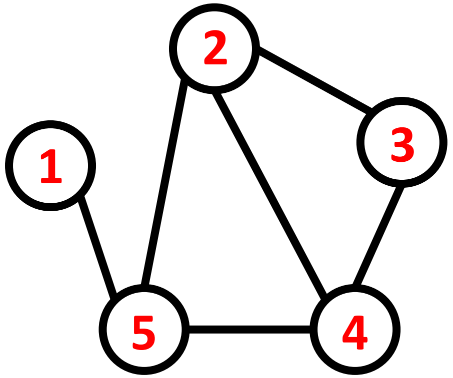
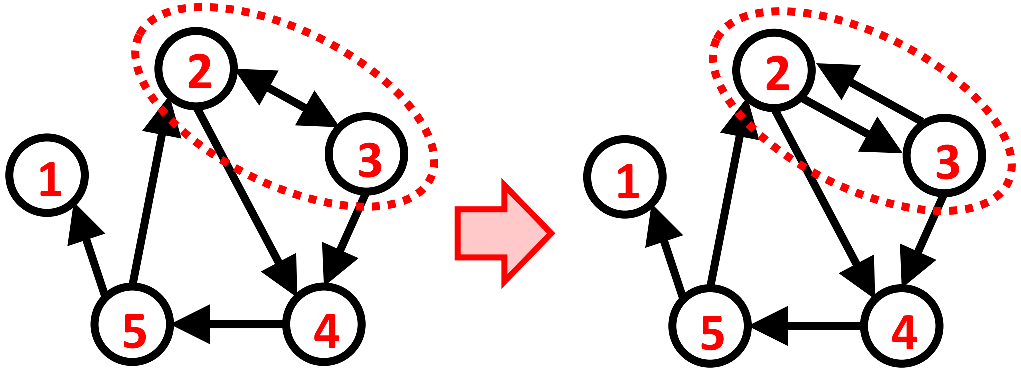
CHAPTER 13:

GRAPH ALGORITHMS

GRAPHS



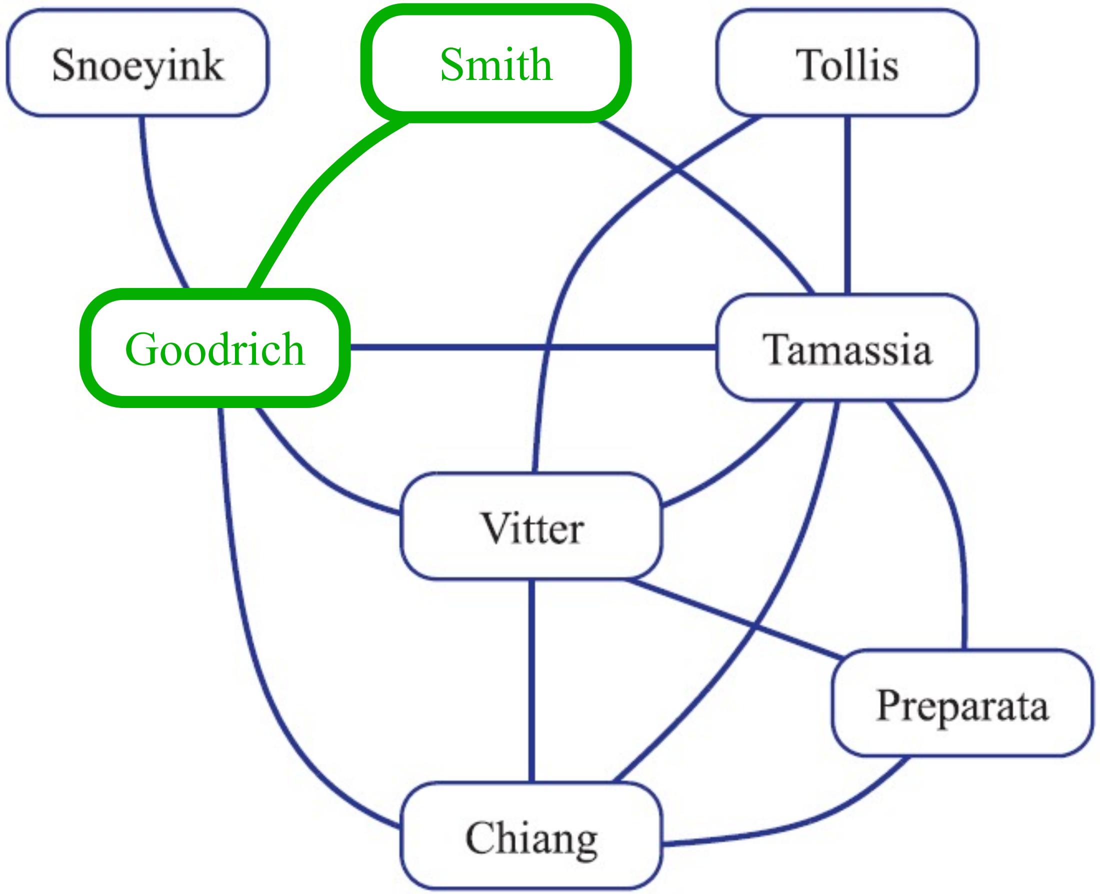
- A graph consists of:
  - A collection of **vertices** (i.e., nodes), V
  - A collection of **edges**, (i.e., links), E
- In the above graph, edges do not have a direction. Thus, it is called “undirected”. In such graphs, an edge between u and v is denoted by either (u,v) or (v,u).
- If the edges have a direction, we call the graph a “directed” graph (or a digraph). In such graphs, an edge from u to v is denoted by (u,v).
- In this example, note that the edge between 2 and 3 has two directions. This is actually built of two edges: (2,3) and (3,2)



GRAPHS – EXAMPLES

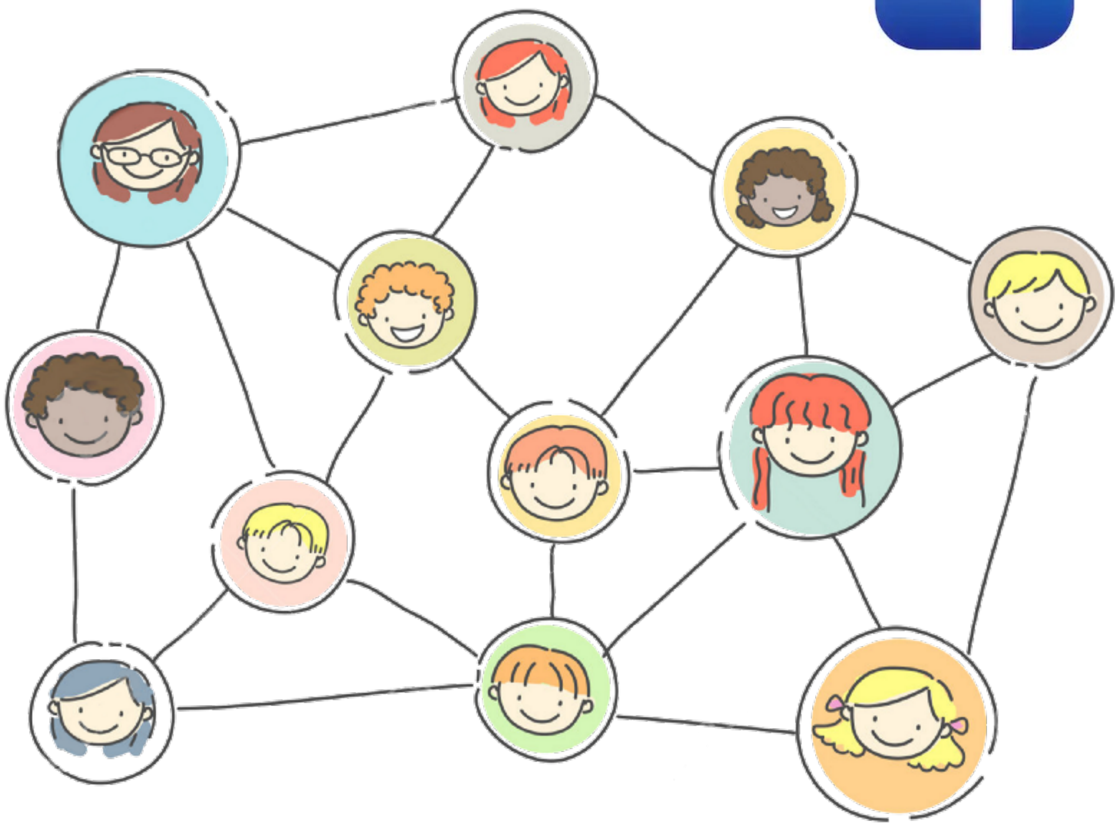
- Co-authorship can be represented as a graph in which:
  - vertices represent authors
  - edges represent co-authorship

- Edges are **undirected**, because if Goodrich co-authored a paper with Smith, then Smith must have also co-authored a paper with Goodrich!



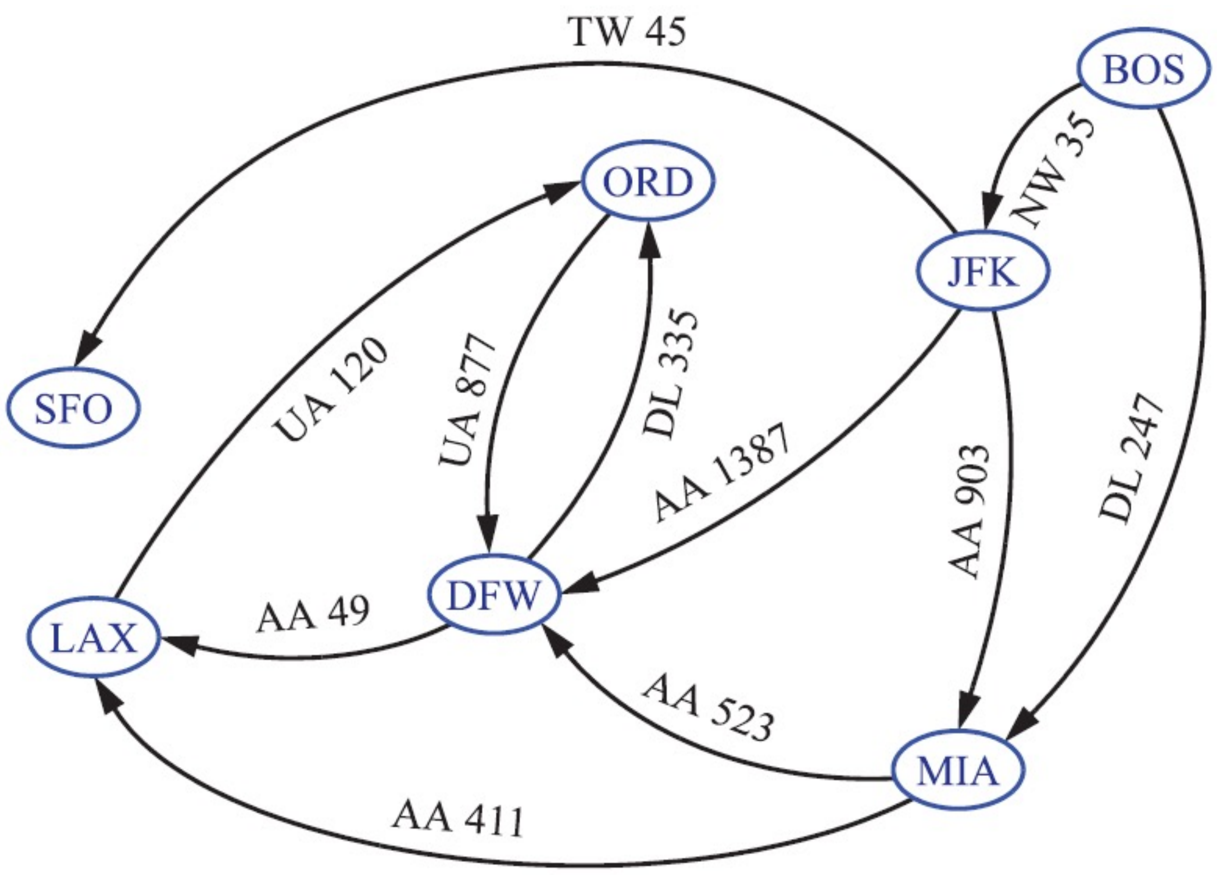
GRAPHS – EXAMPLES

- Facebook can be represented as a graph in which:
  - vertices represent people
  - edges represent friendship
- Edges are **undirected**, because if Mia is a friend of George, then George is also a friend of Mia!
- For Twitter, a directed graph is more suitable; if Adam **follows** Thomas, then Thomas may not follow Adam!



GRAPHS – EXAMPLES

- Air transportation can be represented as a graph in which:
  - vertices represent airports
  - edges represent flights
- Edges are **directed** since each flight has a direction.



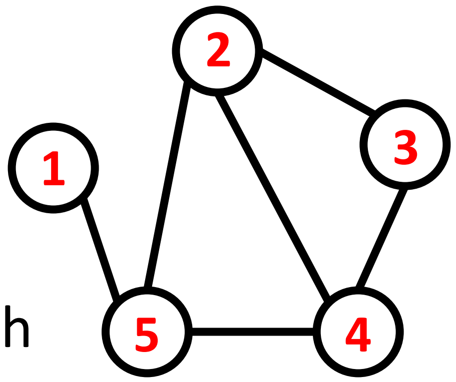
GRAPHS – EXAMPLES

- A city map is a graph where vertices are intersections and directed edges are roads

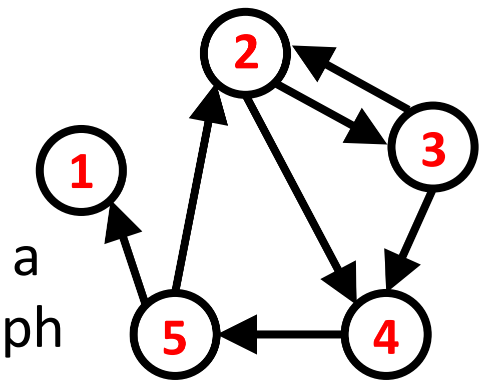
GRAPHS – TERMINOLOGY

- Two vertices joined by an edge are end vertices (or endpoints) for that edge
- If an edge is **directed**, the first endpoint is its origin and the other is its destination
- Two vertices u and v are said to be adjacent, if they are connected by an edge
- An edge is incident on a vertex, if the vertex is one of its endpoints
- The outgoing edges of a vertex v are the directed edges whose **origin** is v
- The incoming edges of a vertex v are the directed edges whose **destination** is v

Example of an **undirected** graph

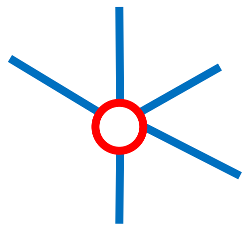


Example of a **directed** graph



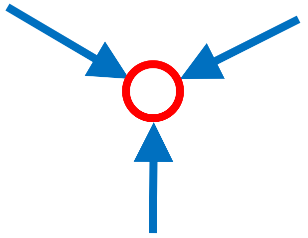
GRAPHS – TERMINOLOGY

- The degree of a vertex v, denoted deg(v), is the number of **incident** edges of v



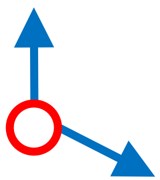
degree = 5

- The in-degree of a vertex v, denoted by indeg(v), is the number of the **incoming** edges of v



in-degree = 3

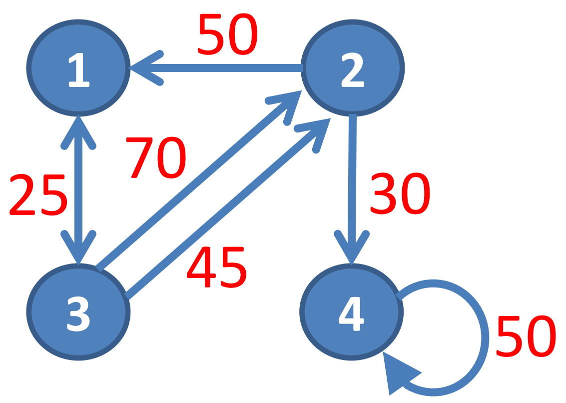
- The out-degree of a vertex v, denoted by outdeg(v), is the number of the **outgoing** edges of v



out-degree = 2

GRAPHS – TERMINOLOGY

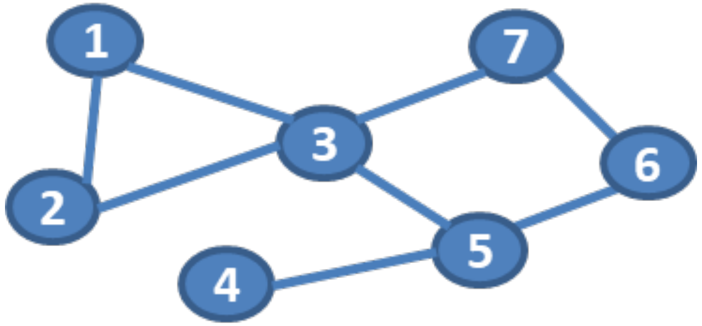
- A graph may have:
  - Parallel edges, e.g., two edges (3,2) and (3,2)
  - Self loops, e.g., an edge (4,4)



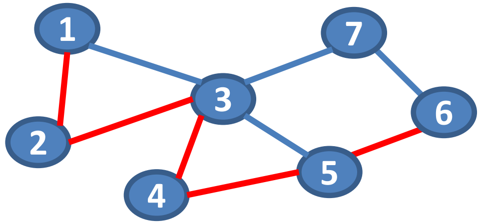
- A graph that does not contain any of these is called a "**simple graph**"
- Moving forward we will only consider **simple graphs**
- Edges could be weighted. The weights can represent various properties, e.g.:
  - Given an air transportation network, the weight can represent flight frequency
  - Given a water-pipes network, the weight could represent pipe capacity
  - Given a road network, the weight could represent road length

GRAPHS – TERMINOLOGY

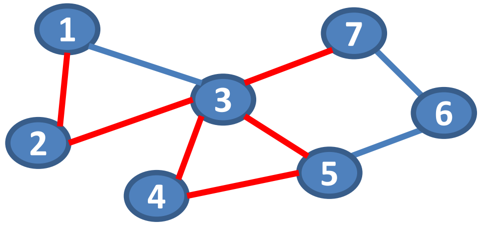
- Given the following graph:



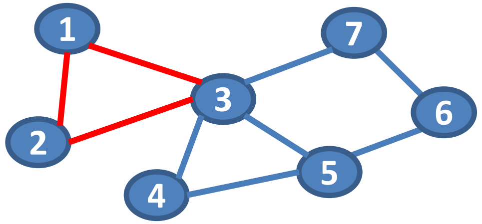
A **simple path** from 1 to 6:
(1, 2, 3, 4, 5, 6)



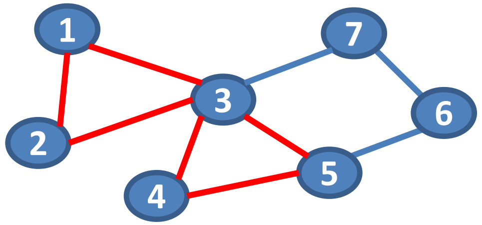
A **path** that is **not simple** from 1 to 7:
(1, 2, 3, 4, 5, 3, 7)



A **simple cycle** from 1 to 1:
(1, 2, 3, 1)

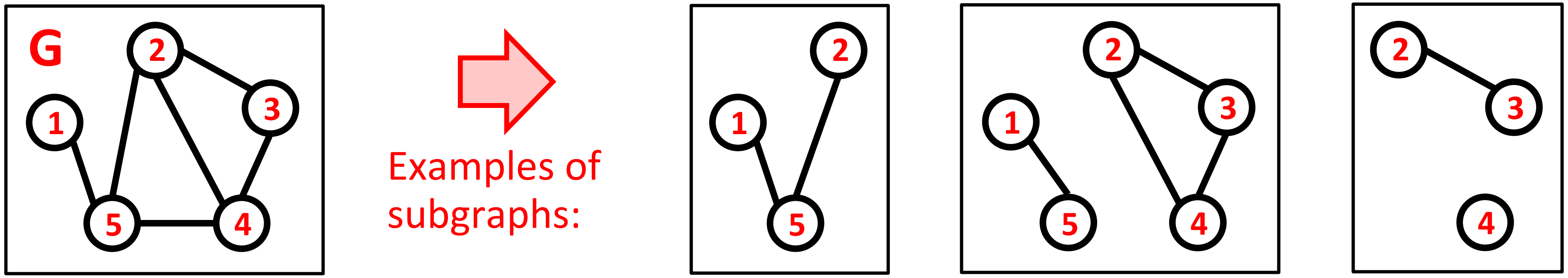


A **cycle** that is **not simple** from 1 to 1:
(1, 2, 3, 4, 5, 3, 1)

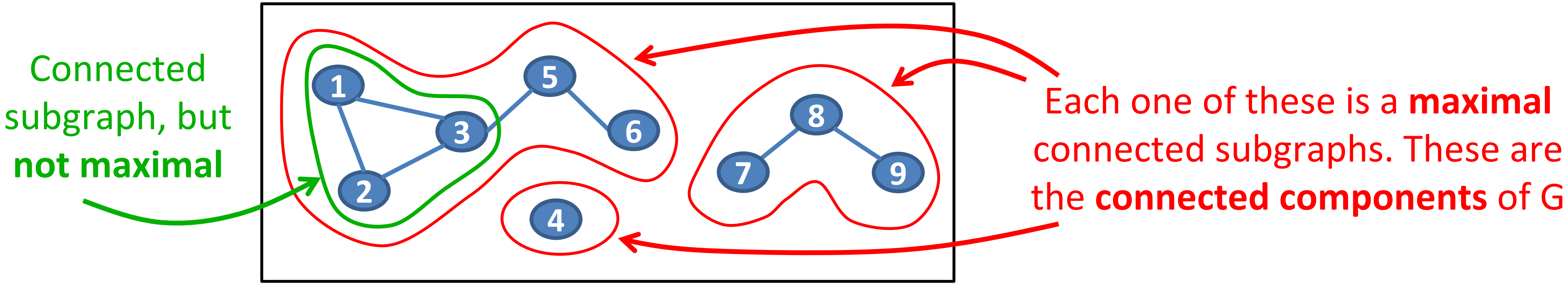


GRAPHS – TERMINOLOGY

- A subgraph of a graph G is a graph whose vertices and edges are subsets of the vertices and edges of G



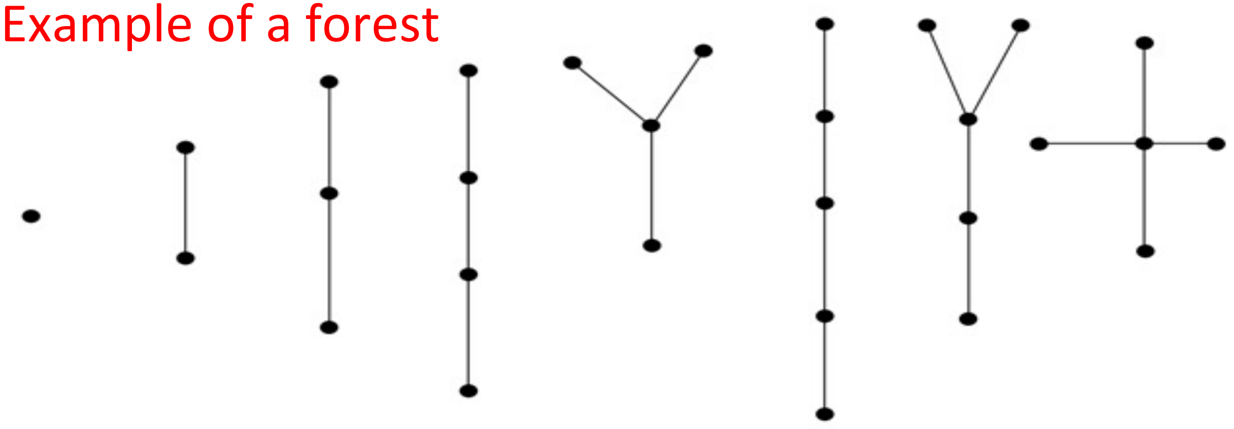
- A graph is connected if there exists a path between any two vertices; otherwise it is disconnected. Here is an example of a disconnected graph, G:



GRAPHS – TERMINOLOGY

- A forest is a graph without cycles
  - It is called a "forest" because it consists of trees.
  - What if it is **connected?**

Example of a forest

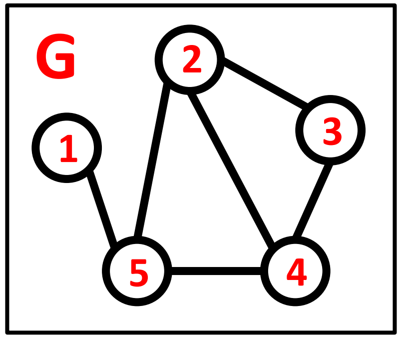


- Note that these trees do not have any vertex that is specified as the root; this makes them different from the trees that we have seen so far.

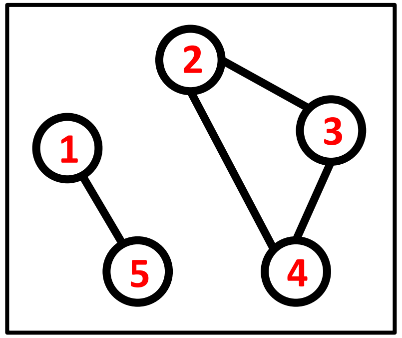
- If a tree has a specified root, it is called a "rooted tree", otherwise it is a "**free** tree"

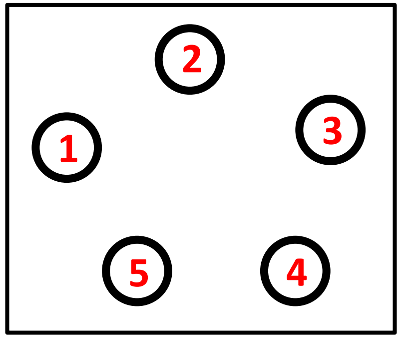
GRAPHS – TERMINOLOGY

- A **spanning** subgraph of G is a subgraph that contains all the vertices of G

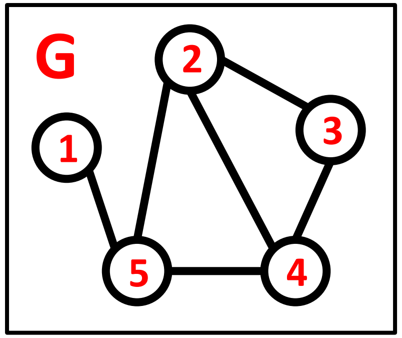


Examples of a **spanning** subgraph:





- A spanning **tree** is a spanning graph that is a free tree



Examples of a **spanning** tree: