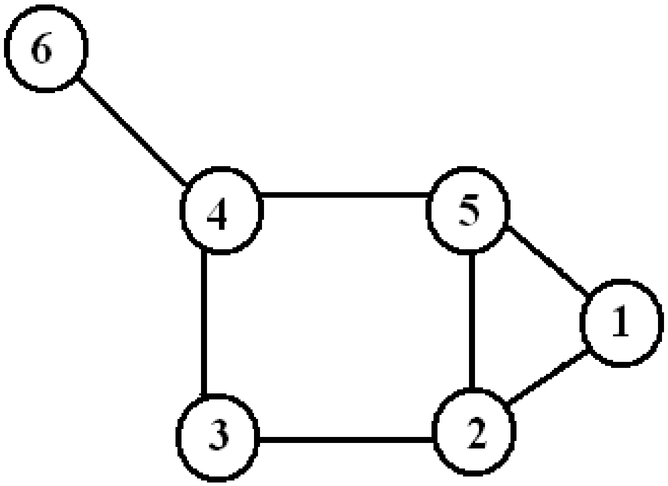
- Every pair of vertices are adjacent
- Has n(n-1)/2 edges



Planar Graphs

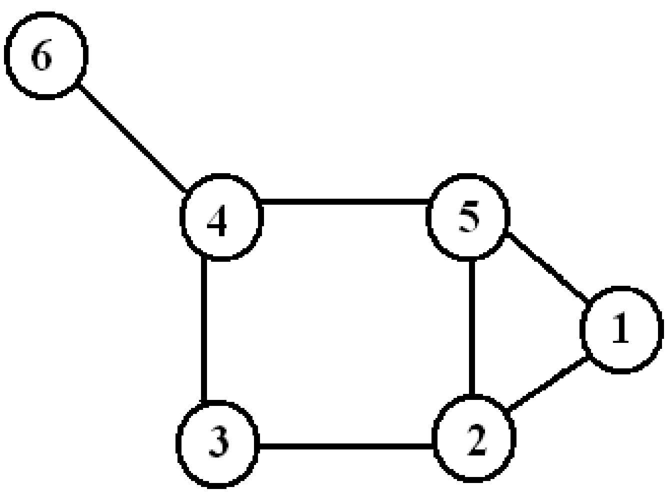
- Can be drawn on a plane such that no two edges intersect
- K_4 is the largest complete graph that is planar

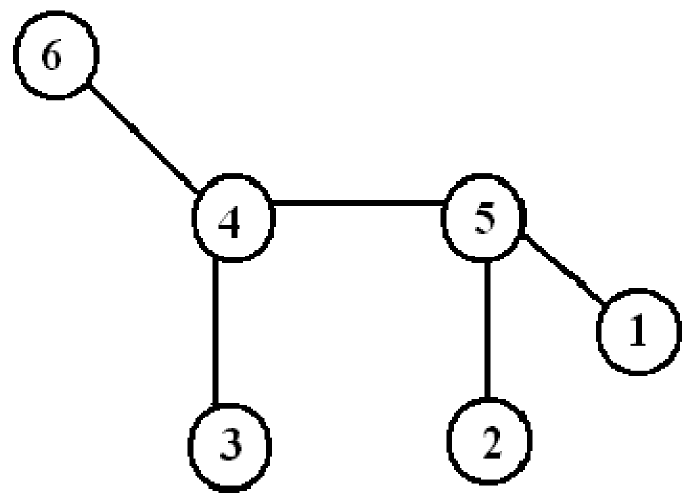




Spanning tree

- Let G be a connected graph. Then a ***spanning tree*** in G is a subgraph of G that includes every node and is also a tree.

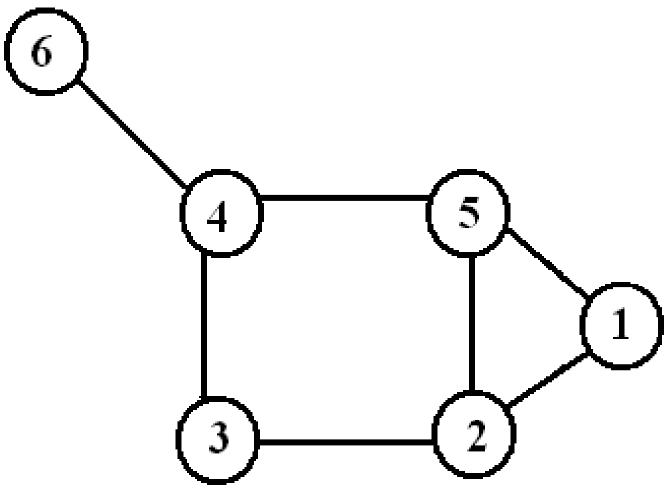




Representation (Matrix)

- Incidence Matrix
- V x E
- [vertex, edges] contains the edge's data
- Adjacency Matrix
- V x V
- Boolean values (adjacent or not)
- Or Edge Weights

Matrices



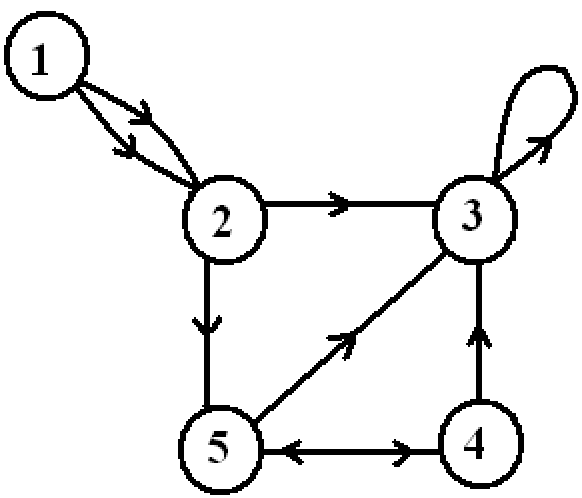
$$
\begin{array}{c}
\\
\begin{array}{c} 1 \\ 2 \\ 3 \\ 4 \\ 5 \\ 6 \end{array}
\end{array}
\begin{array}{c}
\begin{array}{ccccccc} 1,2 & 1,5 & 2,3 & 2,5 & 3,4 & 4,5 & 4,6 \end{array} \\
\begin{pmatrix}
1 & 1 & 0 & 0 & 0 & 0 & 0 \\
1 & 0 & 1 & 1 & 0 & 0 & 0 \\
0 & 0 & 1 & 0 & 1 & 0 & 0 \\
0 & 0 & 0 & 0 & 1 & 1 & 1 \\
0 & 1 & 0 & 1 & 0 & 1 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 1
\end{pmatrix}
\end{array}
$$

$$
\begin{array}{c}
\\
\begin{array}{c} 1 \\ 2 \\ 3 \\ 4 \\ 5 \\ 6 \end{array}
\end{array}
\begin{array}{c}
\begin{array}{cccccc} 1 & 2 & 3 & 4 & 5 & 6 \end{array} \\
\begin{pmatrix}
0 & 1 & 0 & 0 & 1 & 0 \\
1 & 0 & 1 & 0 & 1 & 0 \\
0 & 1 & 0 & 1 & 0 & 0 \\
0 & 0 & 1 & 0 & 1 & 1 \\
1 & 1 & 0 & 1 & 0 & 0 \\
0 & 0 & 0 & 1 & 0 & 0
\end{pmatrix}
\end{array}
$$

Representation (List)

- Edge List
  - pairs (ordered if directed) of vertices
  - Optionally weight and other data
- Adjacency List (node list)

Edge and Node Lists



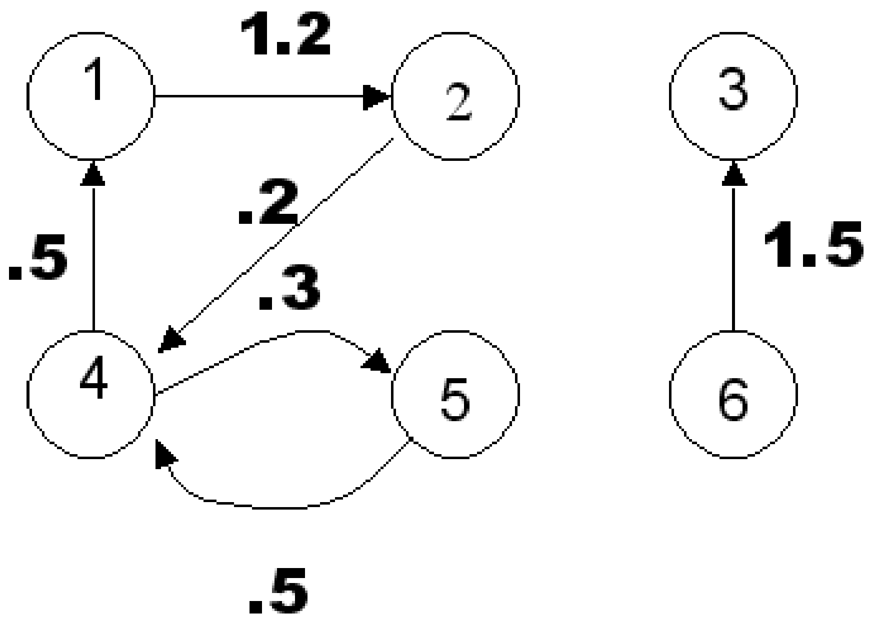
Edge List

1 2
1 2
2 3
2 5
3 3
4 3
4 5
5 3
5 4

Node List

1 2 2
2 3 5
3 3
4 3 5
5 3 4

Edge Lists for Weighted Graphs



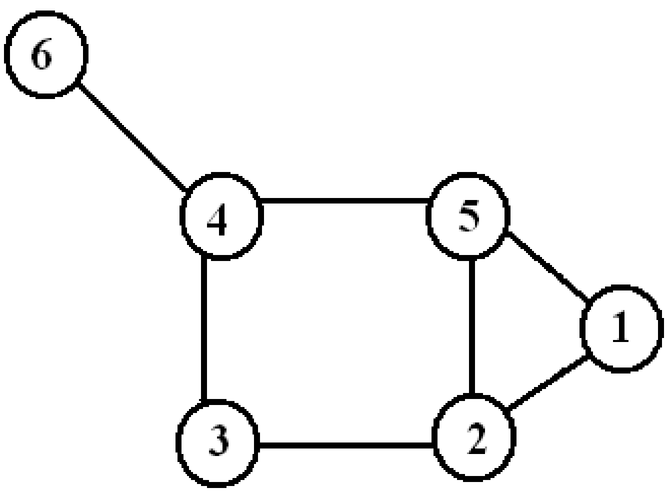
Edge List

1 2 1.2
2 4 0.2
4 5 0.3
4 1 0.5
5 4 0.5
6 3 1.5

Distance Matrix

- $|V| \times |V|$ matrix $D = (d_{ij})$ such that

d_{ij} is the topological distance between i and j.

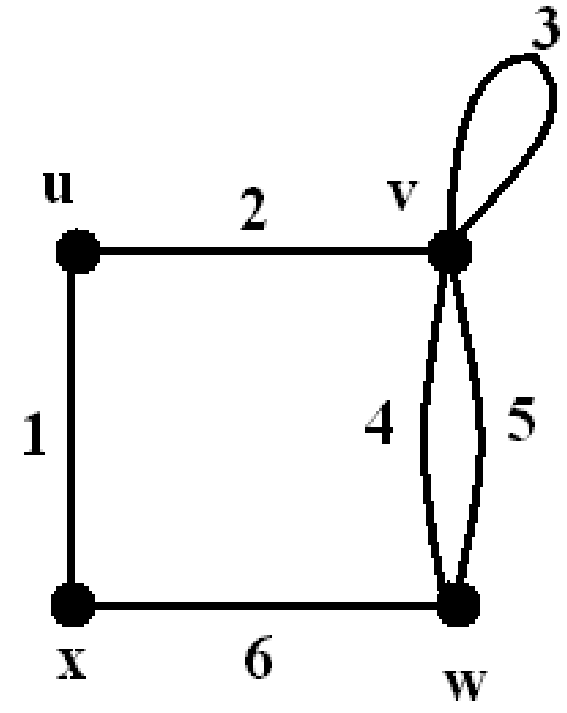


$$\begin{array}{c c} & \begin{array}{cccccc} 1 & 2 & 3 & 4 & 5 & 6 \end{array} \\ \begin{array}{c} 1 \\ 2 \\ 3 \\ 4 \\ 5 \\ 6 \end{array} & \begin{pmatrix} 0 & 1 & 2 & 2 & 1 & 3 \\ 1 & 0 & 1 & 2 & 1 & 3 \\ 2 & 1 & 0 & 1 & 2 & 2 \\ 2 & 2 & 1 & 0 & 1 & 1 \\ 1 & 1 & 2 & 1 & 0 & 2 \\ 3 & 3 & 2 & 1 & 2 & 0 \end{pmatrix} \end{array}$$

Exercise 1

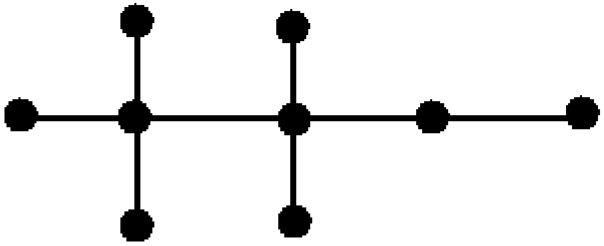
Which of the following statements hold for this graph?

(a)nodes v and w are adjacent;
(b)nodes v and x are adjacent;
(c)node u is incident with edge 2;
(d)Edge 5 is incident with node x.

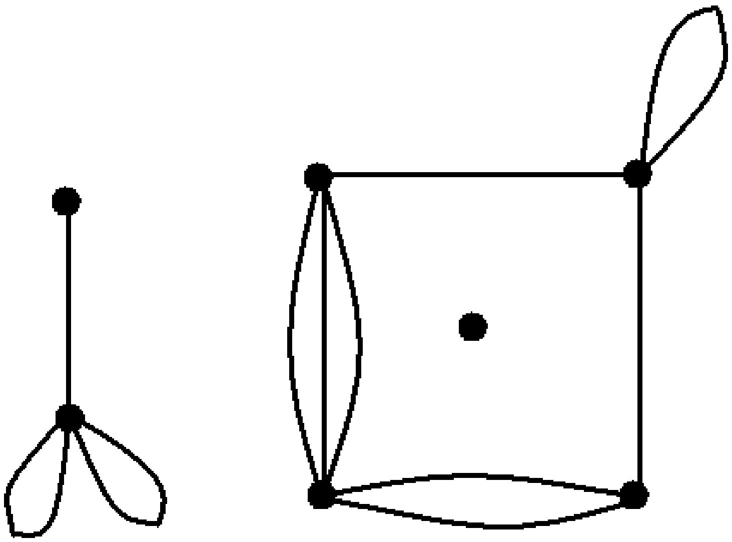


Exercise 2

Write down the degree sequence of each of the following graphs:



(a)



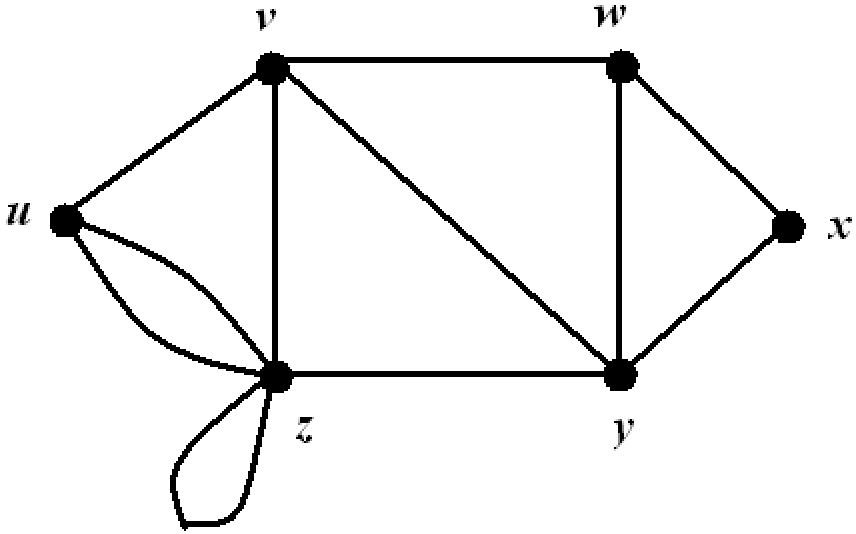
(b)

Exercise 3

Complete the following statements concerning the graph given below:

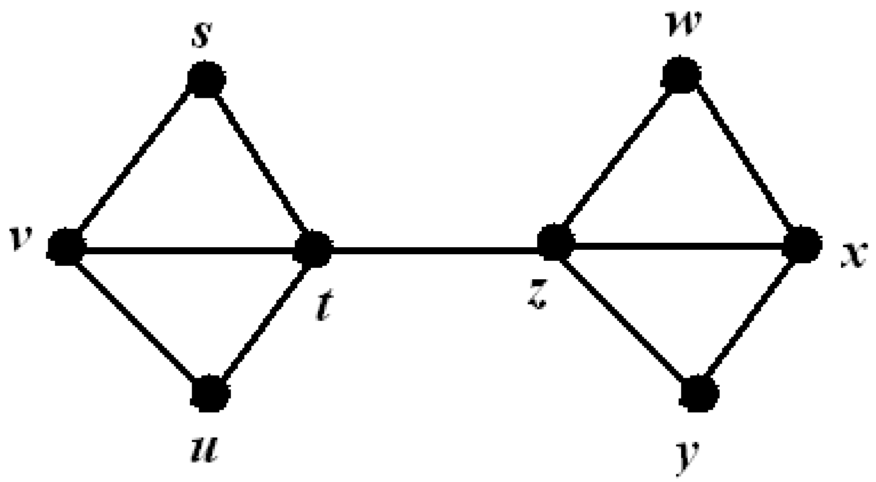
(a) *xyzzvy* is a ________of length____between___and__;

(b) *uvyz* is ___________of length ____between___and__.



Exercise 4

Write down all the paths between s and y in the following graph. Build the distance matrix of the graph.



Exercise 5

Draw the graphs given by the following representations:

Node list

1 2 3 4
2 4
3 4
4 1 2 3
5 6
6 5

Edge list

1 2
1 4
2 2
2 4
2 4
3 2
4 3

Adjacency matrix

$$\begin{array}{c c} & \begin{array}{ccccc} 1 & 2 & 3 & 4 & 5 \end{array} \\ \begin{array}{c} 1 \\ 2 \\ 3 \\ 4 \\ 5 \end{array} & \begin{pmatrix} 0 & 2 & 0 & 1 & 1 \\ 2 & 0 & 0 & 1 & 1 \\ 0 & 0 & 0 & 0 & 0 \\ 1 & 1 & 0 & 0 & 2 \\ 1 & 1 & 0 & 2 & 0 \end{pmatrix} \end{array}$$

HW / Assignment 1

Find the degree for each vertex, if adjacency matrix is given:

Sample input

5

1 1 0 0 0

0 0 1 0 1

1 0 1 0 1

0 1 0 0 1

1 0 0 0 1

Sample output:

5 4 5 2 6

HW / Assignment 2

Distance matrix is given.
Find shortest path between 1 and last vertex:

Sample input

5
0 2 0 0 7
2 0 1 0 3
0 1 0 0 2
0 0 0 0 2
7 3 2 2 0

Sample output:

5
1 2 3 5