Introduction to Graphs

Tecniche di Programmazione – A.A. 2016/2017

Summary

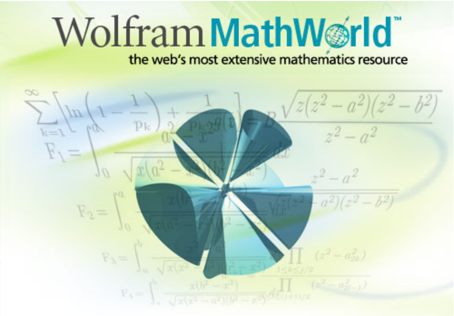
Definition: Graph

Introduction to Graphs

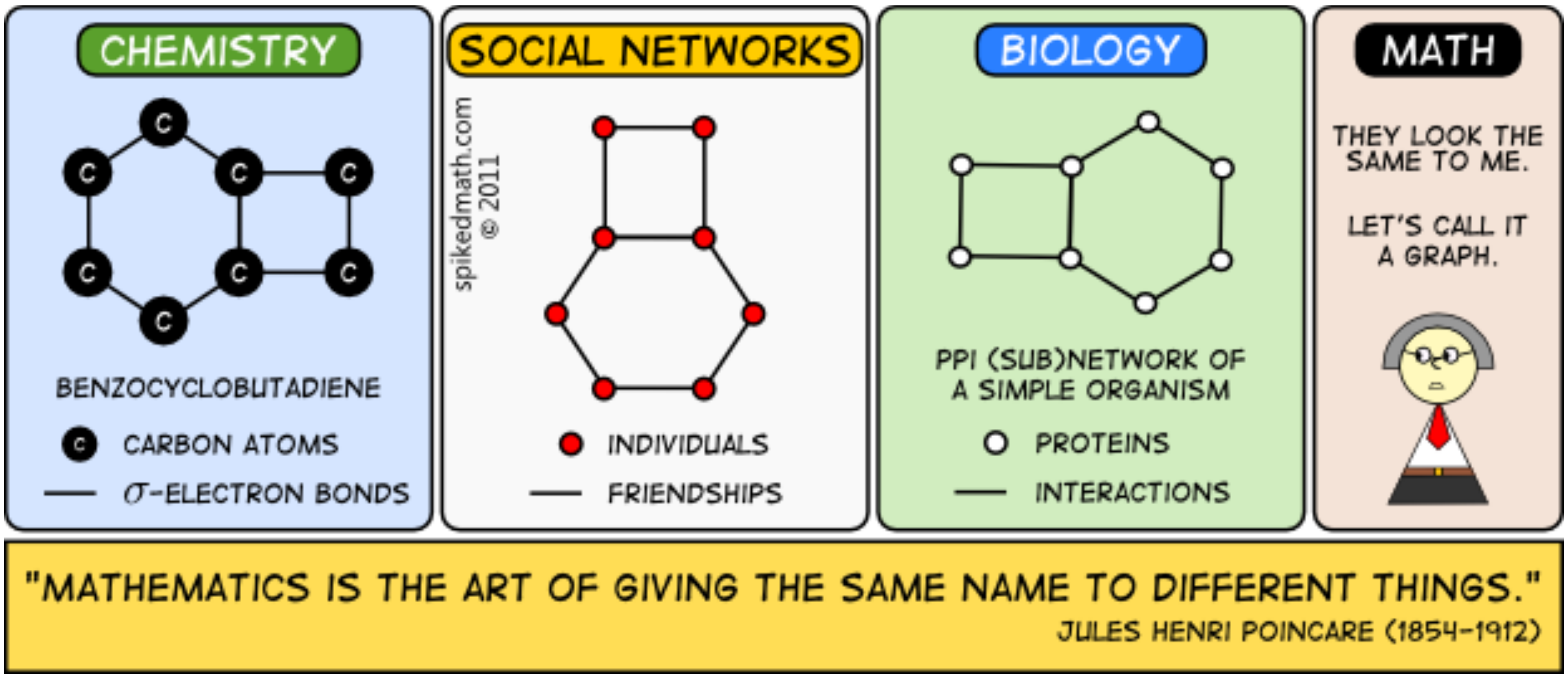
Definition: **Graph**

- A **graph** is a collection of **points** and **lines** connecting some (possibly empty) subset of them.
- The points of a graph are most commonly known as **graph vertices**, but may also be called "nodes" or simply "points."
- The lines connecting the vertices of a graph are most commonly known as **graph edges**, but may also be called "arcs" or "lines."

http://mathworld.wolfram.com/



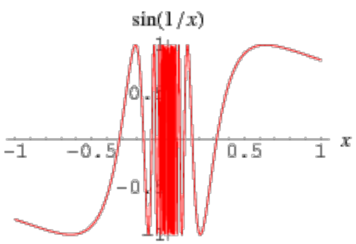
What's in a name?

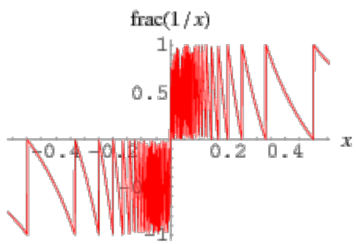


http://spikedmath.com/382.html

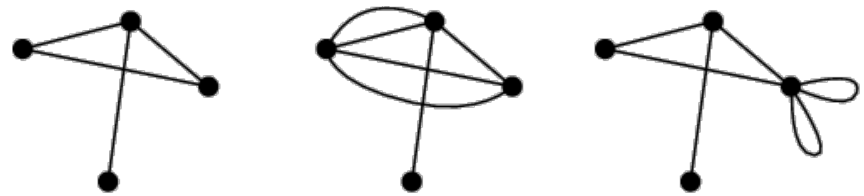
Big warning: Graph ≠ Graph ≠ Graph

Graph (plot)
(italiano: grafico)





Graph (maths)
(italiano: grafo)



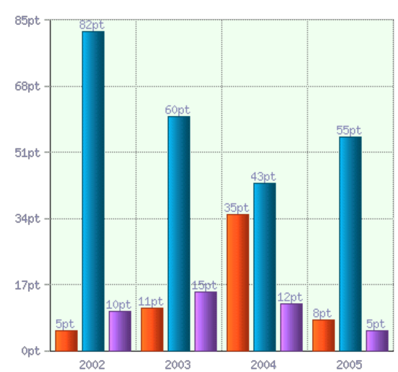
simple graph

nonsimple graph with multiple edges

nonsimple graph with loops

$\neq$

Graph (chart)
(italiano: grafico)



History

- The study of graphs is known as **graph theory**, and was first systematically investigated by D. König in the 1930s
- Euler's proof about the *walk across all seven bridges of Königsberg* (1736), now known as the *Königsberg bridge* problem, is a famous precursor to graph theory.
- In fact, the study of various sorts of paths in graphs has many applications in real-world problems.

Königsberg Bridge Problem

- Can the 7 bridges the of the city of Königsberg over the river Preger all be traversed in a single trip without doubling back, with the additional requirement that the trip ends in the same place it began?

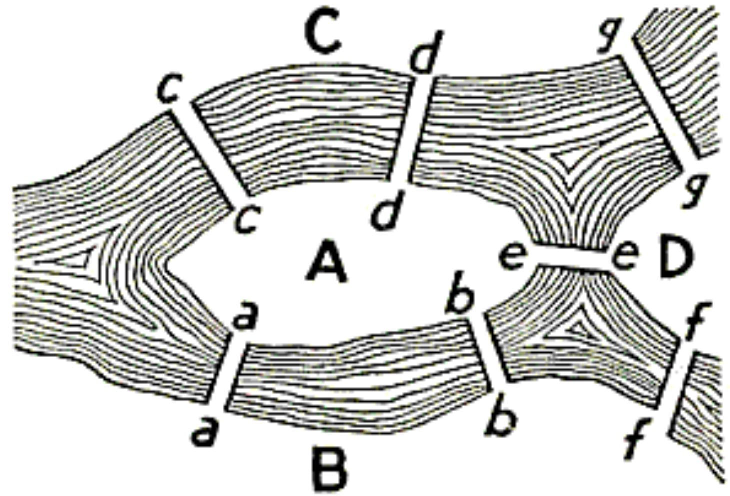
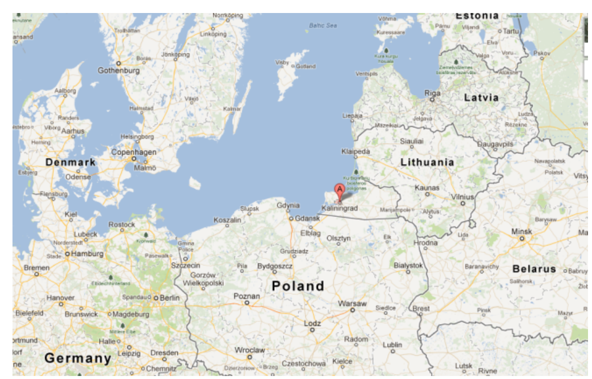


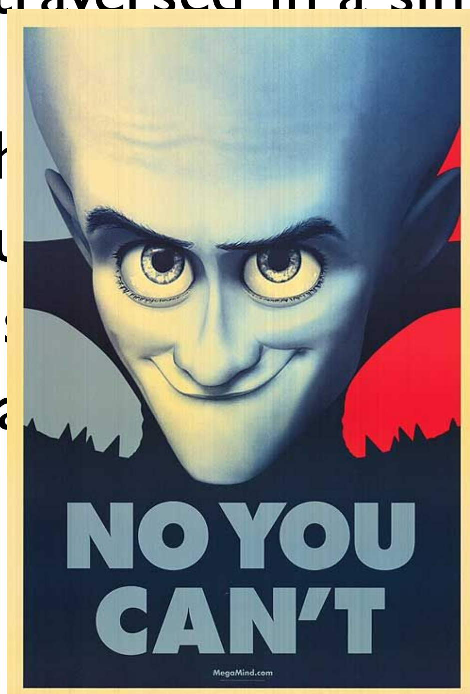
FIGURE 98. *Geographic Map: The Königsberg Bridges.*



Today: Kaliningrad, Russia

Königsberg Bridge Problem

- Can the 7 bridges the of
  the city of Königsberg
  over the river Preger all
  be traversed in a single
  trip … back,
  with …
  requ… trip
  ends … ce it
  beg…



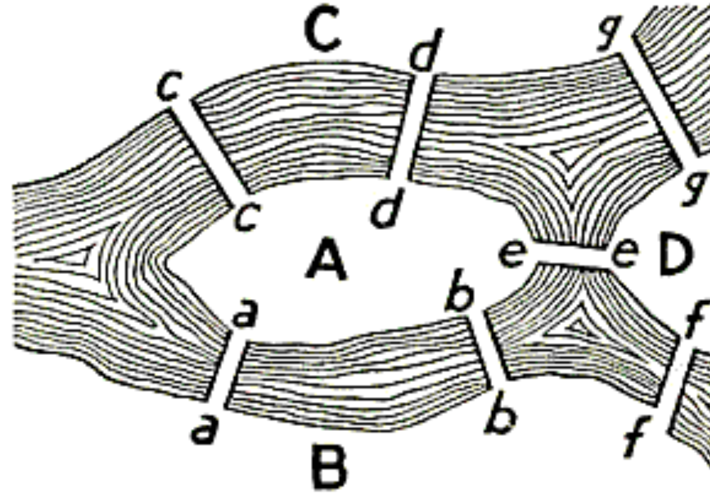
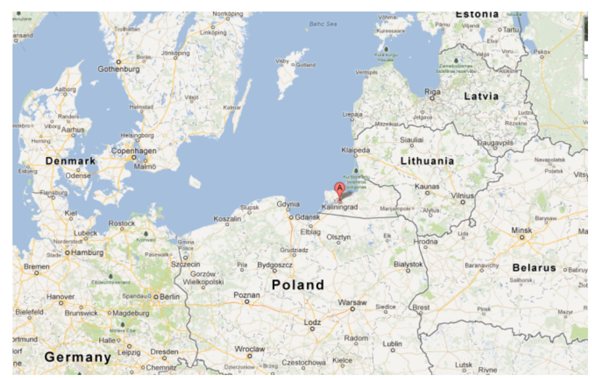
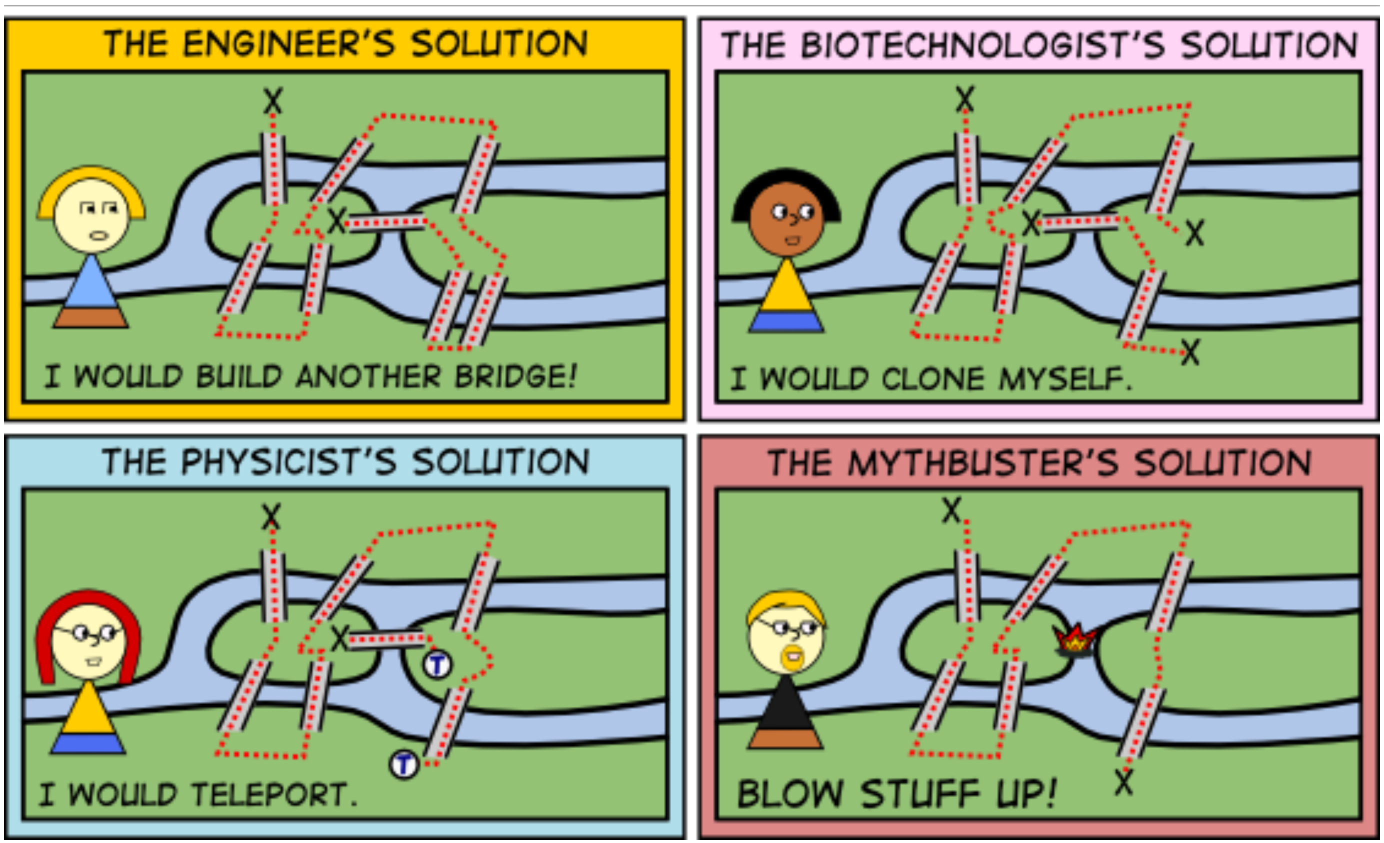


FIGURE 98. *Geographic Map: The Königsberg Bridges.*



Today: Kaliningrad, Russia

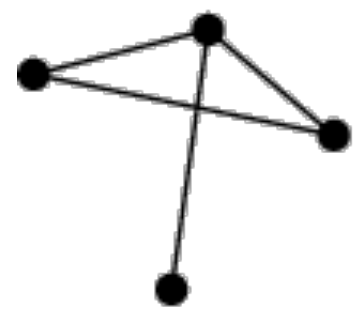
Unless…



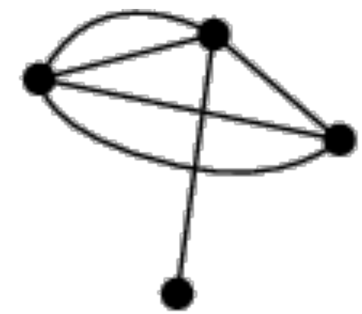
http://spikedmath.com/541.html

Types of graphs: edge cardinality

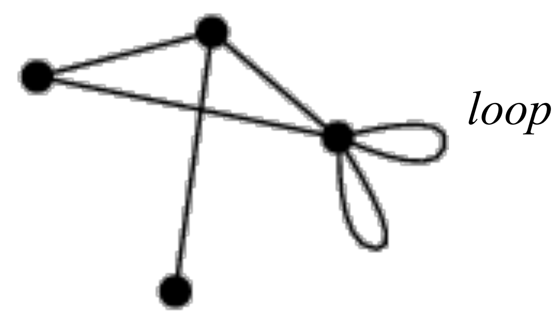
- Simple graph:
  - At most one edge (i.e., either one edge or no edges) may connect any two vertices
- Multigraph:
  - Multiple edges are allowed between vertices
- Loops:
  - Edge between a vertex and itself
- Pseudograph:
  - Multigraph with loops



simple graph



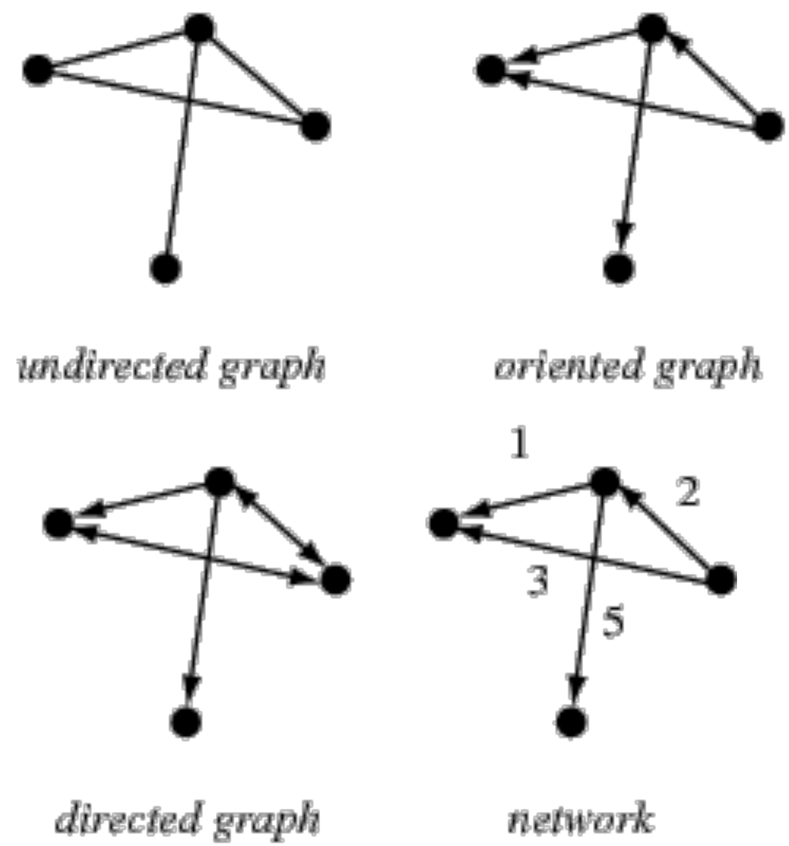
multigraph



pseudograph

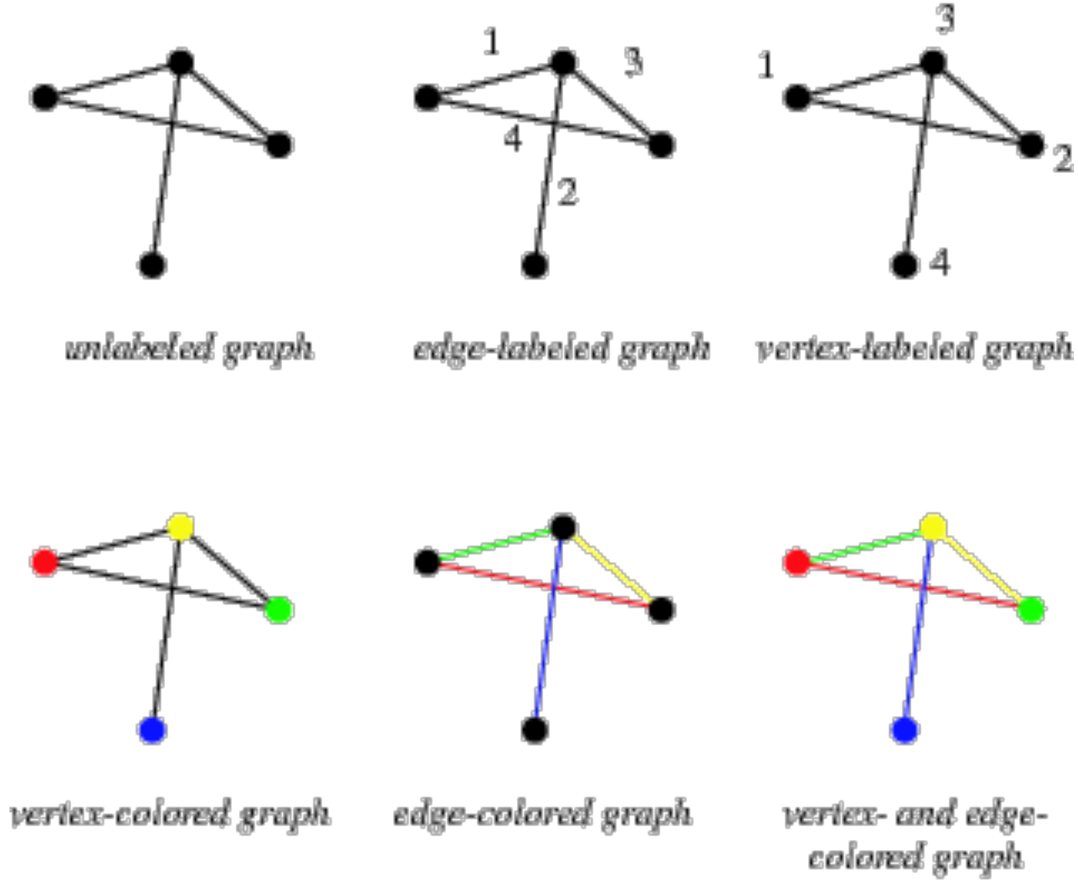
Types of graphs: edge direction

- Undirected
- Oriented
  - Edges have **one** direction (indicated by arrow)
- Directed
  - Edges may have **one or two** directions
- Network
  - Oriented graph with weighted edges



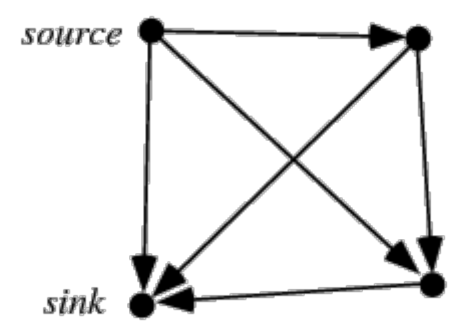
Types of graphs: labeling

- Labels
  - None
  - On Vertices
  - On Edges
- Groups (=colors)
  - Of Vertices
    - no edge connects two identically colored vertices
  - Of Edges
    - adjacent edges must receive different colors
  - Of both

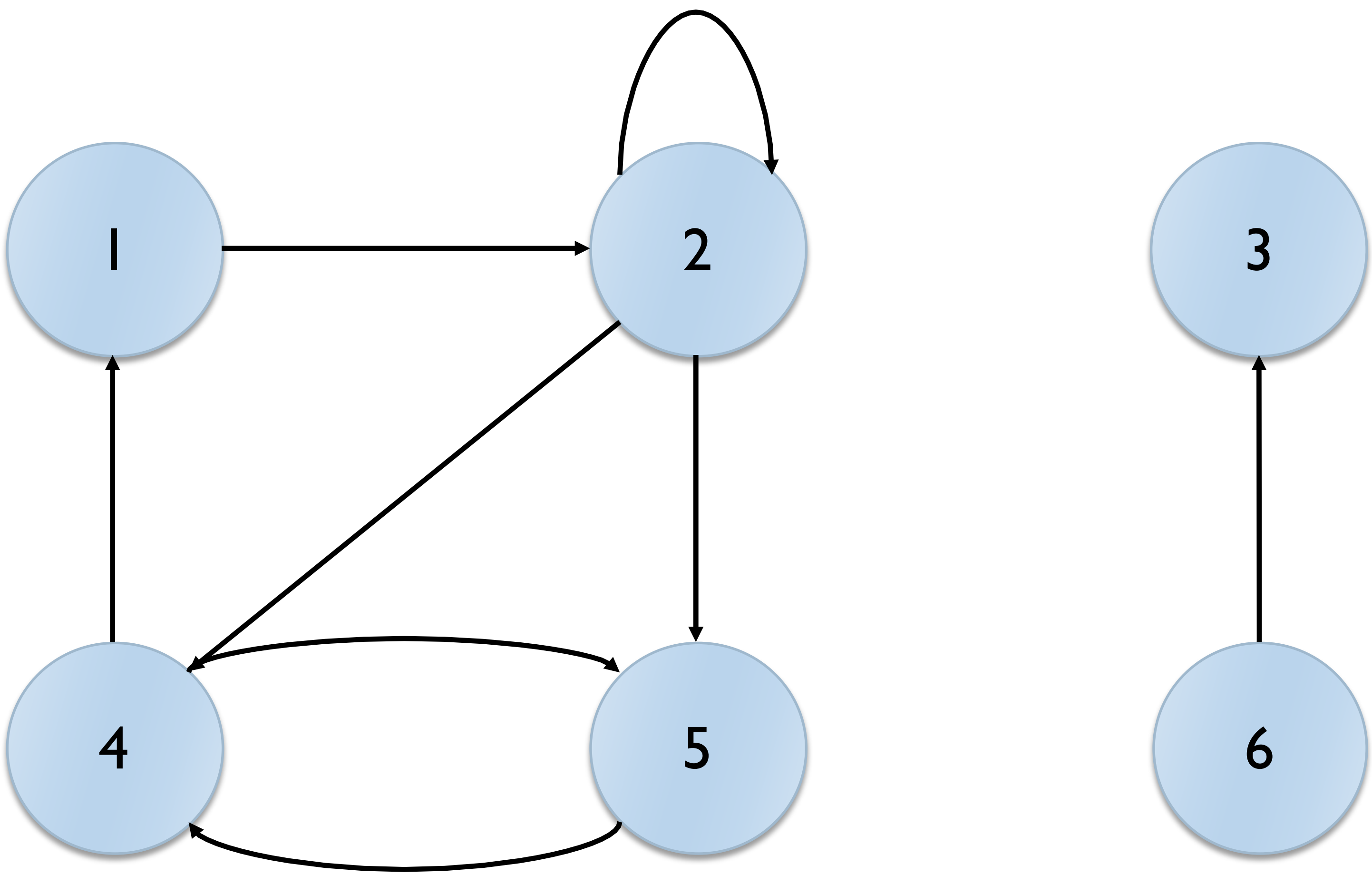


Directed and Oriented graphs

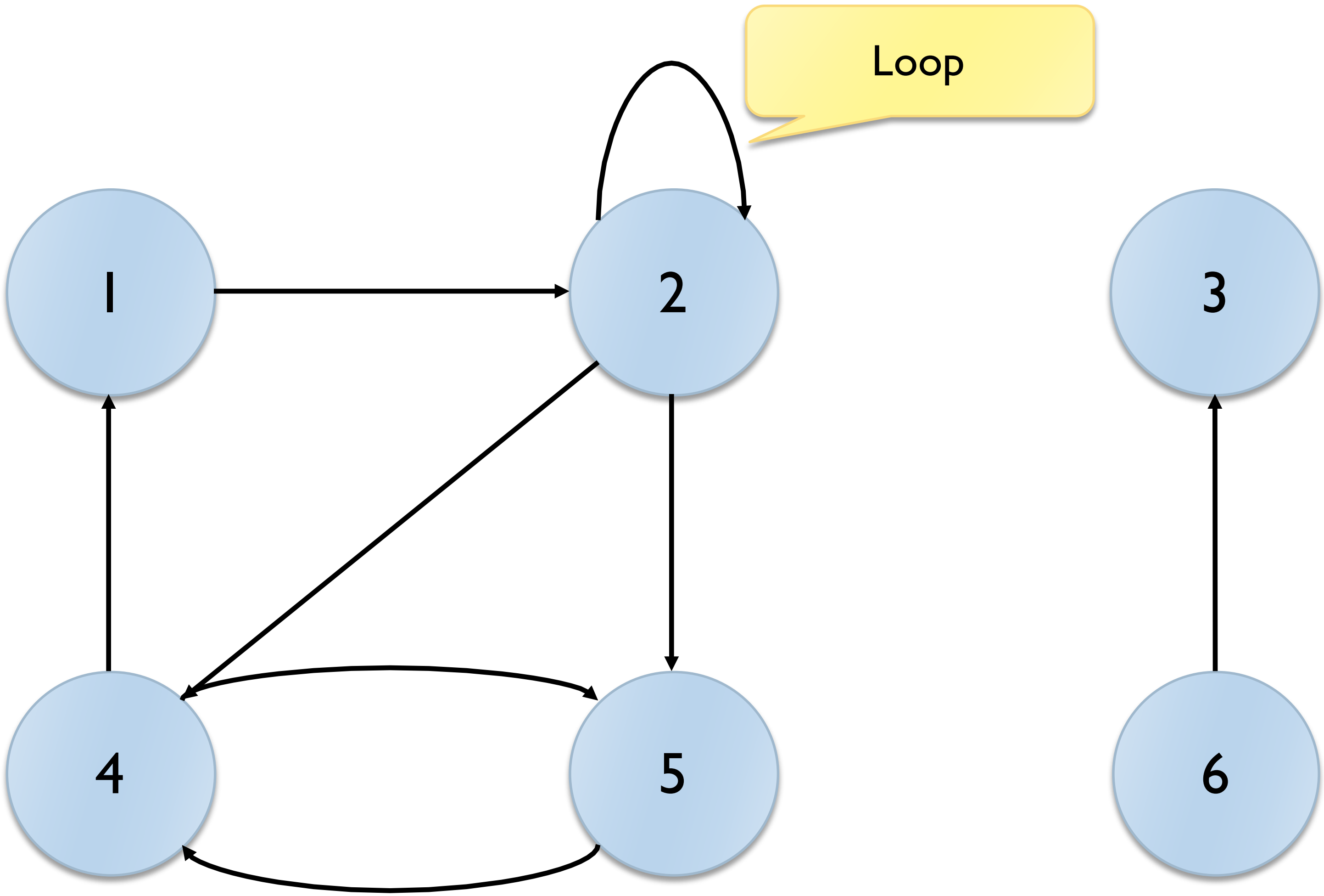
- A Directed Graph (*di-graph*) G is a pair (V,E), where
  - V is a (finite) set of *vertices*
  - E is a (finite) set of *edges*, that identify a binary relationship over V
    - $E \subseteq V \times V$



Example



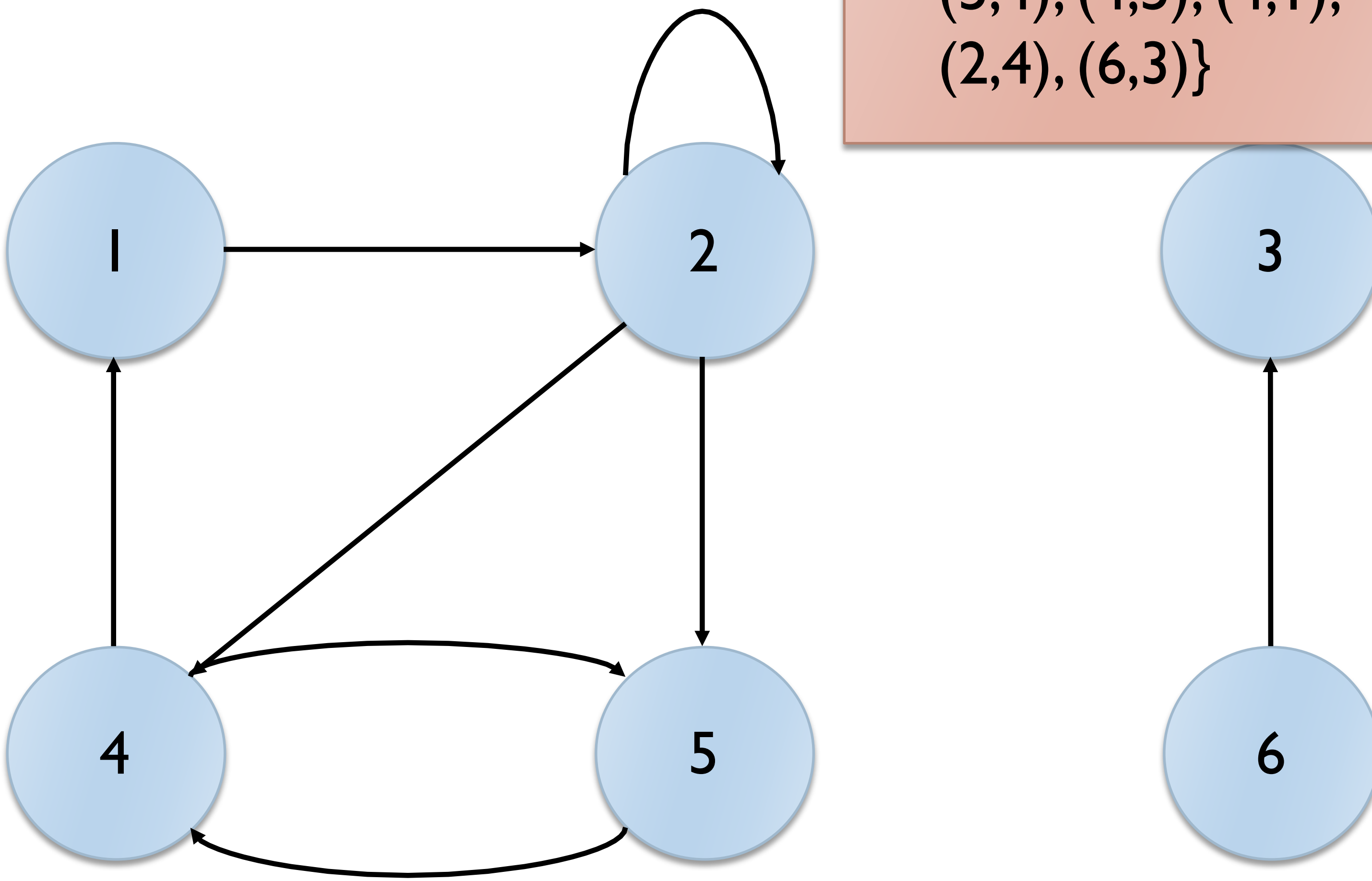
Example



Example

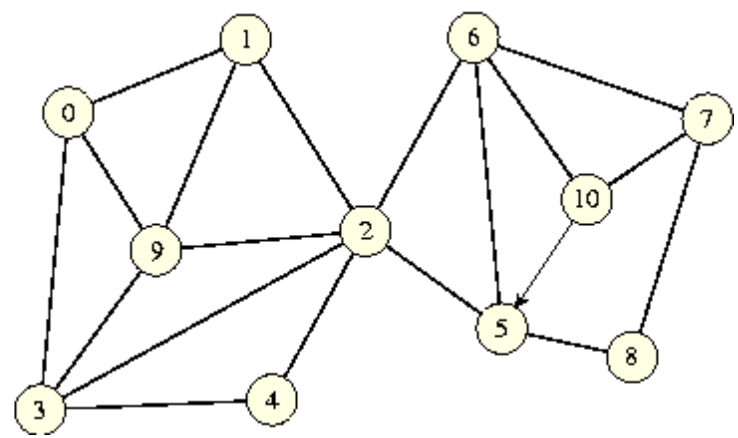
V={1,2,3,4,5,6}

E={(1,2), (2,2), (2,5), (5,4), (4,5), (4,1), (2,4), (6,3)}



Undirected graph

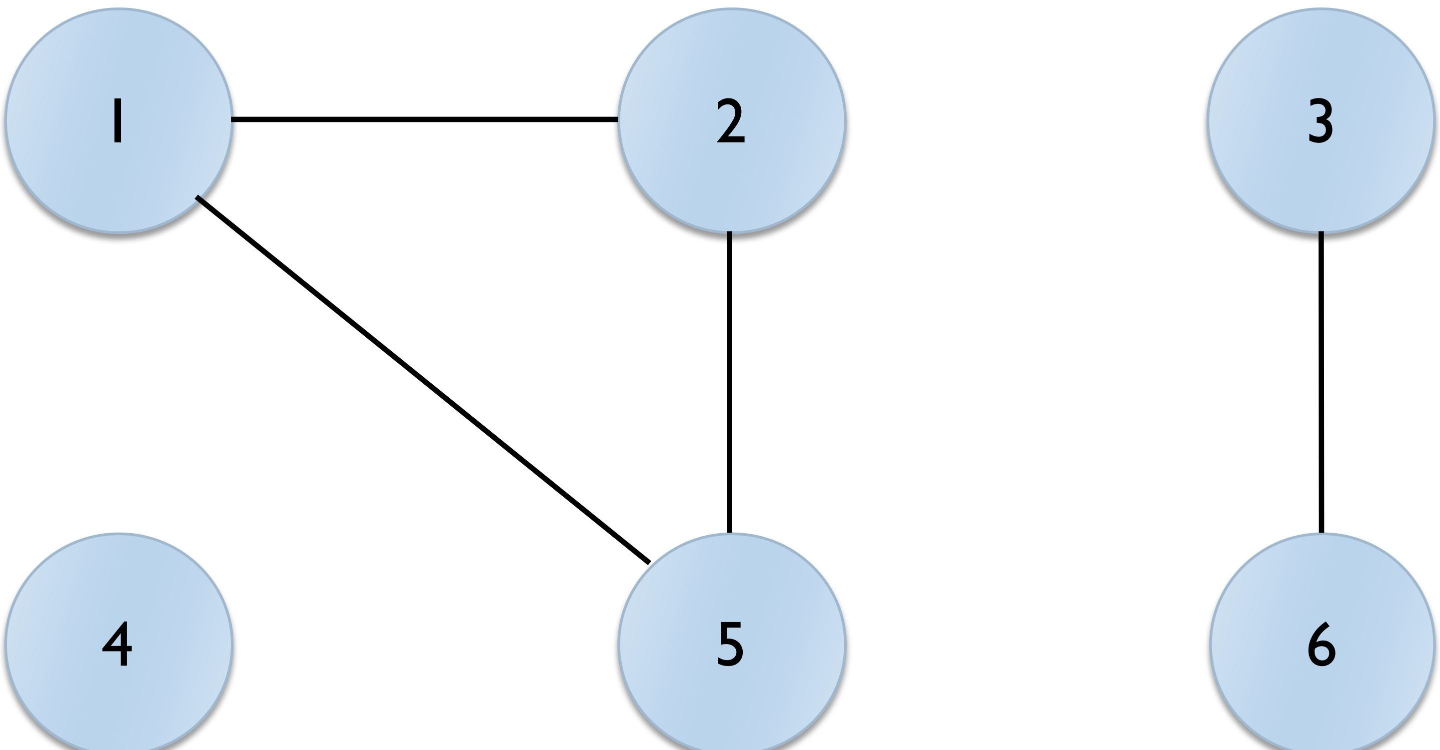
- Ad **Undirected** Graph is still represented as a couple G=(V,E), but the set E is made of **non-ordered pairs** of vertices



Example

V={1,2,3,4,5,6}

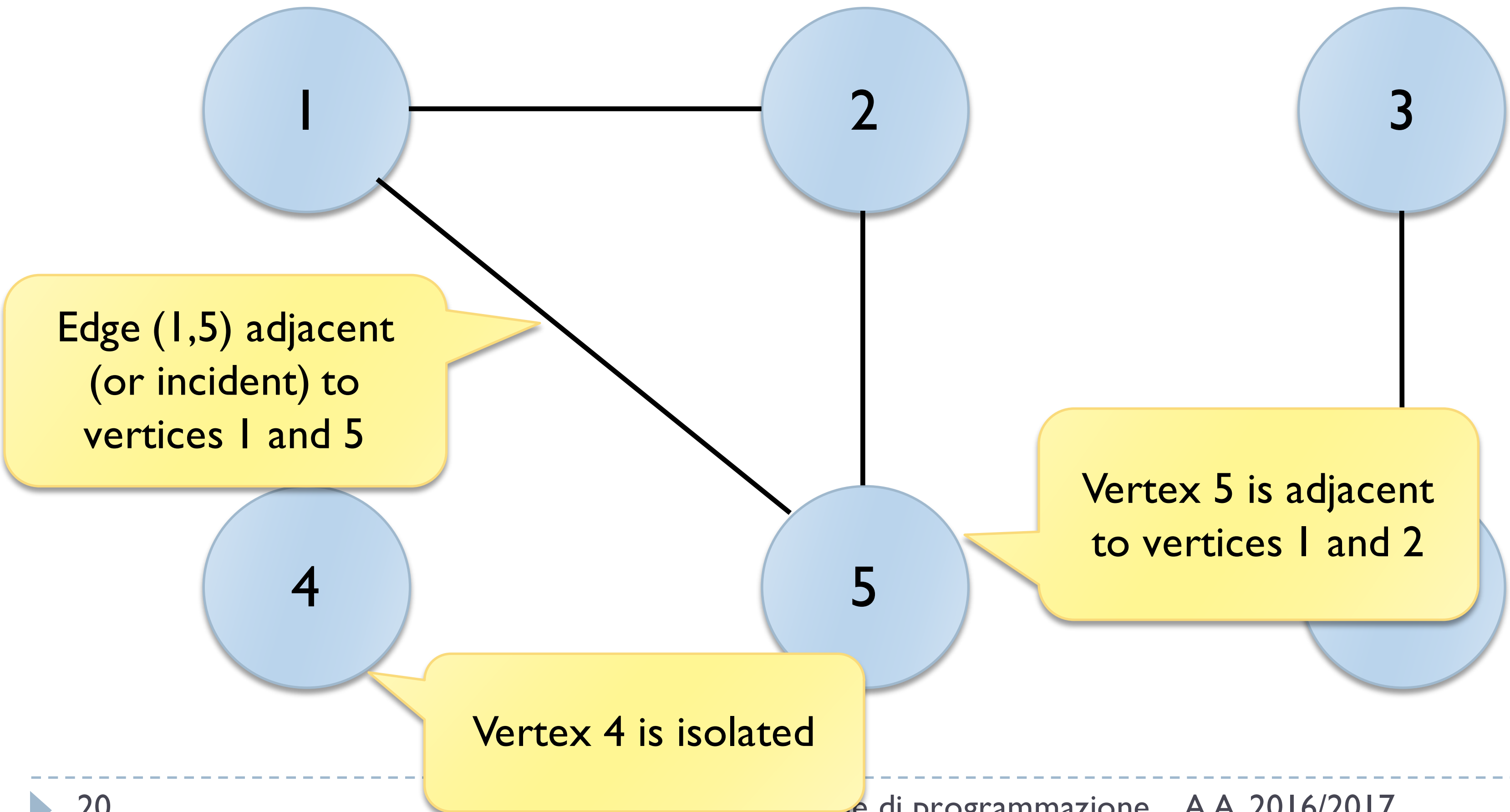
E={(1,2), (2,5), (5,1), (6,3)}



Example

$V=\{1,2,3,4,5,6\}$

$E=\{(1,2), (2,5), (5,1), (6,3)\}$

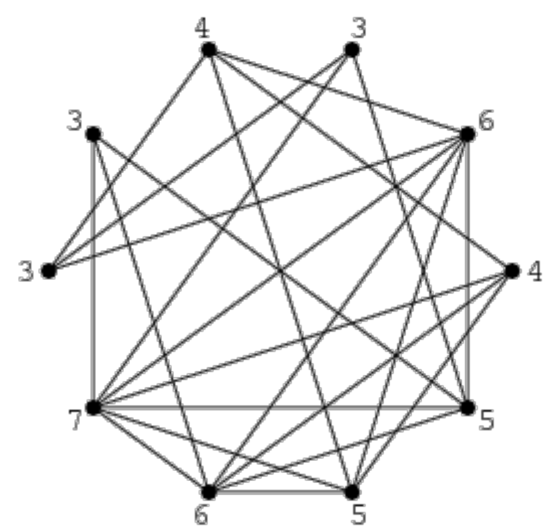


Related Definitions

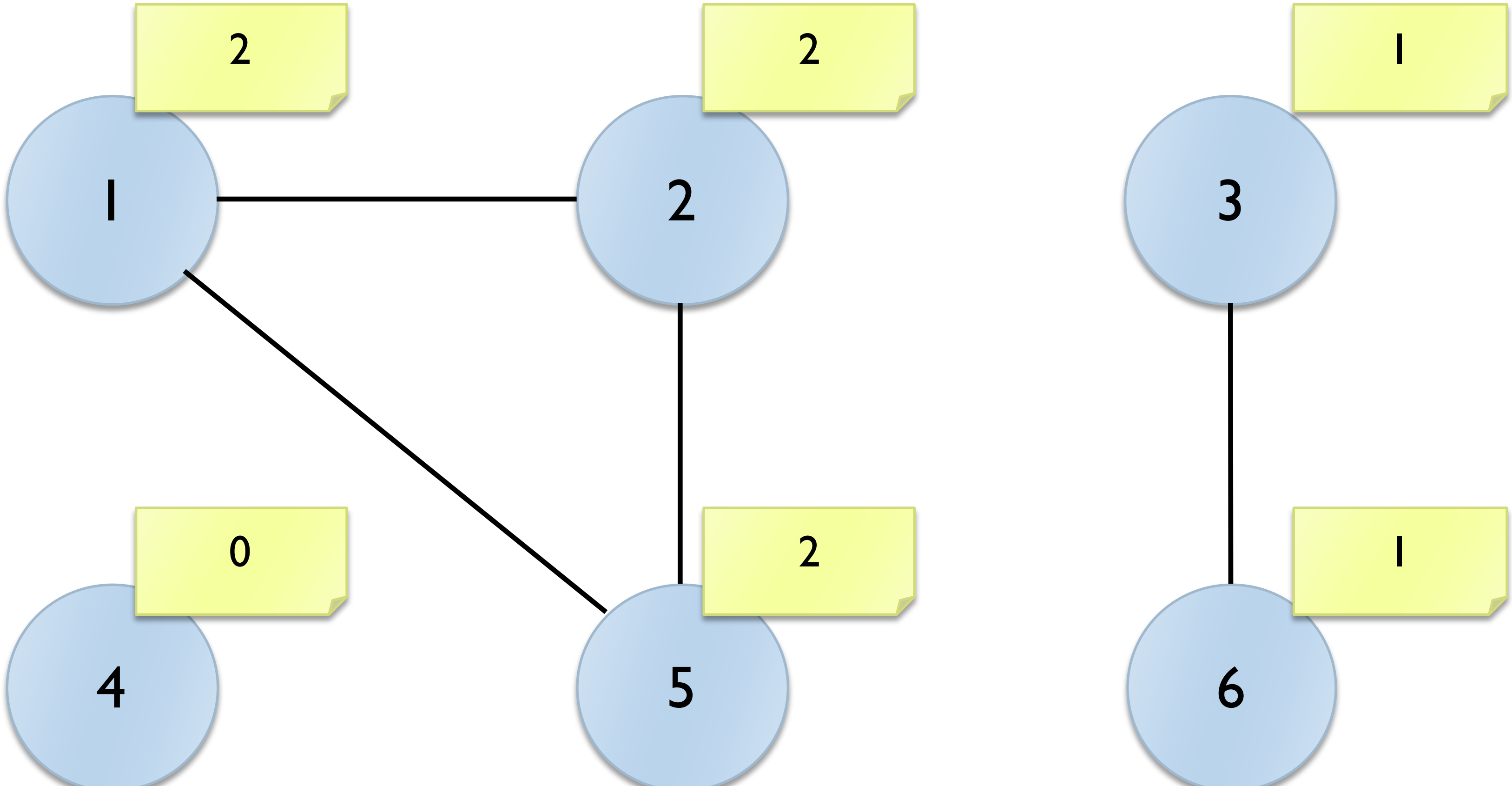
Introduction to Graphs

Degree

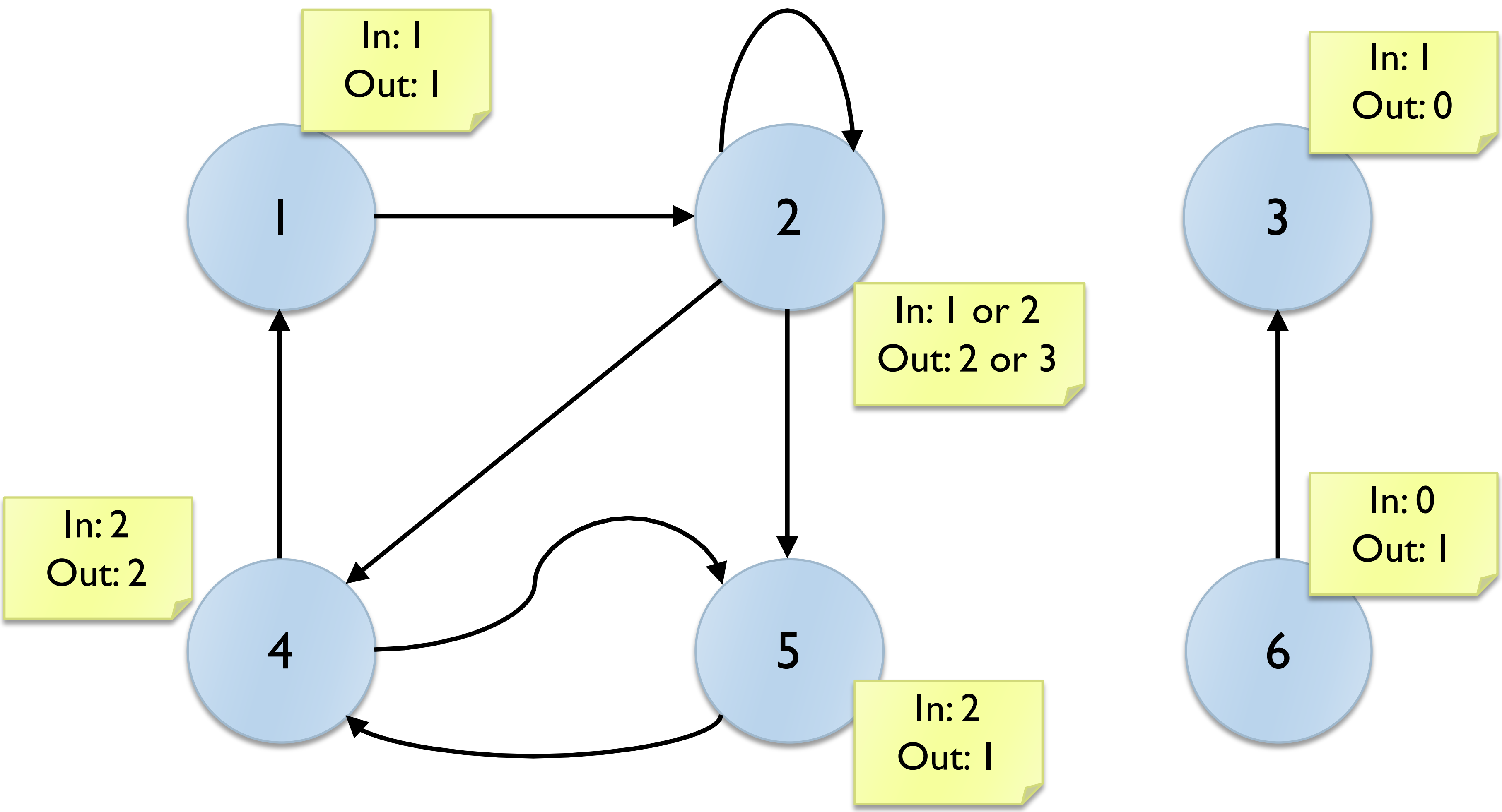
- In an *undirected* graph,
  - the **degree** of a vertex is the number of incident edges
- In a *directed* graph
  - The **in-degree** is the number of incoming edges
  - The **out-degree** is the number of departing edges
  - The **degree** is the sum of in-degree and out-degree
- A vertex with degree 0 is **isolated**



Degree



Degree

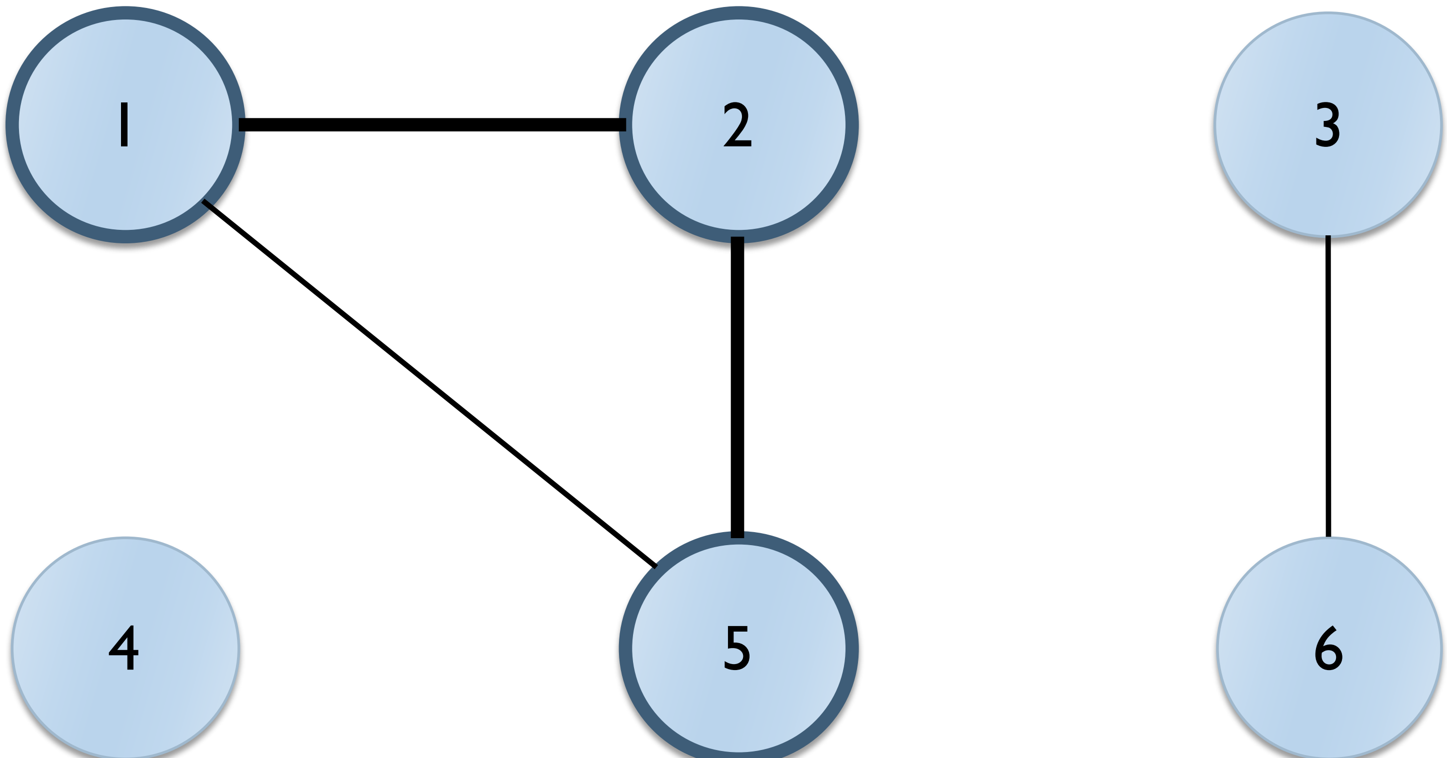

In: 1
Out: 1
1
2
In: 1 or 2
Out: 2 or 3
3
In: 1
Out: 0
4
In: 2
Out: 2
5
In: 2
Out: 1
6
In: 0
Out: 1

Paths

- A **path** on a graph G=(V,E) also called a trail, is a sequence $\{v_1, v_2, \ldots, v_n\}$ such that:
  - $v_1, \ldots, v_n$ are vertices: $v_i \in V$
  - $(v_1, v_2), (v_2, v_3), \ldots, (v_{n-1},v_n)$ are graph edges: $(v_{i-1},v_i) \in E$
  - v_i are distinct (for "simple" paths).
- The **length** of a path is the number of edges (n-1)
- If there exist a path between v_A and v_B we say that v_B is **reachable** from v_A

Example

Path = { 1, 2, 5 }
Length = 2

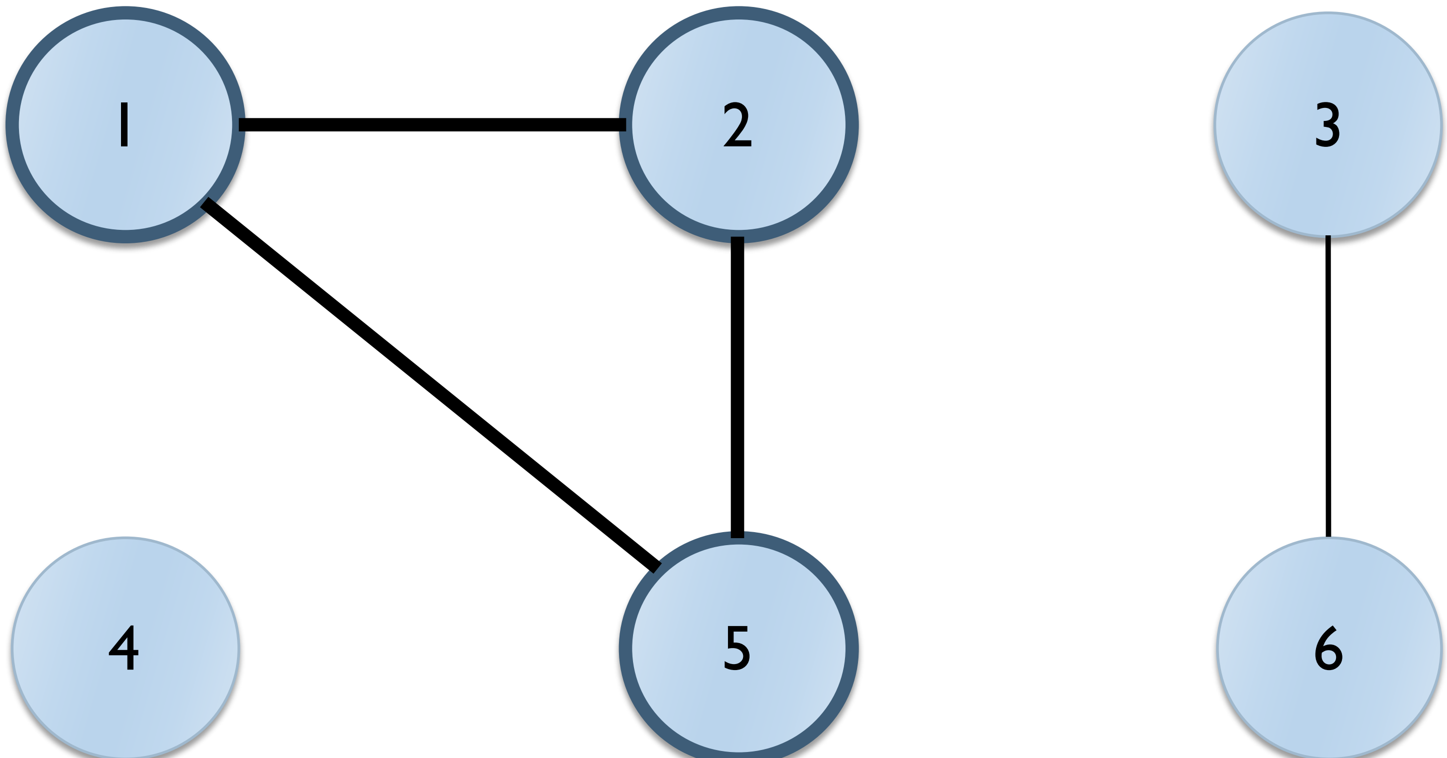


Cycles

- A cycle is a path where $v_1 = v_n$
- A graph with no cycles is said acyclic

Example

Path = { 1, 2, 5, 1 }
Length = 3

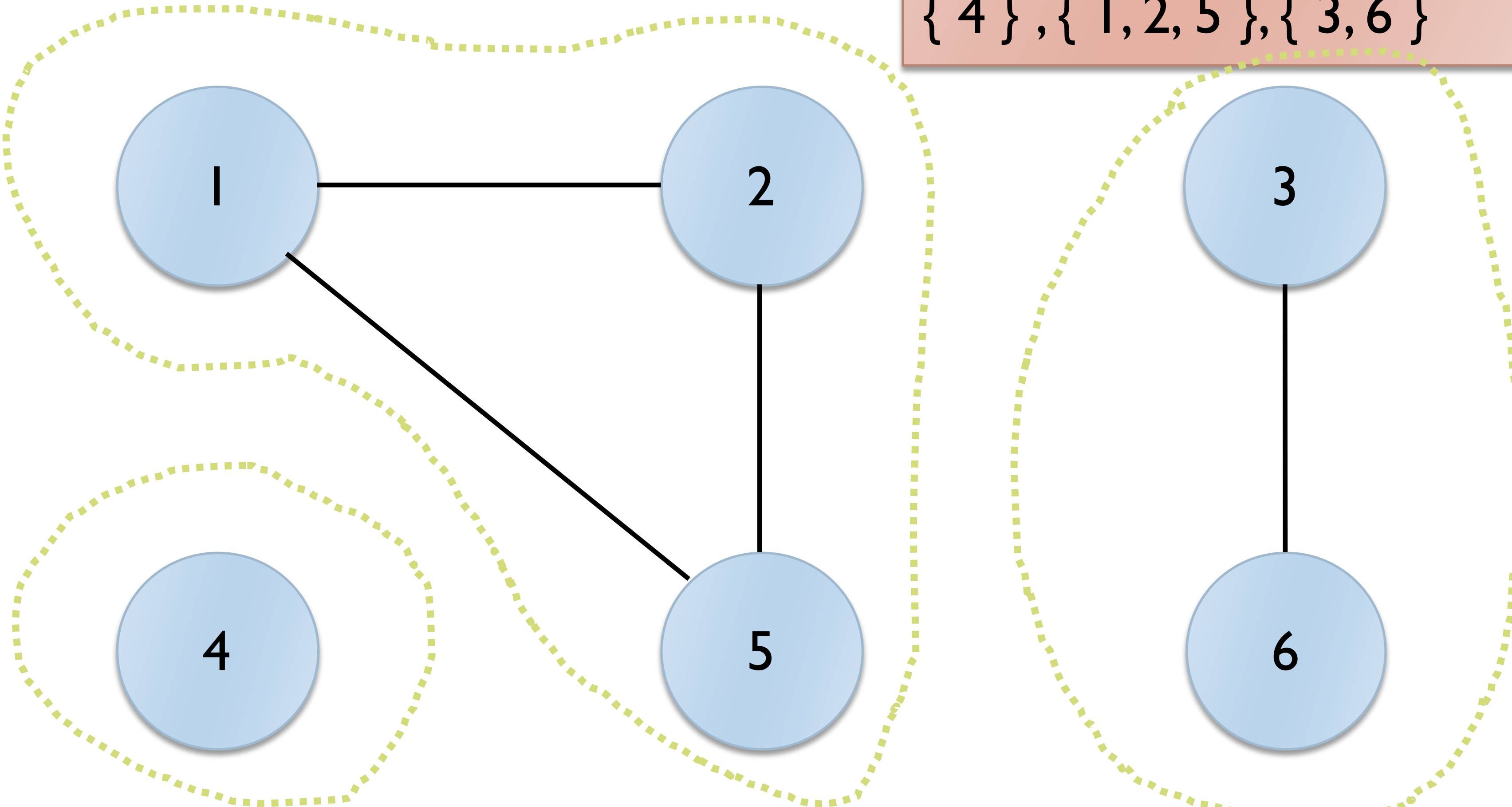


Reachability (Undirected)

- An undirected graph is **connected** if, for every couple of vertices, there is a path connecting them
- The connected sub-graph of maximum size are called **connected components**
- A connected graph has exactly one connected component

Connected components

The graph is **not** connected.
Connected components = 3
{ 4 } ,{ 1, 2, 5 }, { 3, 6 }

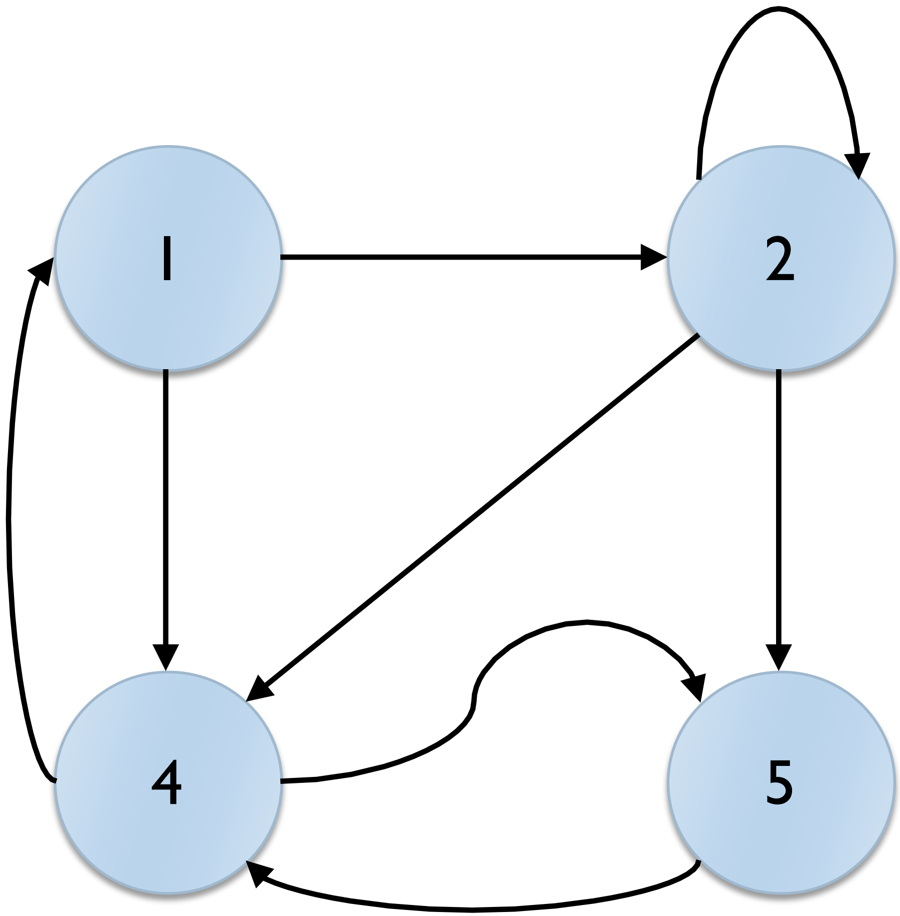


Reachability (Directed)

- A directed graph is **strongly connected** if, for <u>every</u> ordered pair of vertices (v, v’), there exists at least one path connecting v to v’

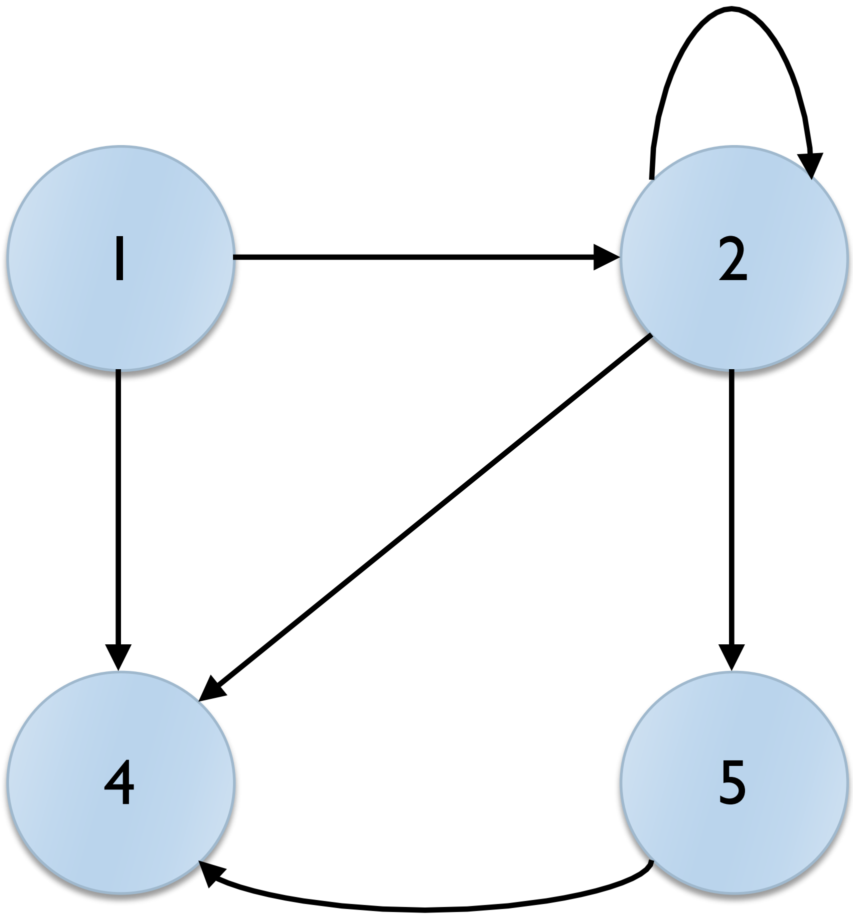
Example

The graph is **strongly connected**



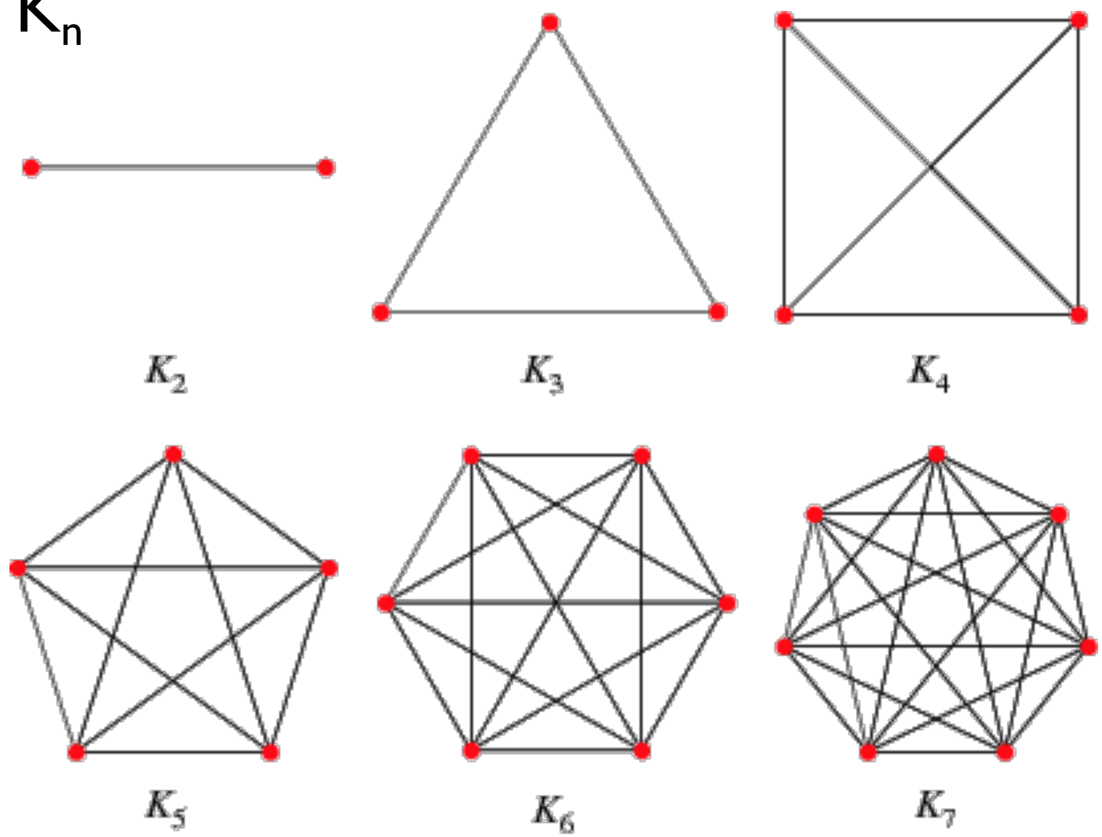
Example

The graph is **not** strongly connected



Complete graph

- A graph is complete if, for every pair of vertices, there is an edge connecting them (they are adjacent)
- Symbol: K_n



Complete graph: edges

- In a **complete** graph with n vertices, the number of **edges** is
  - $n(n-1)$, if the graph is directed
  - $n(n-1)/2$, if the graph is undirected
  - If self-loops are allowed, then
    - n^2 for directed graphs
    - $n(n-1)$ for undirected graphs

Density

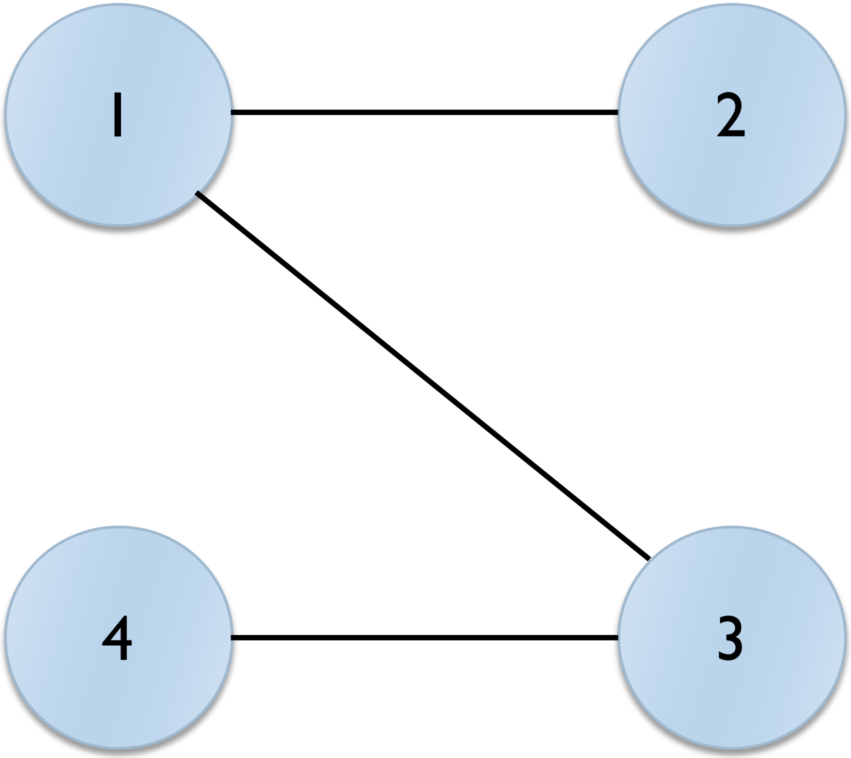
- The density of a graph G=(V,E) is the ratio of the number of edges to the total number of edges

$$d = \frac{|E(G)|}{\left|E\left(K_{|V(G)|}\right)\right|}$$

Esempio

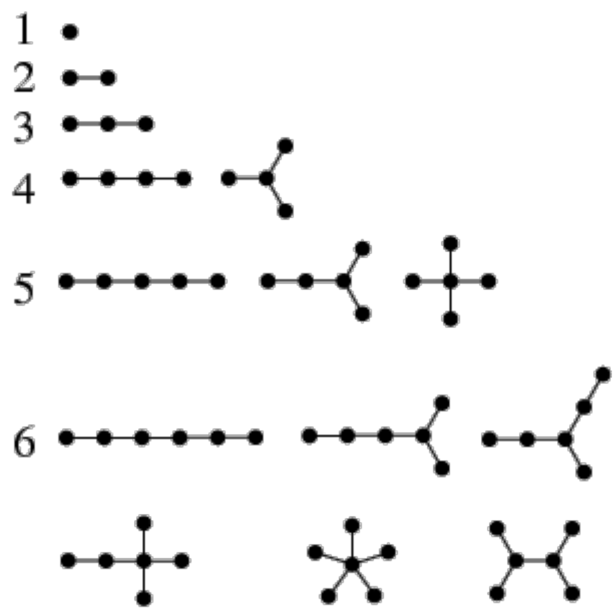
Density = 0.5

Existing: 3 edges
Total: 6 possible edges



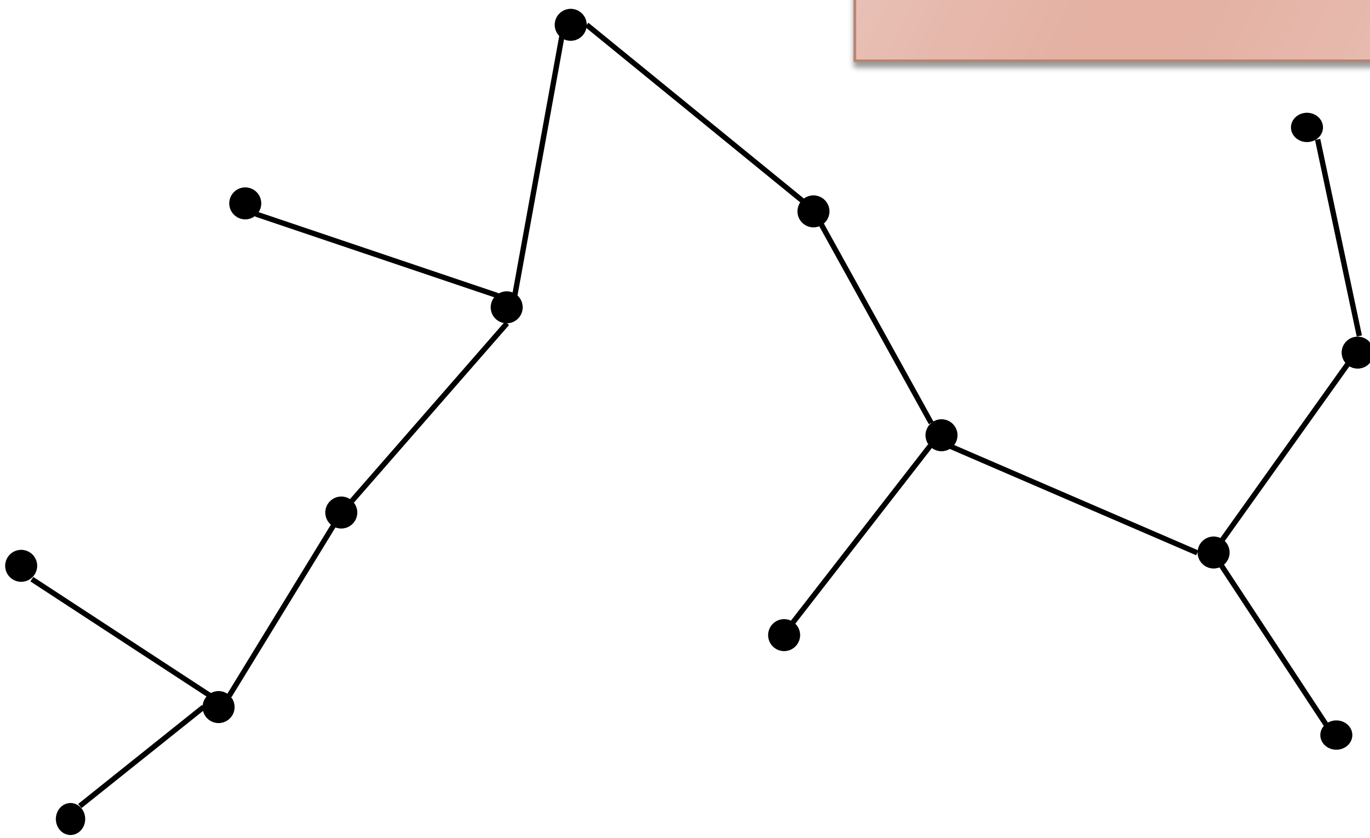
Trees and Forests

- An undirected acyclic graph is called **forest**
- An undirected acyclic connected graph is called **tree**



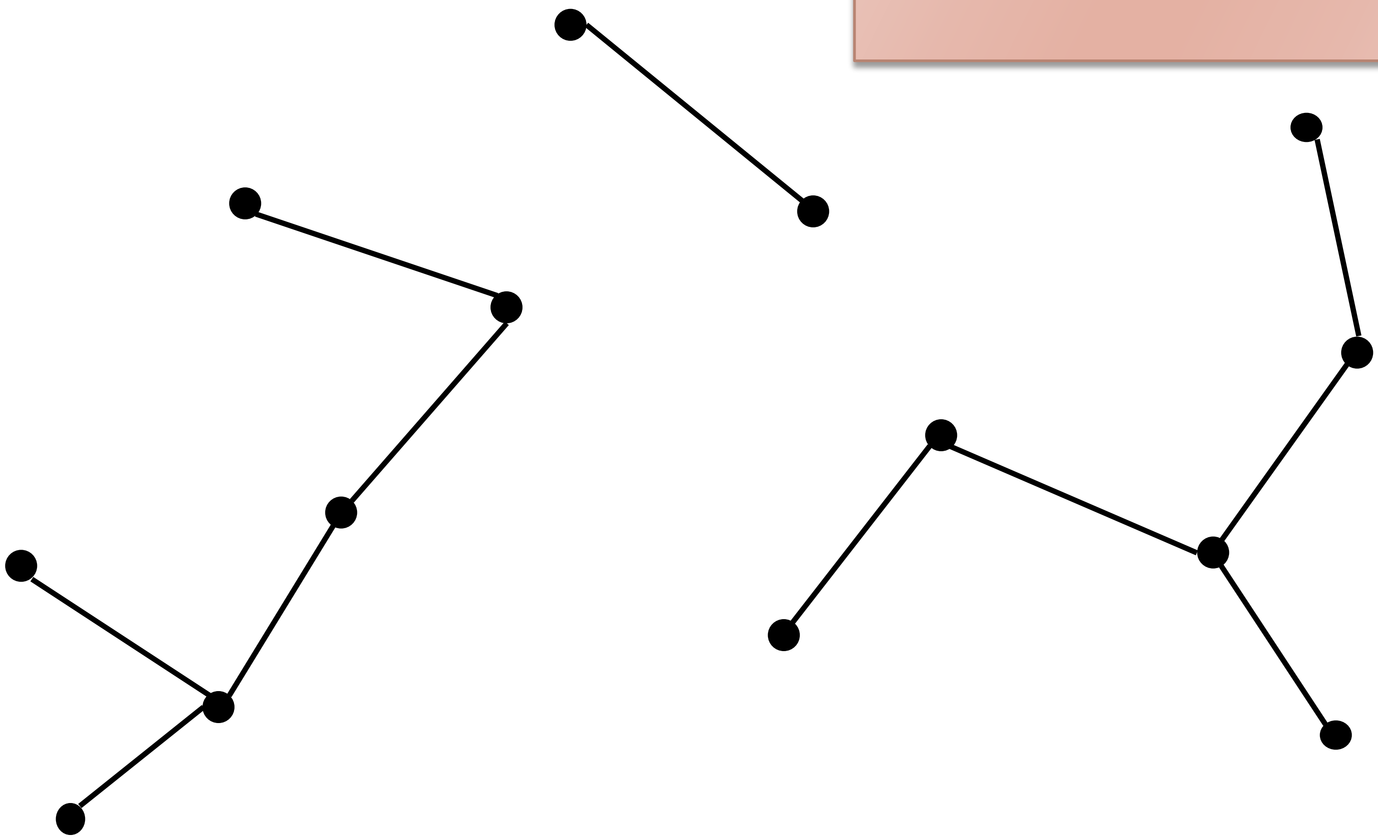
Example

Tree



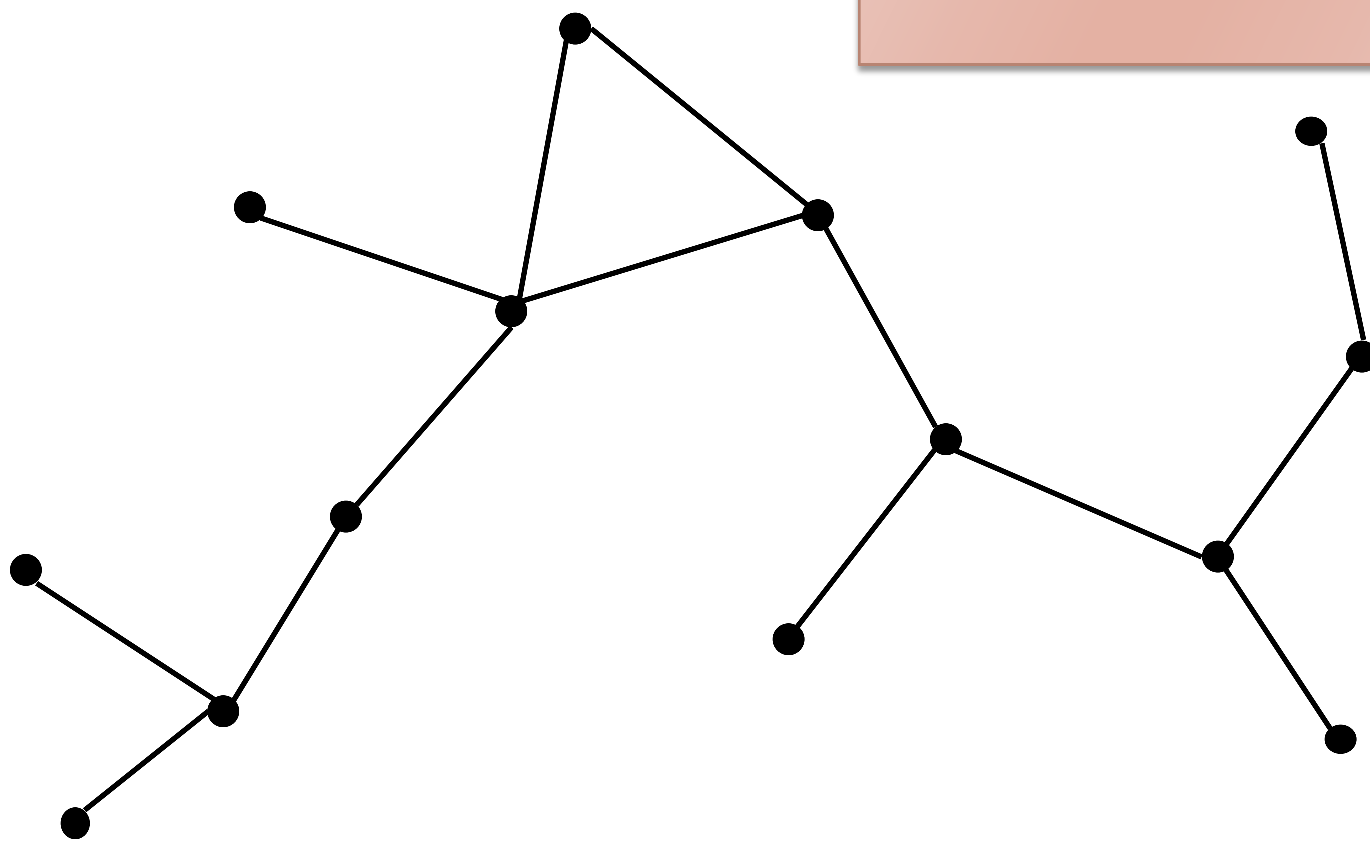
Example

Forest



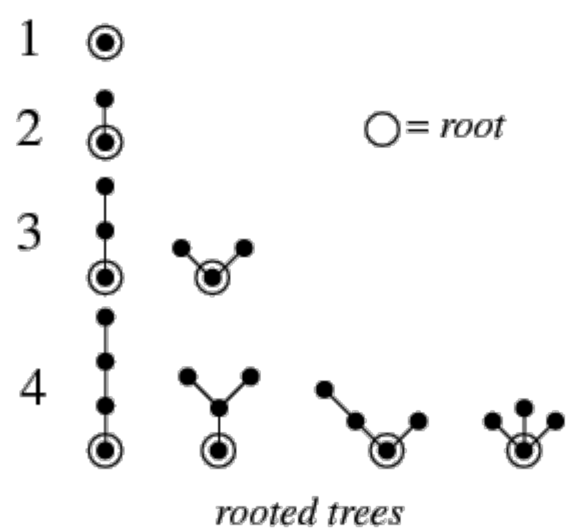
Example

This is not a tree nor a forest
(it contains a cycle)



Rooted trees

- In a tree, a special node may be singled out
- This node is called the "**root**" of the tree
- Any node of a tree can be the root

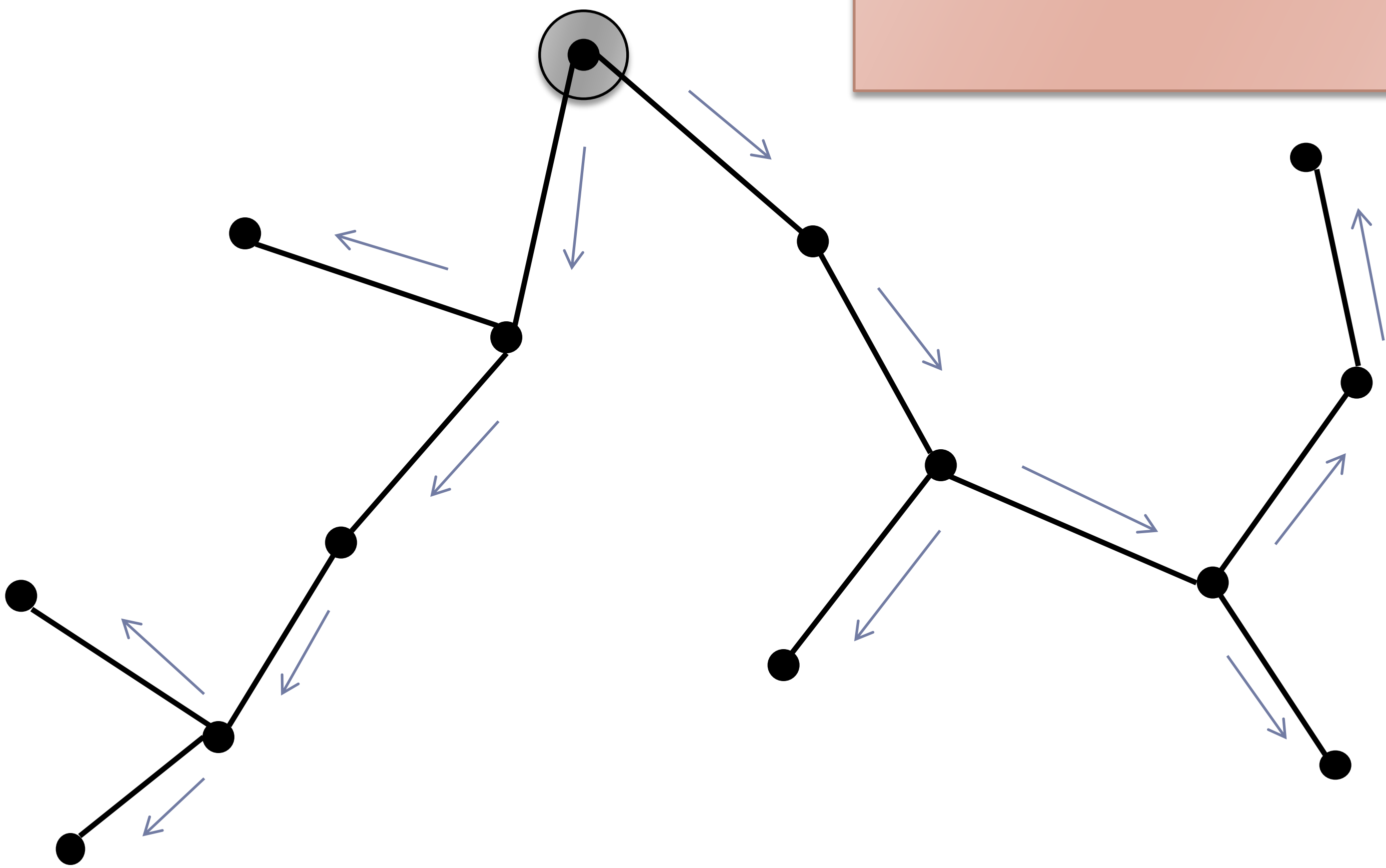


Tree (implicit) ordering

- The root node of a tree **induces an ordering** of the nodes
- The root is the "ancestor" of all other nodes/vertices
  - "children" are "away from the root"
  - "parents" are "towards the root"
- The root is the only node without parents
- All other nodes have exactly one parent
- The furthermost (children-of-children-of-children…) nodes are "leaves"

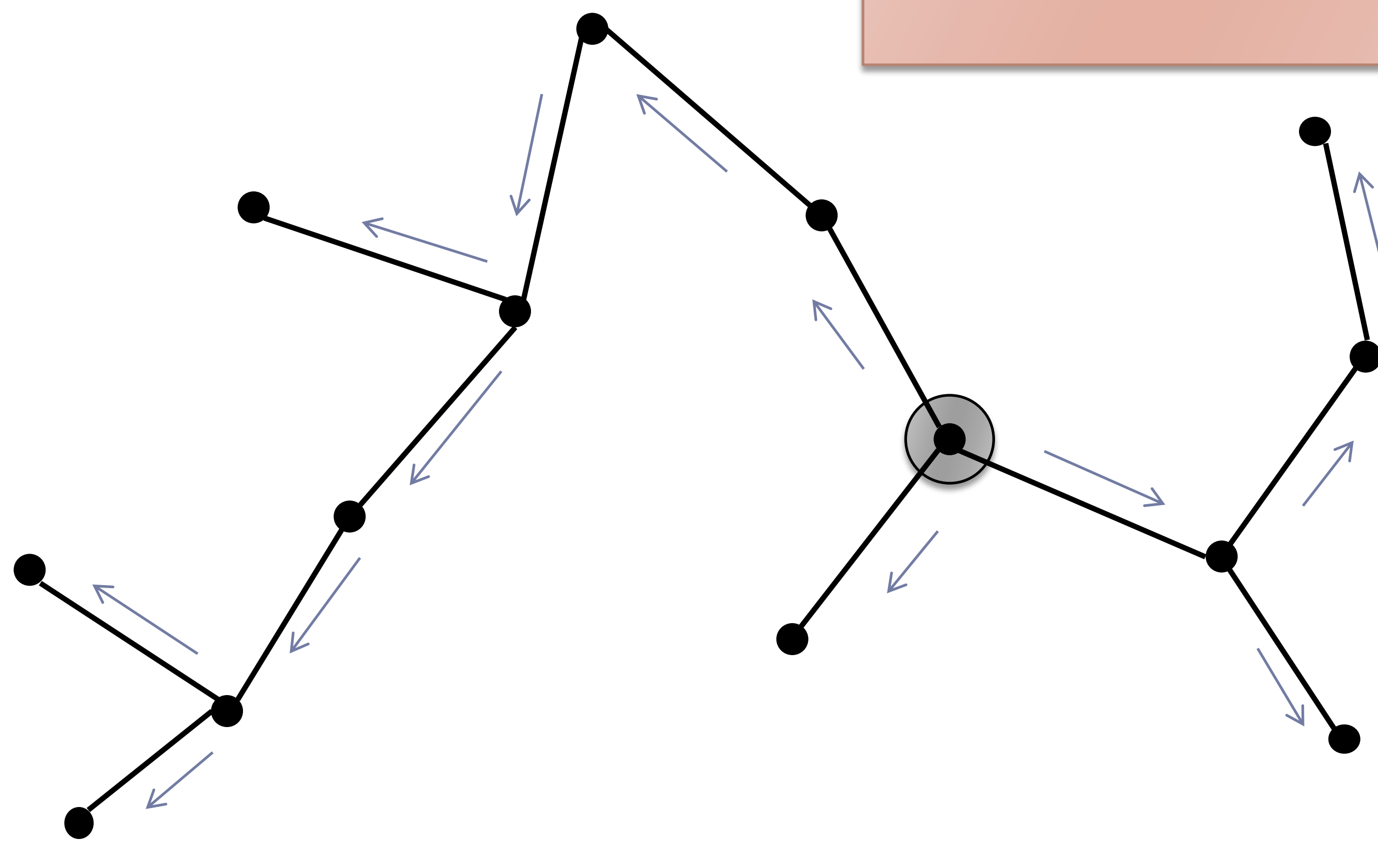
Example

Rooted Tree



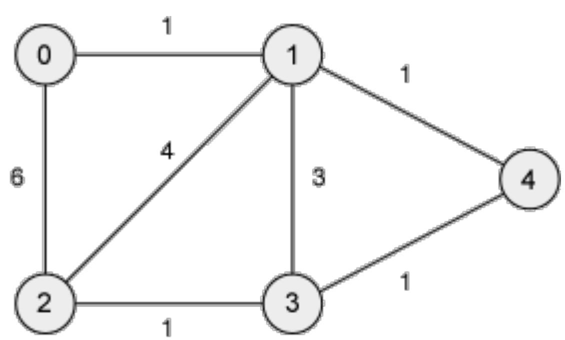
Example

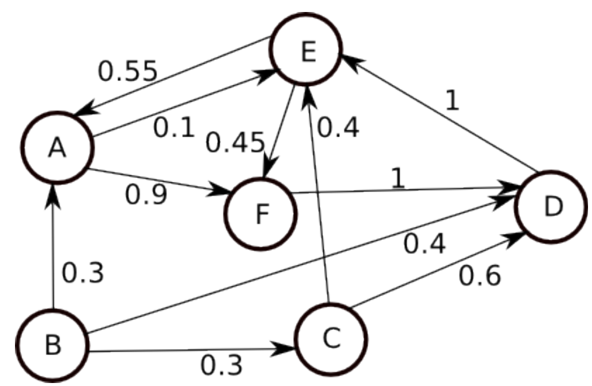
Rooted Tree



Weighted graphs

- A weighted graph is a graph in which each branch (edge) is given a numerical weight.
- A weighted graph is therefore a special type of labeled graph in which the labels are numbers (which are usually taken to be positive).





Applications

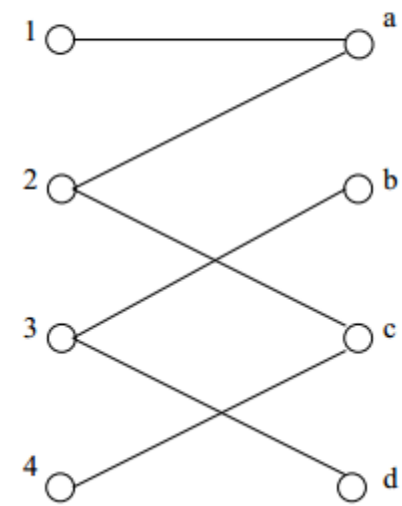
Introduction to Graphs

Graph applications

- Graphs are everywhere
  - Facebook friends (and posts, and 'likes')
  - Football tournaments (complete subgraphs + binary tree)
  - Google search index (V=page, E=link, w=pagerank)
  - Web analytics (site structure, visitor paths)
  - Car navigation (GPS)
  - Market Matching

Market matching

- H = Houses (1, 2, 3, 4)
- B = Buyers (a, b, c, d)
- V = H ∪ B
- Edges: $(h, b) \in E$ if *b* would like to buy *h*
- Problem: can all houses be sold and all buyers be satisfied?
- Variant: if the graph is weighted with a purchase offer, what is the most convenient solution?
- Variant: consider a 'penalty' for unsold items



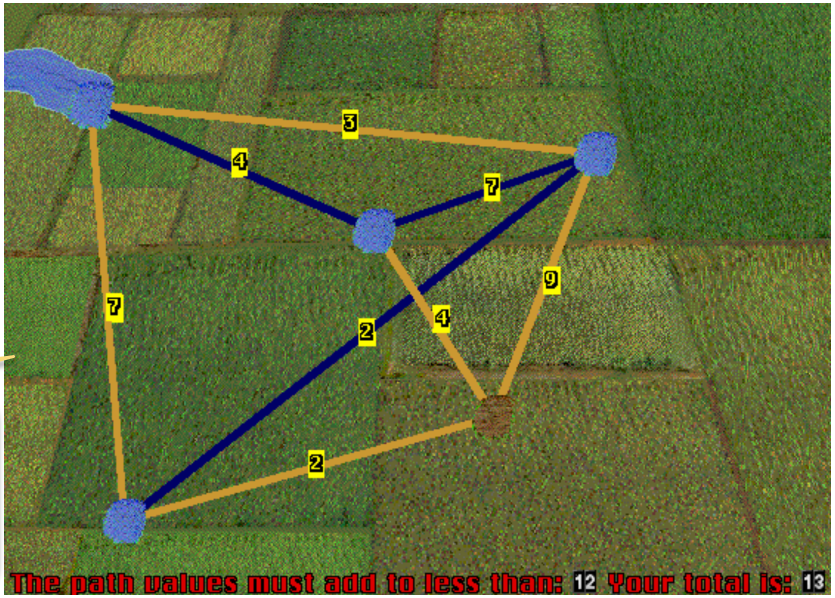
This graph is called "bipartite": H ∩ B = ∅

Connecting cities

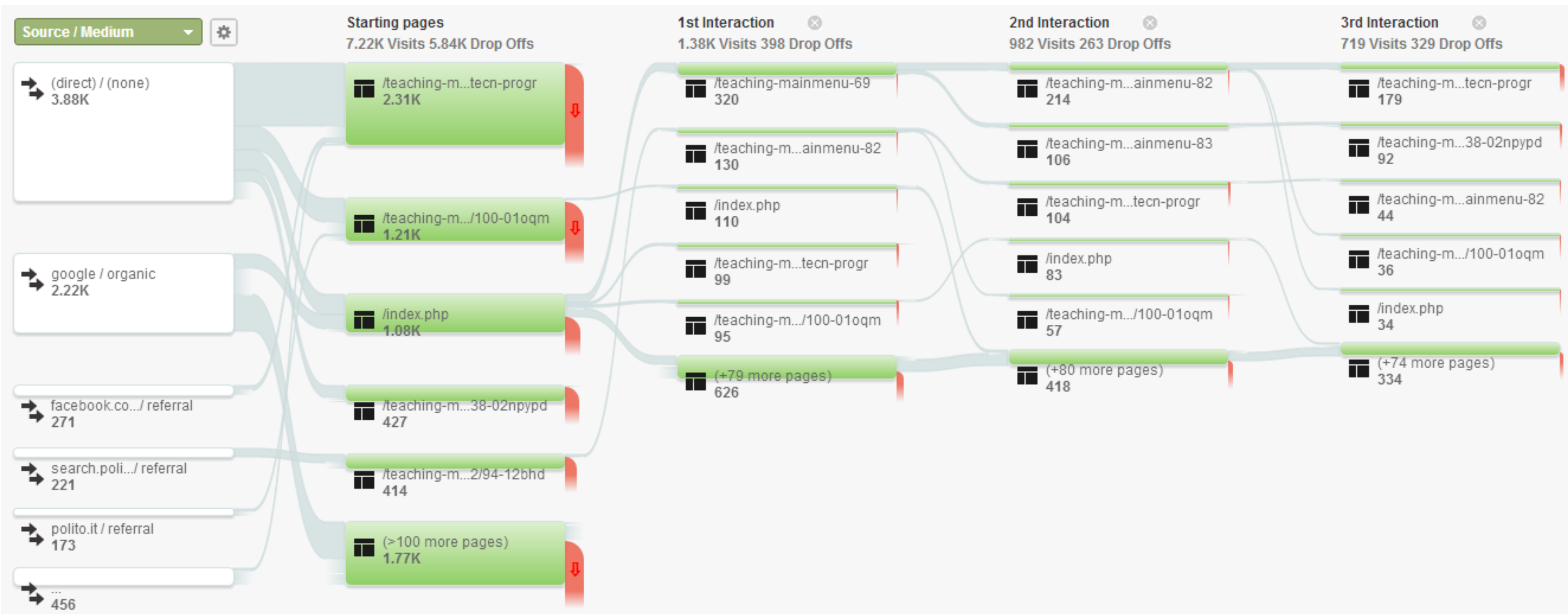
- We have a water reservoir
- We need to serve many cities
  - Directly or indirectly
- What is the most efficient set of inter-city water connections?

- Also for telephony, gas, electricity, …

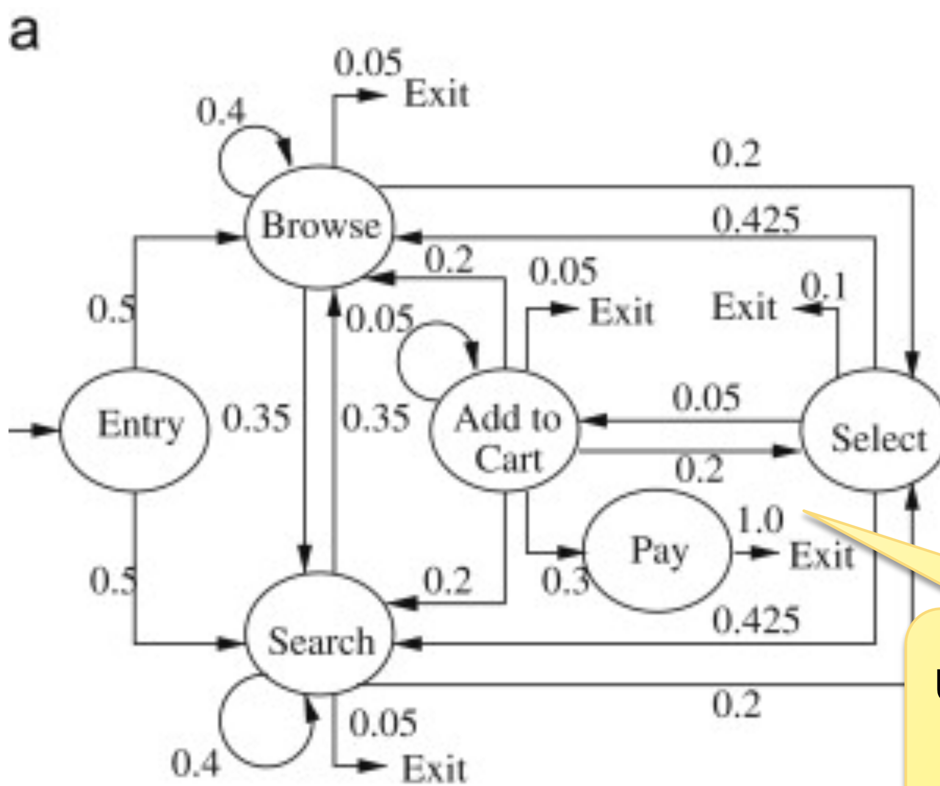
We are searching for the "minimum spanning tree"



Google Analytics (Visitors Flow)



Customer behavior



User actions encoded as frequencies

Street navigation

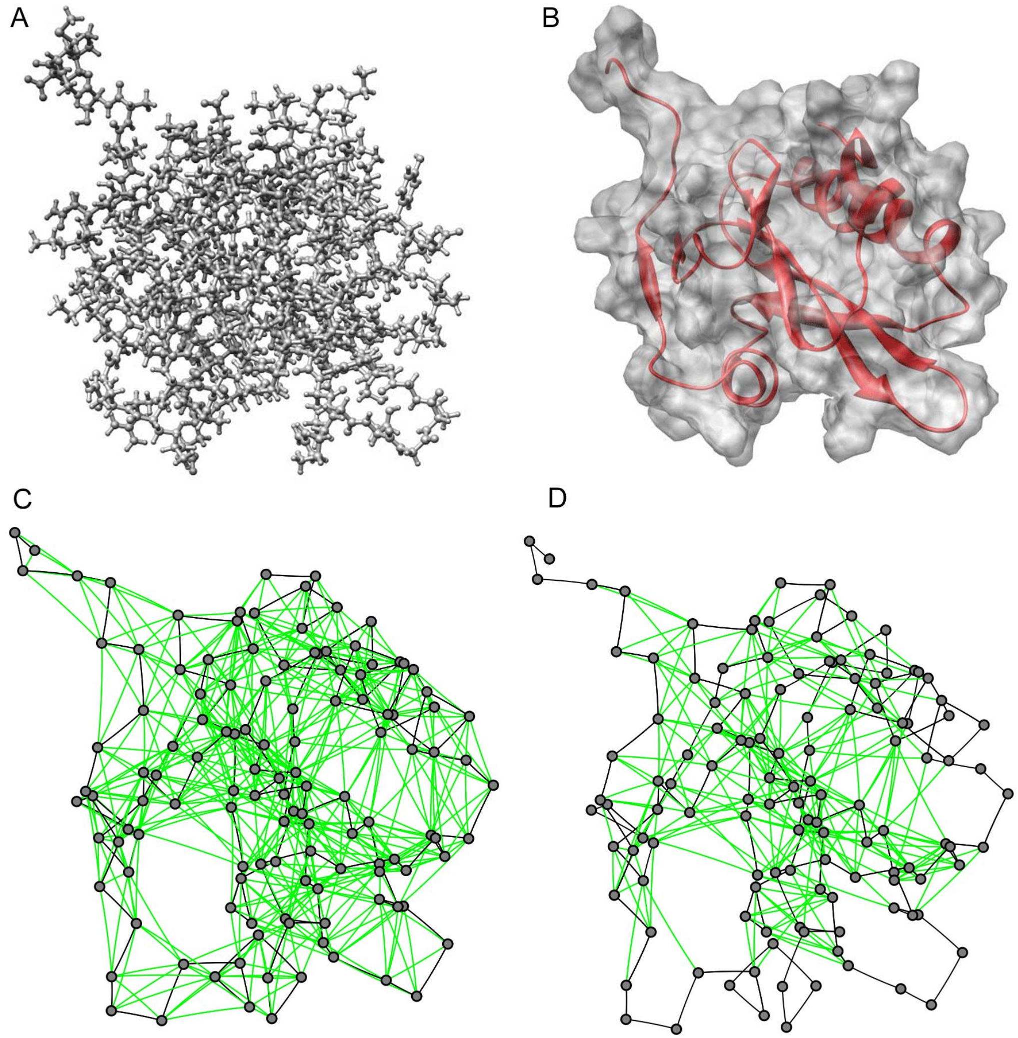
TSP: The traveling salesman problem

We must find a "Hamiltonian cycle"

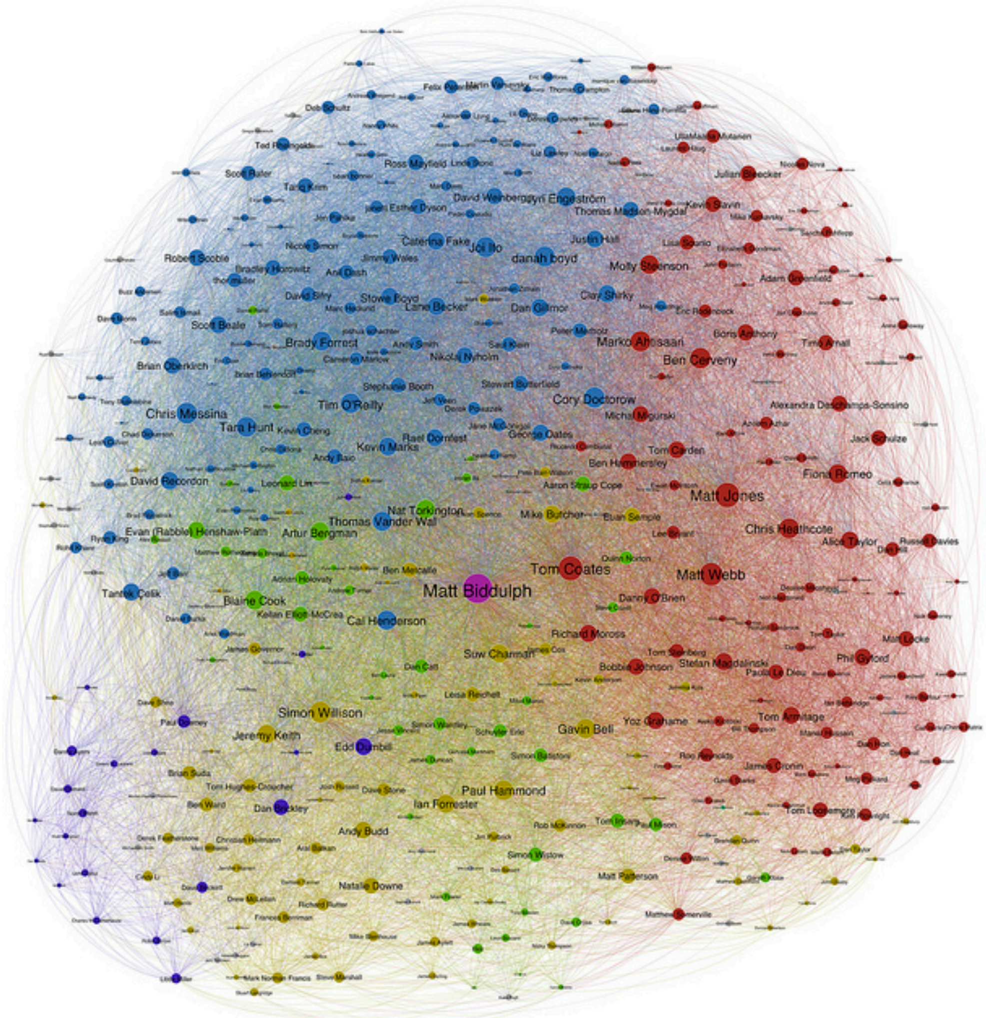
Train maps


Kensal Rise
Brondesbury Park
Harlesden
Kensal Green
Queen's Park
Brondesbury
West Hampstead
Finchley Road
Swiss Cottage
St. John's Wood
Chalk Farm
Camden Town
Kentish Town
Caledonian Road
Camden Road
Holloway Road
Drayton Park
Highbury & Islington
Caledonian Road & Barnsbury
Canonbury
Hackney Central
Mornington Crescent
King's Cross St. Pancras
Kilburn Park
Maida Vale
Warwick Avenue
Royal Oak
Westbourne Park
Ladbroke Grove
Edgware Road
Marylebone
Baker Street
Great Portland Street
Euston
Angel
Essex Road
Dalston Kingsland
Farringdon
Barbican
Old Street
Liverpool Street
Bethnal Green
Paddington
Warren Street
Euston Square
Regent's Park
Goodge Street
Russell Square
Bayswater
Moorgate
Shoreditch
White City
Holland Park
Marble Arch
Queensway
Oxford Circus
Holborn
Aldgate East
Notting Hill Gate
Lancaster Gate
Bond Street
Tottenham Court Road
Chancery Lane
St. Paul's
Shepherd's Bush
Bank
Covent Garden
Aldgate
Kensington (Olympia)
High Street Kensington
Green Park
Leicester Square
City Thameslink
Cannon Street
Shadwell
Goldhawk Road
Hyde Park Corner
Knightsbridge
Piccadilly Circus
Monument
Tower Hill
Tower Gateway
Wapping
Gloucester Road
Charing Cross
Blackfriars
Temple
Mansion House
Barons Court
Sloane Square
Westminster
Embankment
River Thames
Rotherhithe
Ravenscourt Park
West Kensington
Earl's Court
South Kensington
Victoria
St. James's Park
Bermondsey
Canada Water
West Brompton
Southwark
London Bridge
Waterloo
Fulham Broadway
Surrey Quays
Parsons Green
Pimlico
Lambeth North
Borough

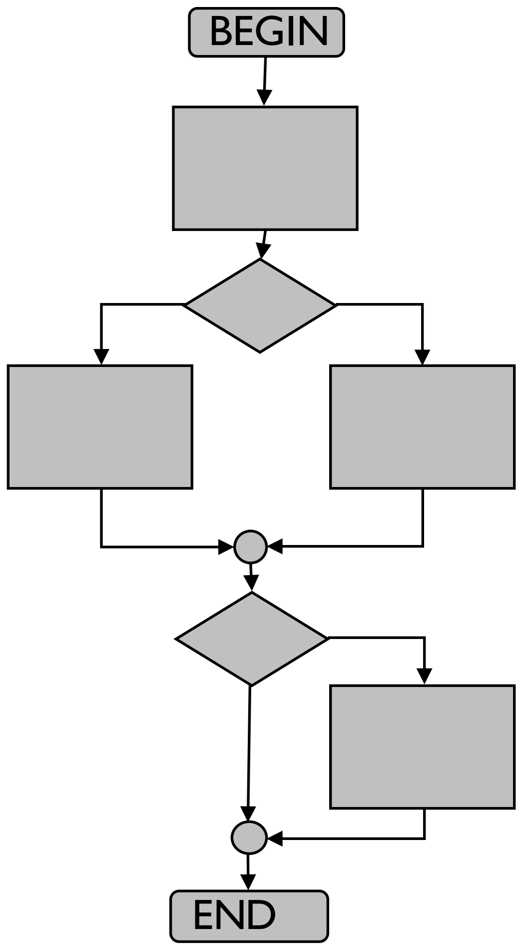
Chemistry (Protein folding)

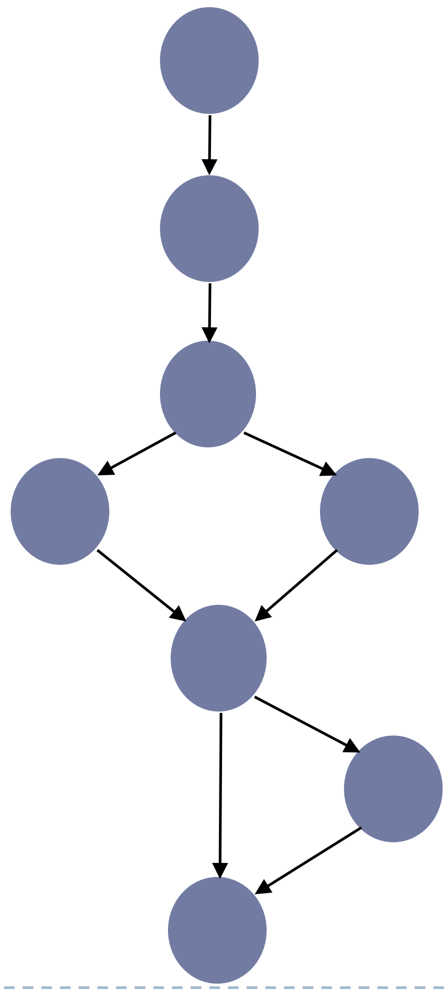


Facebook friends



Flow chart





Licenza d’uso

- Queste diapositive sono distribuite con licenza Creative Commons “Attribuzione - Non commerciale - Condividi allo stesso modo (CC BY-NC-SA)”
- Sei libero:
  - di riprodurre, distribuire, comunicare al pubblico, esporre in pubblico, rappresentare, eseguire e recitare quest'opera
  - di modificare quest'opera
- Alle seguenti condizioni:
  - **Attribuzione** — Devi attribuire la paternità dell'opera agli autori originali e in modo tale da non suggerire che essi avallino te o il modo in cui tu usi l'opera.
  - **Non commerciale** — Non puoi usare quest'opera per fini commerciali.
  - **Condividi allo stesso modo** — Se alteri o trasformi quest'opera, o se la usi per crearne un'altra, puoi distribuire l'opera risultante solo con una licenza identica o equivalente a questa.
- http://creativecommons.org/licenses/by-nc-sa/3.0/