



上海交通大学

SHANGHAI JIAO TONG UNIVERSITY



上海交通大学

约翰·霍普克罗夫特
计算机科学中心

John Hopcroft Center for Computer Science

CS 445: Combinatorics

Shuai Li

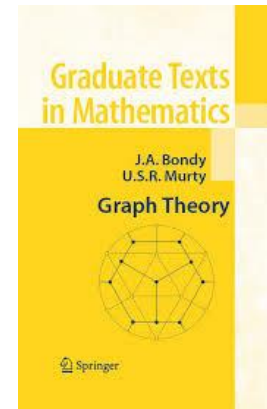
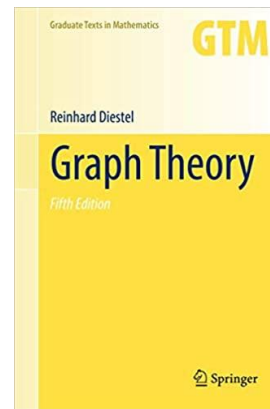
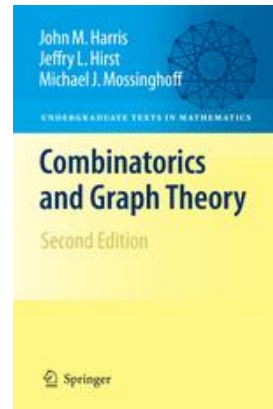
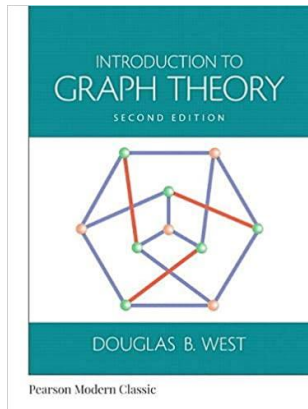
John Hopcroft Center, Shanghai Jiao Tong University

<https://shuaili8.github.io>

<https://shuaili8.github.io/Teaching/CS445/index.html>

References

- References:
 - Introduction to Graph Theory, by Douglas West
 - Combinatorics and Graph Theory, by Harris, Hirst and Mossinghoff
 - Graph Theory, Reinhard Diestel
 - Graph Theory, by Bondy and Murty
 - A Course in Combinatorics, J. H. Van Lint



Previous courses

- Discrete Mathematics
 - Basic concepts for graph theory
- Mathematical Foundations of Computer Science (CS 499)
 - Basic notions and hand shaking lemma
 - Graph isomorphism and graph score
 - Applications of handshake lemma: Parity argument
 - The number of spanning trees
 - Isomorphism of trees
 - Random graphs

Goal

- Knowledge of the basic problems for graph theory
 - Bipartite graphs/Matching/Coloring/Flows/...
- Knowledge of the important counting related results on graphs
- Familiar with the common proof techniques
- Awareness of the popular applications of graphs in many fields

Grading policy

- Attendance and participation: 5%
- Assignments: 35%
- Midterm exam: 15%
- Entry editing: 5%
- Reading report: 10%
- Final exam: 30%

Honor code

- Discussions are encouraged
- **Independently** write-up homework and project
- Same reports and homework will be **reported**

Teaching Assistant

- Fang Kong (孔芳)
 - Email: fangkong@sjtu.edu.cn
 - 2nd year PhD student
 - Research interests on bandit algorithms
 - Office hour: Thu 7-9 PM
- Ruofeng Yang (杨若峰)
 - Email: wanshuiyin@sjtu.edu.cn
 - Senior undergraduate student
 - Research interests on optimization
 - Office hour: Wed 7-9 PM

Course Outline

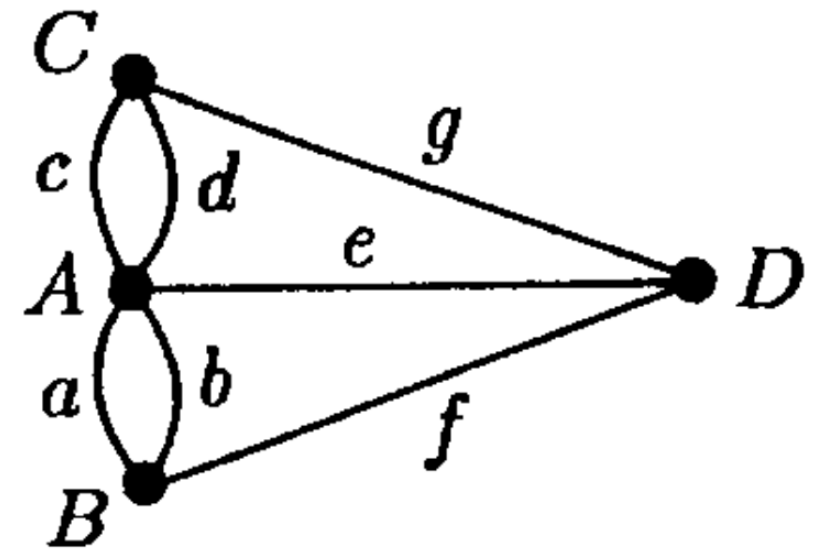
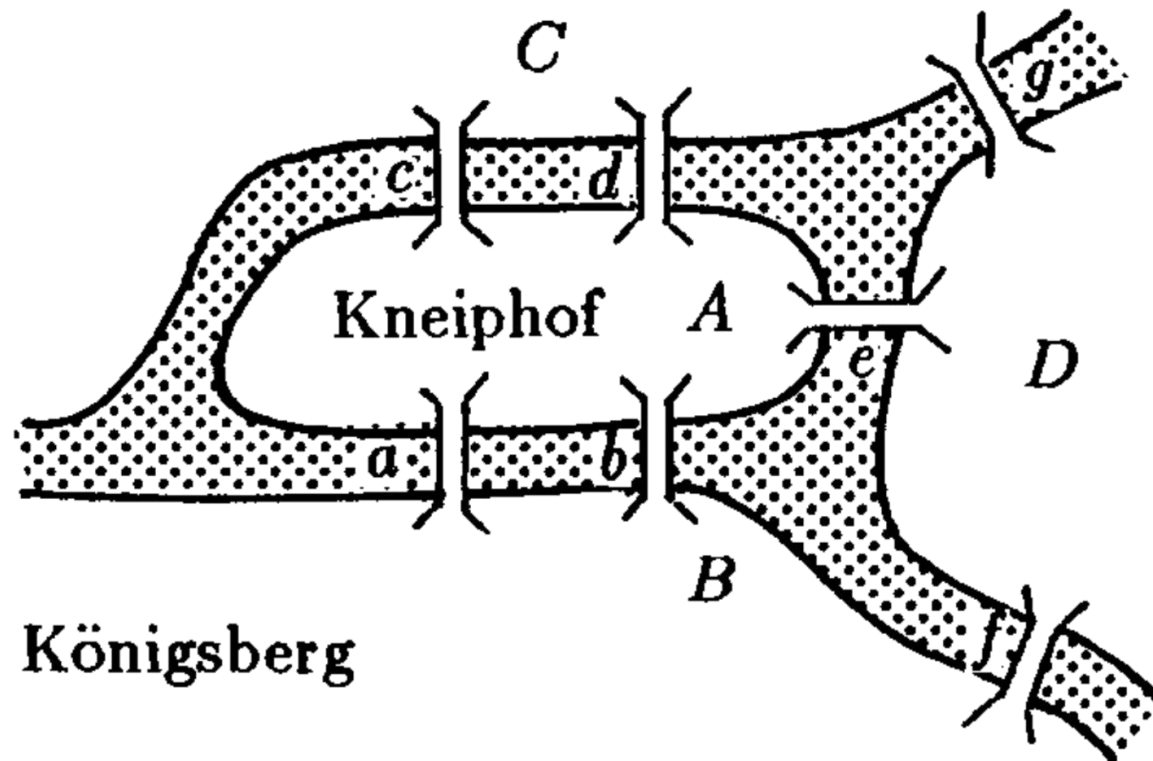
- Basics
 - Graphs, paths and cycles, connectivity, bipartite graphs
- Trees
- Matching
- Connectivity
- Planar Graphs
- Coloring
- Circuits

Introduction

Seven bridges of Königsberg 七桥问题

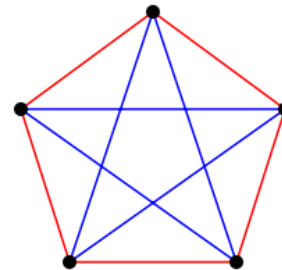
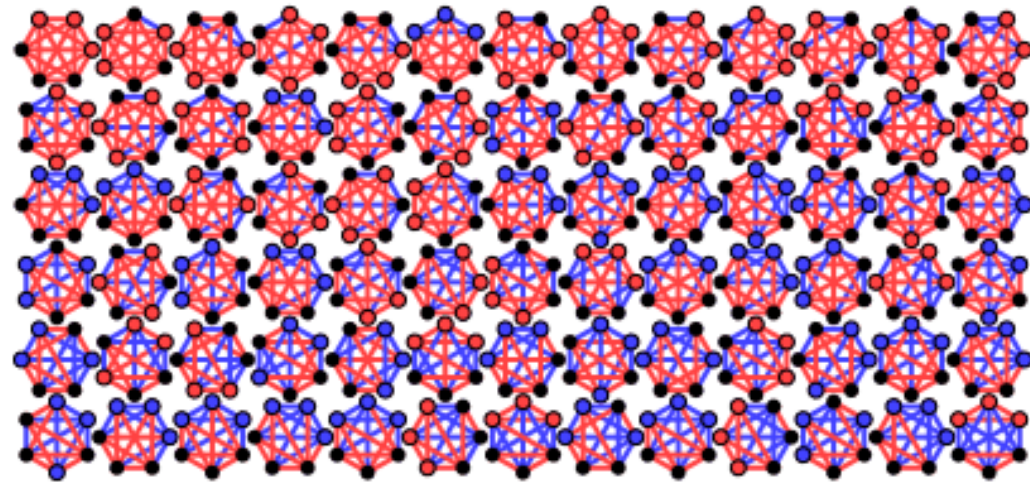
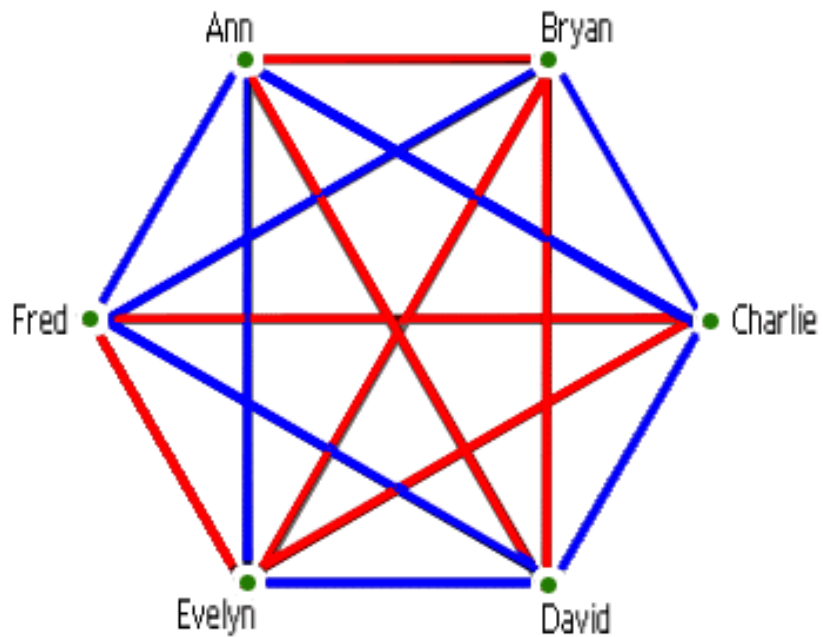


- Leonhard Euler 1736: Is it possible to make a walk through the city, returning to the starting point and crossing each bridge exactly once?



The friendship riddle

- Does every set of six people contain three mutual acquaintances or three mutual strangers?



$$R(3,3)=6$$

$$R(3,4)=R(4,3)=9$$

$$R(3,5)=R(5,3)=14$$

$$R(3,6)=R(6,3)=18$$

Examples of general combinatorics problems using graph theory

- Instant Insanity 四色方柱问题

- make a stack of these cubes so that all four colors appear on each of the four sides of the stack

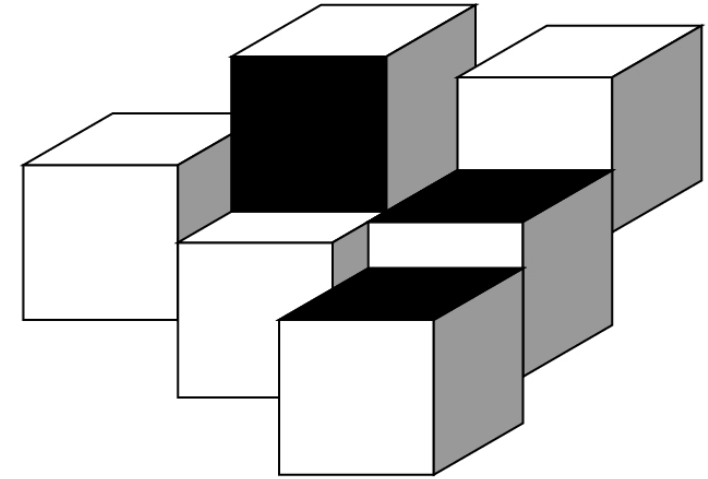
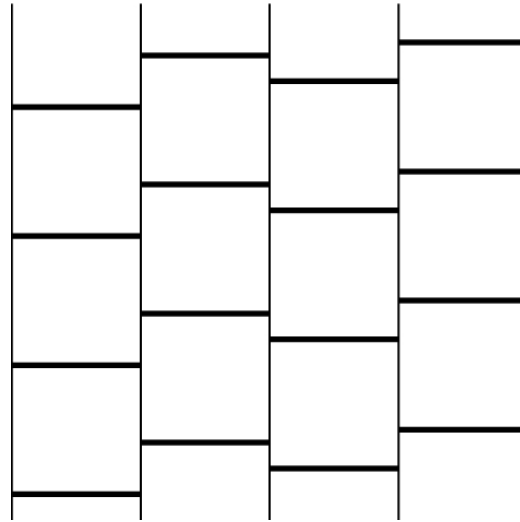


- A set problem

- Let A_1, \dots, A_n be n distinct subsets of the n -set $N := \{1, \dots, n\}$. Show that there is an element $x \in N$ such that the sets $A_i \setminus \{x\}$, $1 \leq i \leq n$, are all distinct

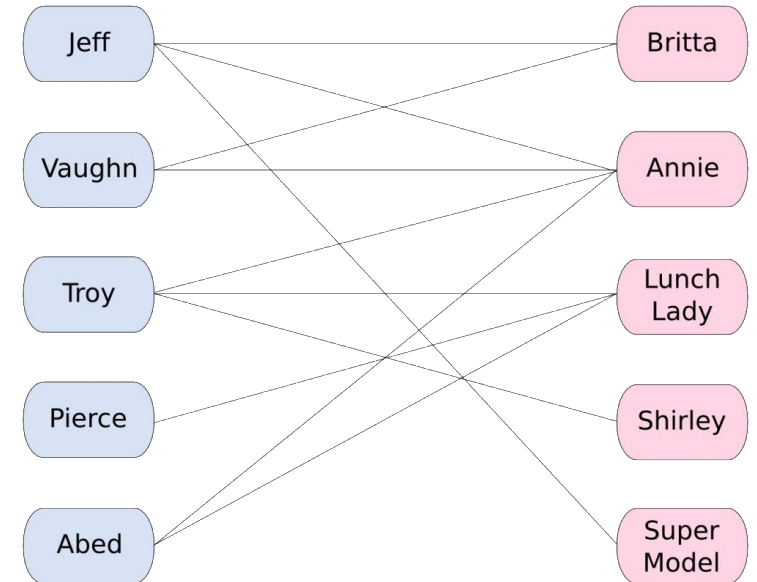
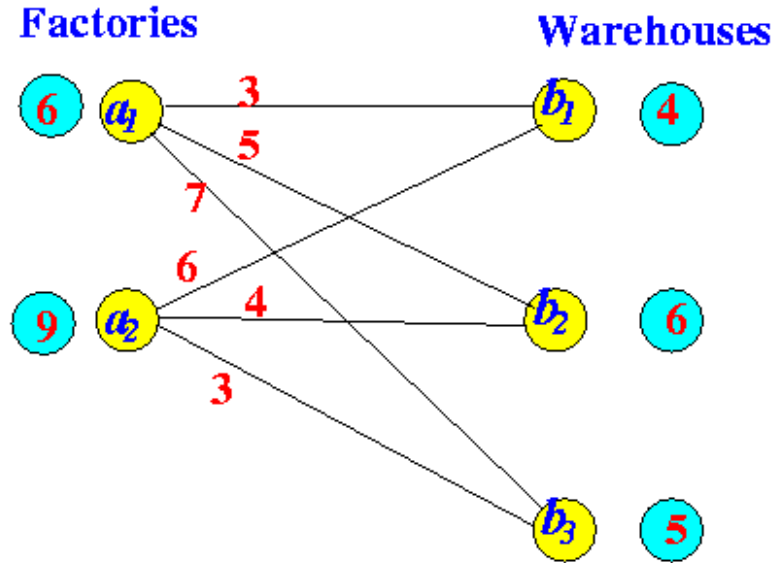
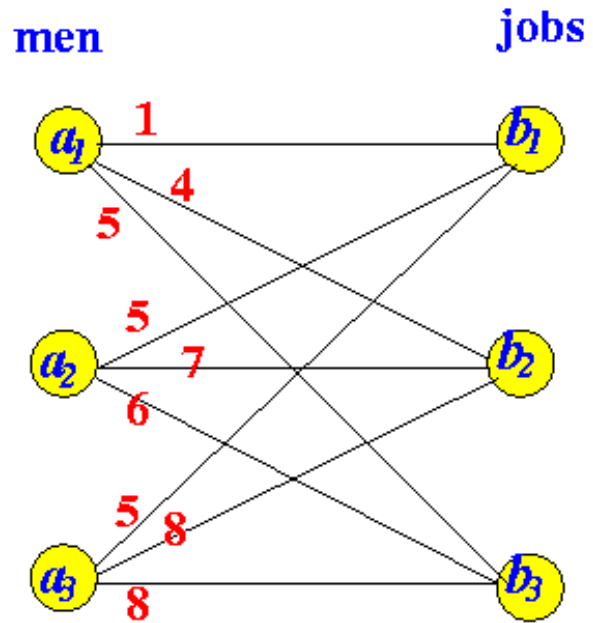
Keller's conjecture

- In 1930, Keller conjectured that any tiling of n -dimensional space by translates of the unit cube must contain a pair of cubes that share a complete $(n - 1)$ -dimensional face



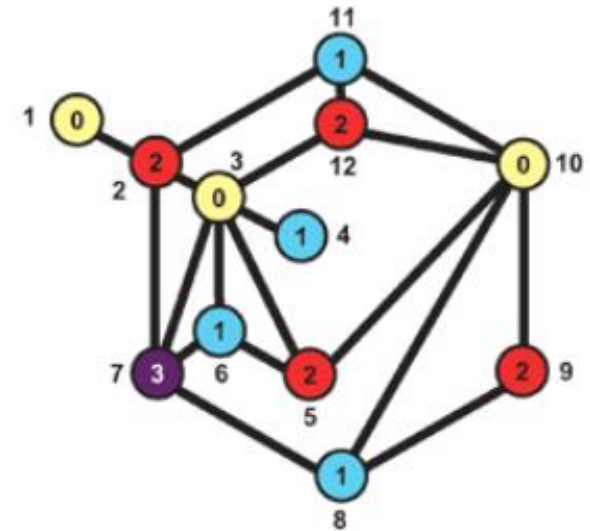
- Corrádi and Szabó transfer it into a graph theory problem
 - Constructing Keller graph
- The conjecture is solved by computer search recently (June 2020)

Assignment problems



Scheduling and coloring

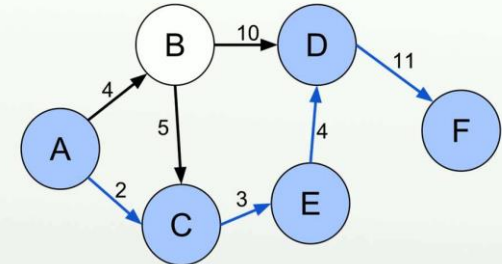
- University examination timetabling
 - Two courses linked by an edge if they have the same students
- Meeting scheduling
 - Two meetings are linked if they have same member



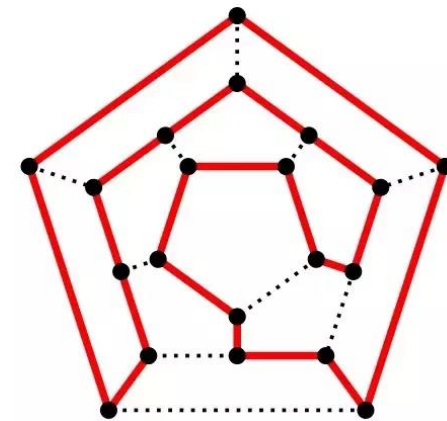
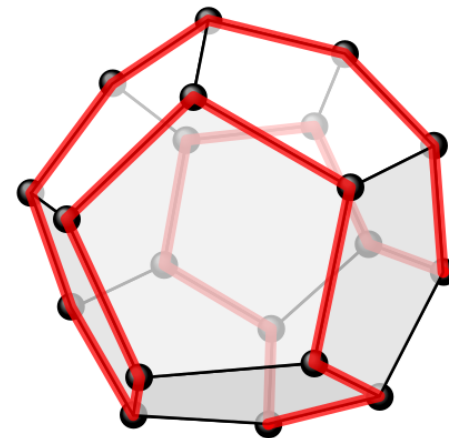
Routes in road networks

- How can we find the shortest route from A to F ?
- If the vertices of the graph represent our house and other places to visit, then we may want to follow a route that visits every vertex exactly once, so as to visit everyone without overstaying our welcome
 - Hamilton circuit

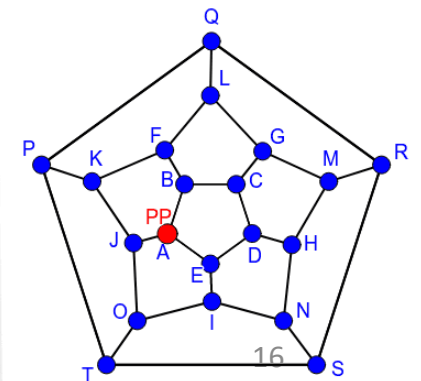
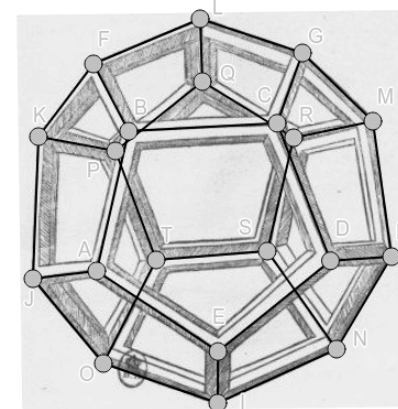
Shortest path problem



https://en.wikipedia.org/wiki/File:Shortest_path_with_direct_weights.svg

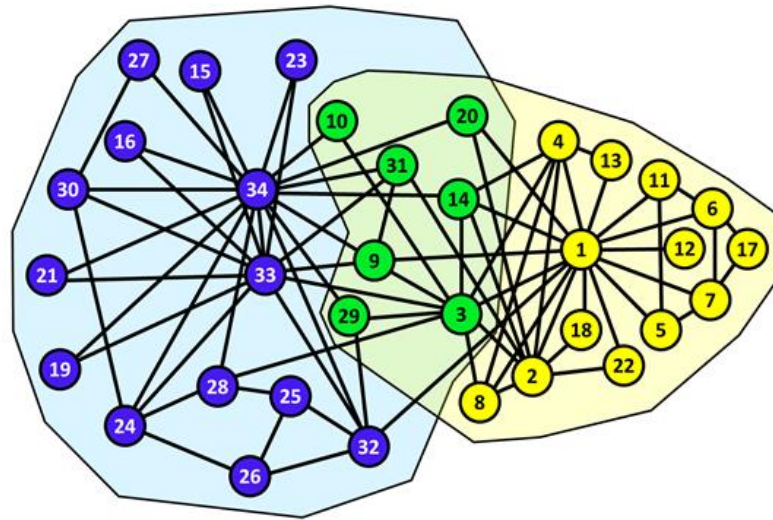
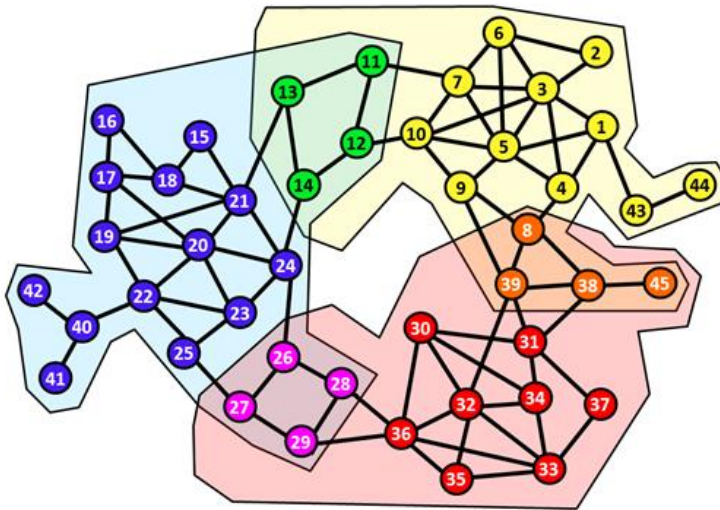
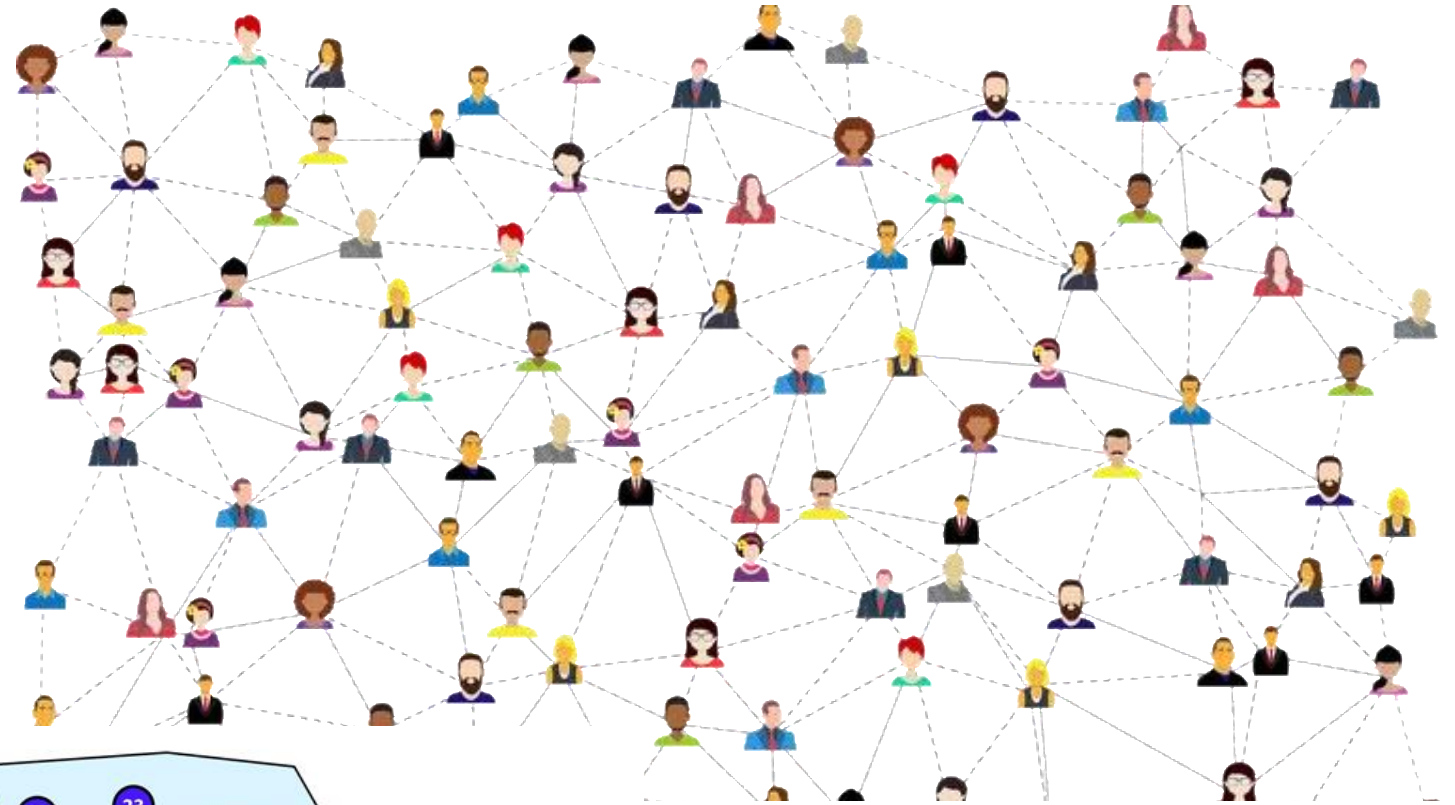


✓ Leonardo da Vinci: DVODECEDRON



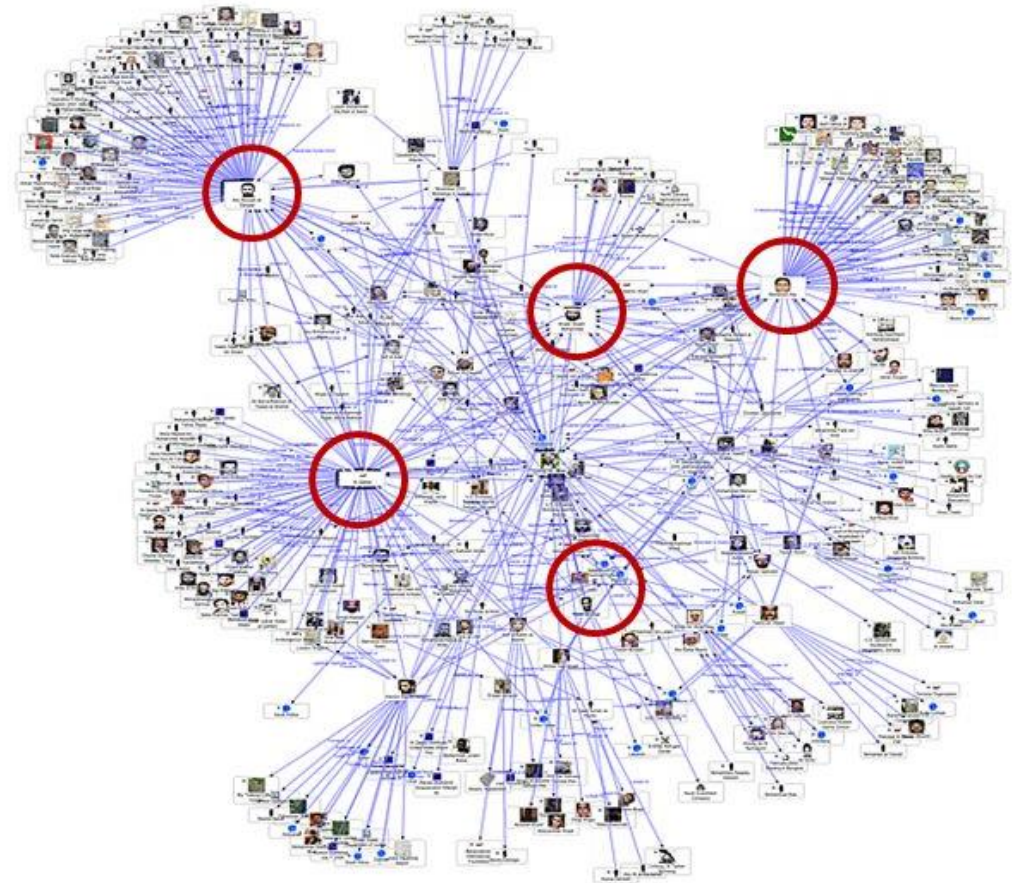
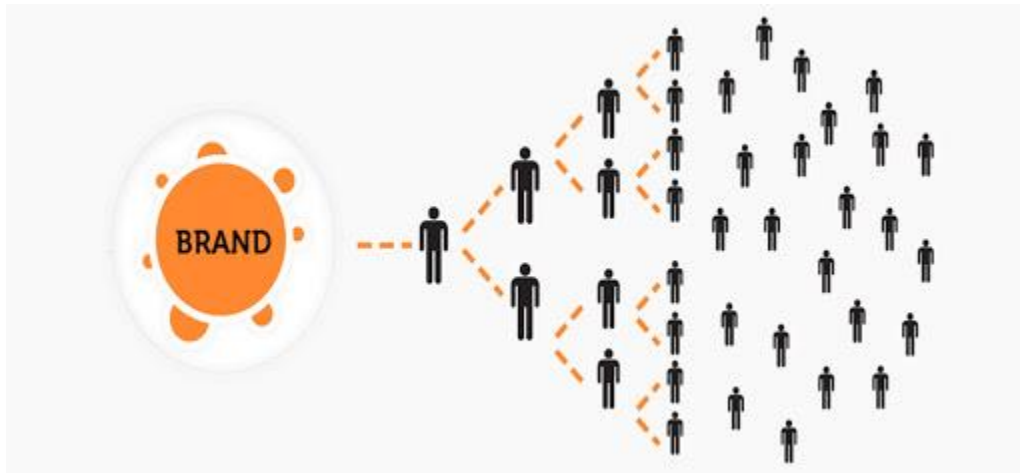
Social network

- Recommendation
- Clustering



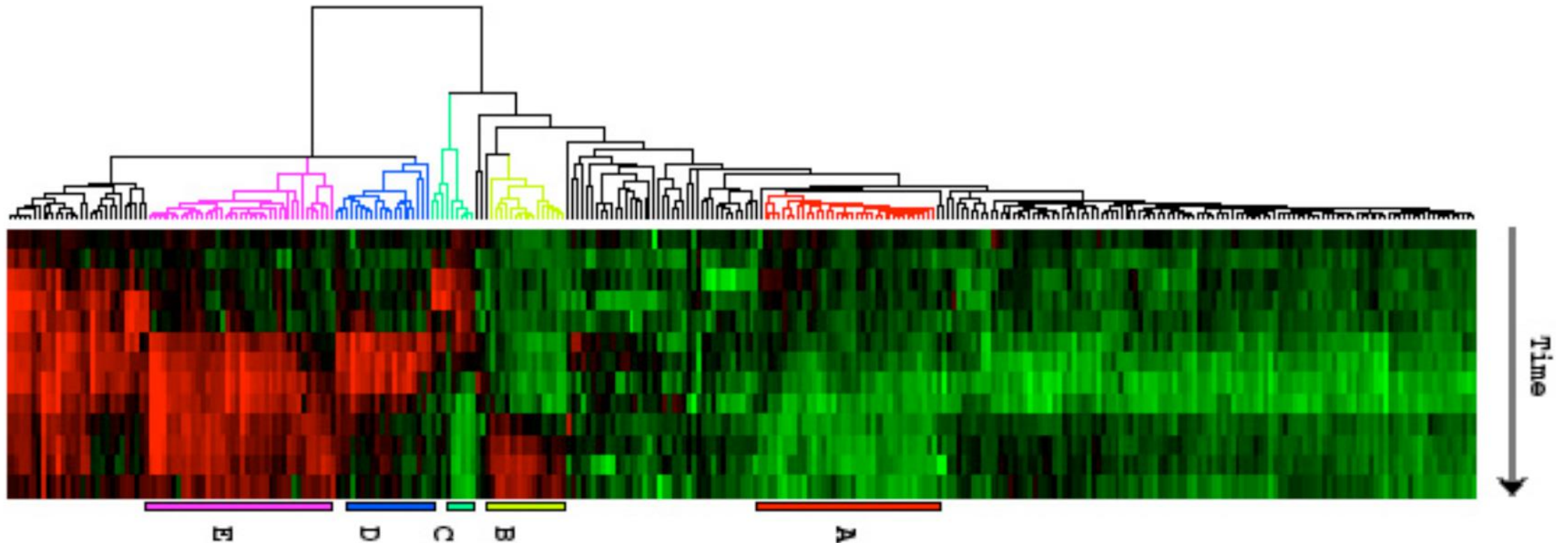
Influence maximization

- Select the best seed set

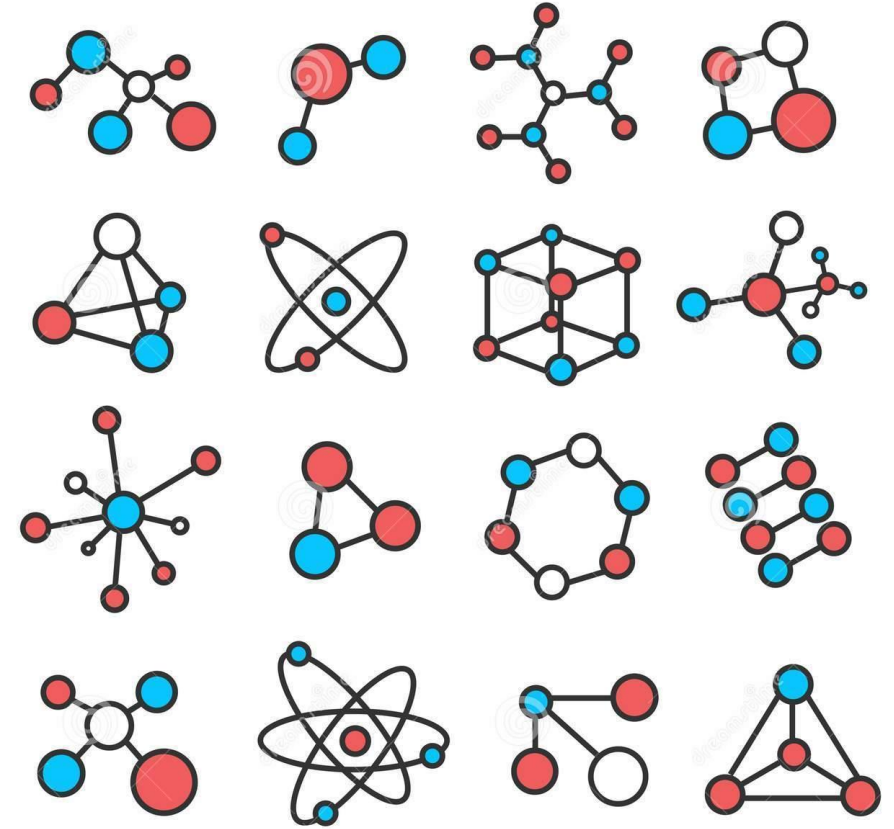
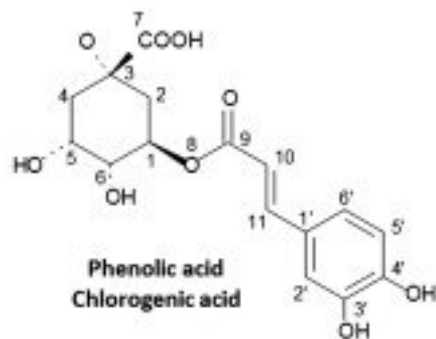
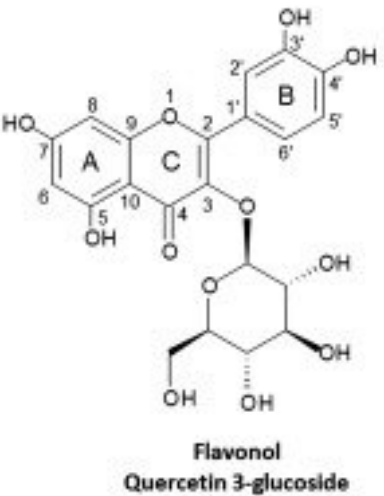
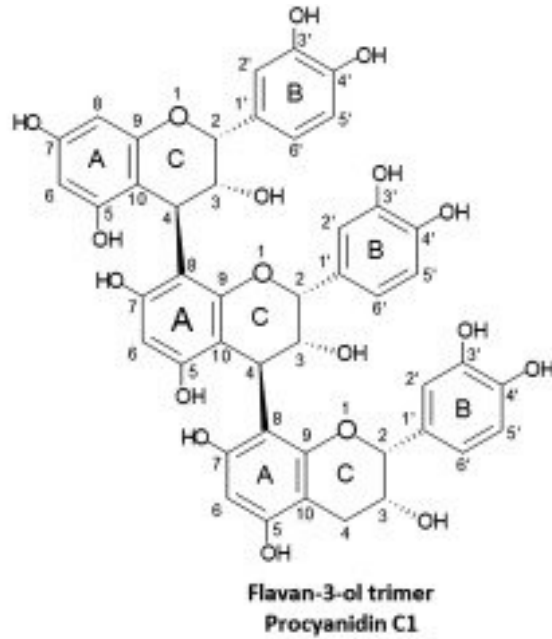
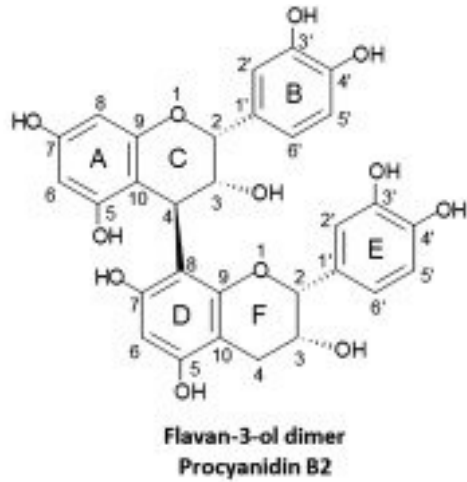


Gene structure

- Tree graph
 - Agglomerative clustering method



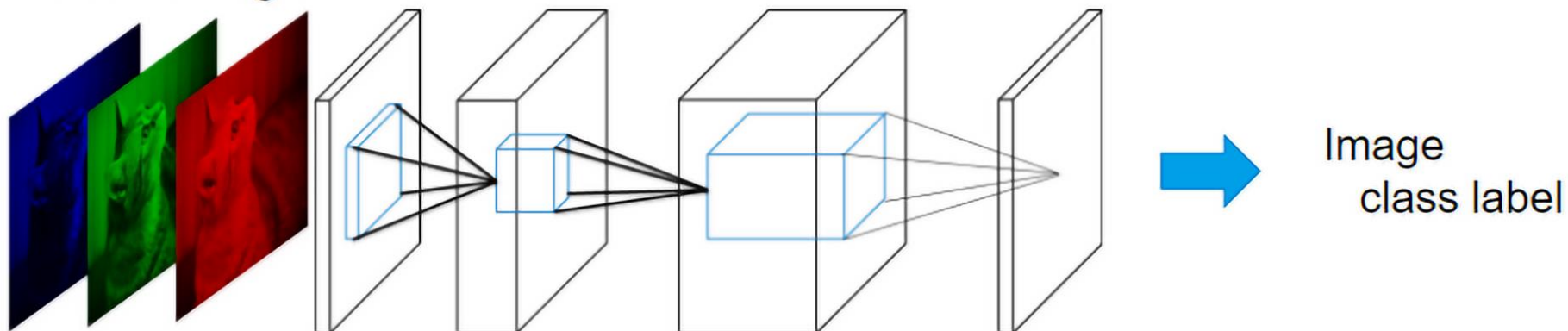
Molecular structure



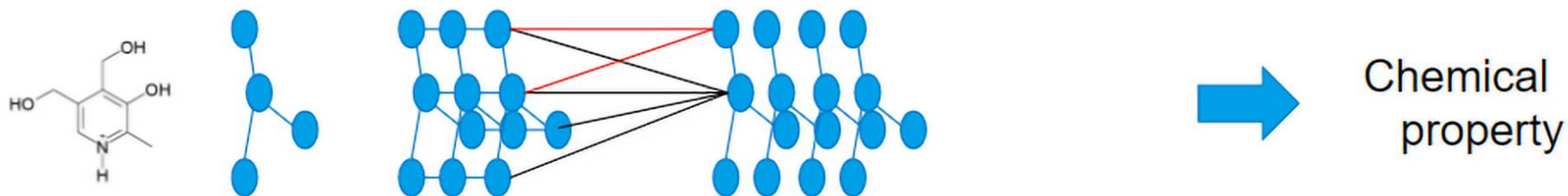
Graph neural network (GNN)

How Graph Convolutions work

CNN on image



Graph convolution

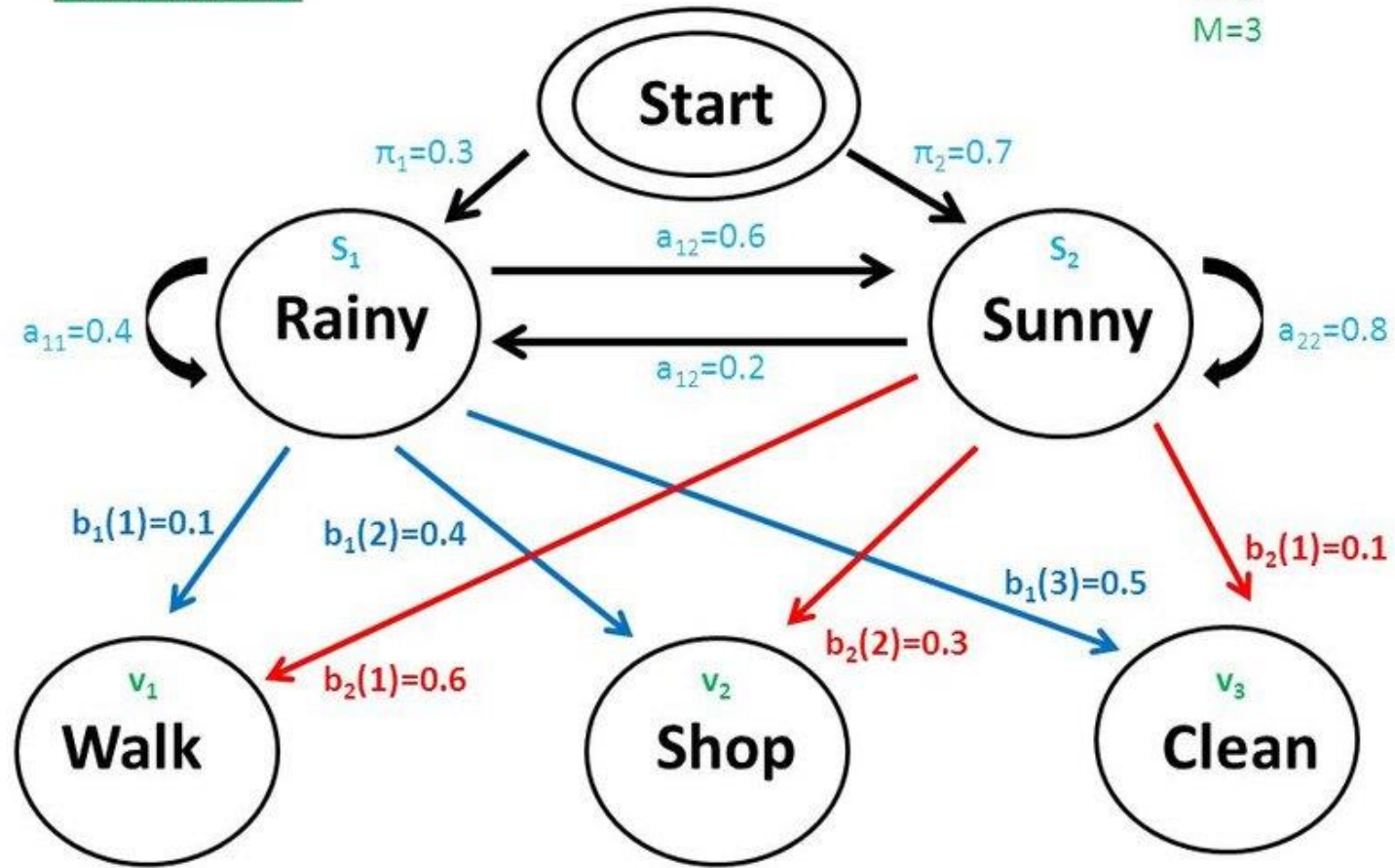


Convolution "kernel" depends on Graph structure

Hidden Markov Model

Example (cont):

$N=2$
 $M=3$



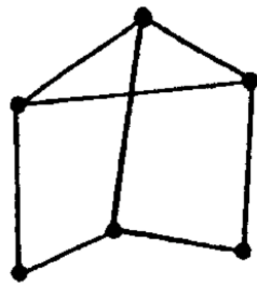
Basics

Graphs

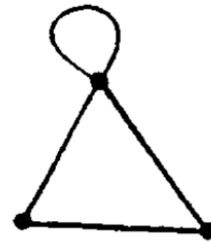
- **Definition** A graph G is a pair (V, E)
 - V : set of vertices
 - E : set of edges
 - $e \in E$ corresponds to a pair of endpoints $x, y \in V$

We mainly focus on
Simple graph:
No loops, no multi-edges

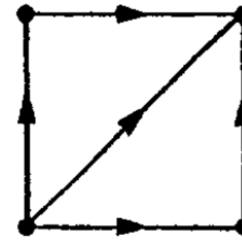
edge	ends
a	x, z
b	y, w
c	x, z
d	z, w
e	z, w
f	x, y
g	z, w



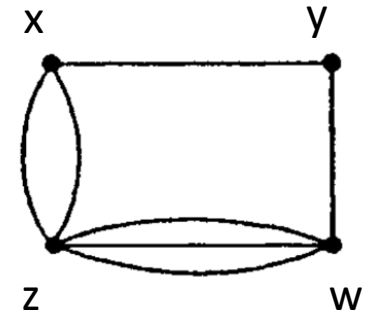
(i) graph



(ii) graph with loop



(iii) digraph



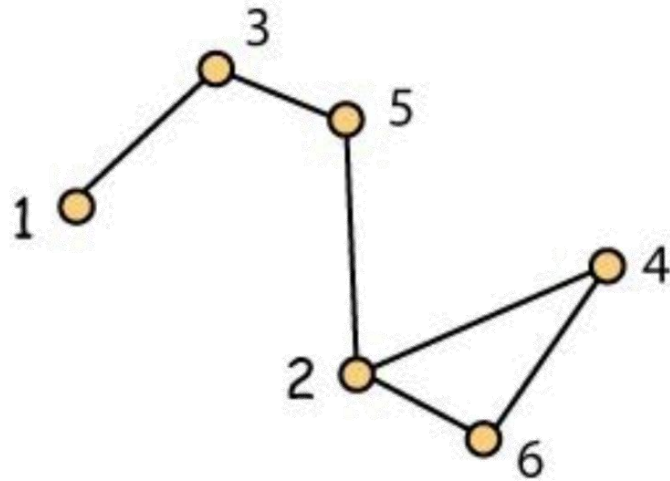
(iv) multiple edges

Figure 1.2

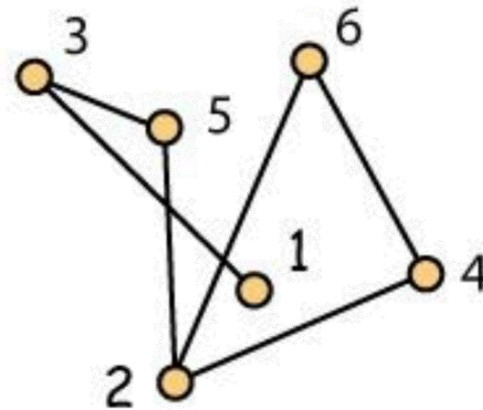
Figure 1.1

Graphs: All about adjacency

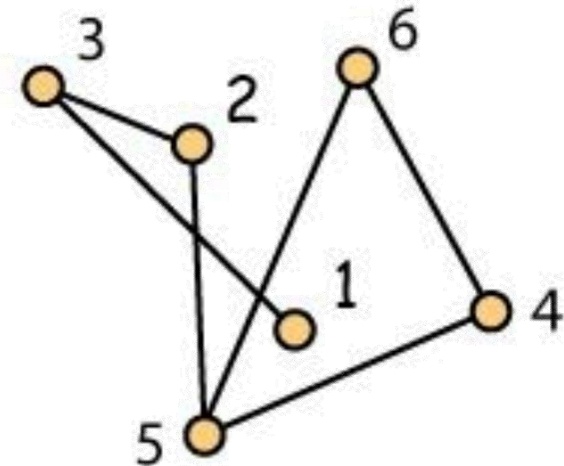
- Same graph or not



(a)



(b)

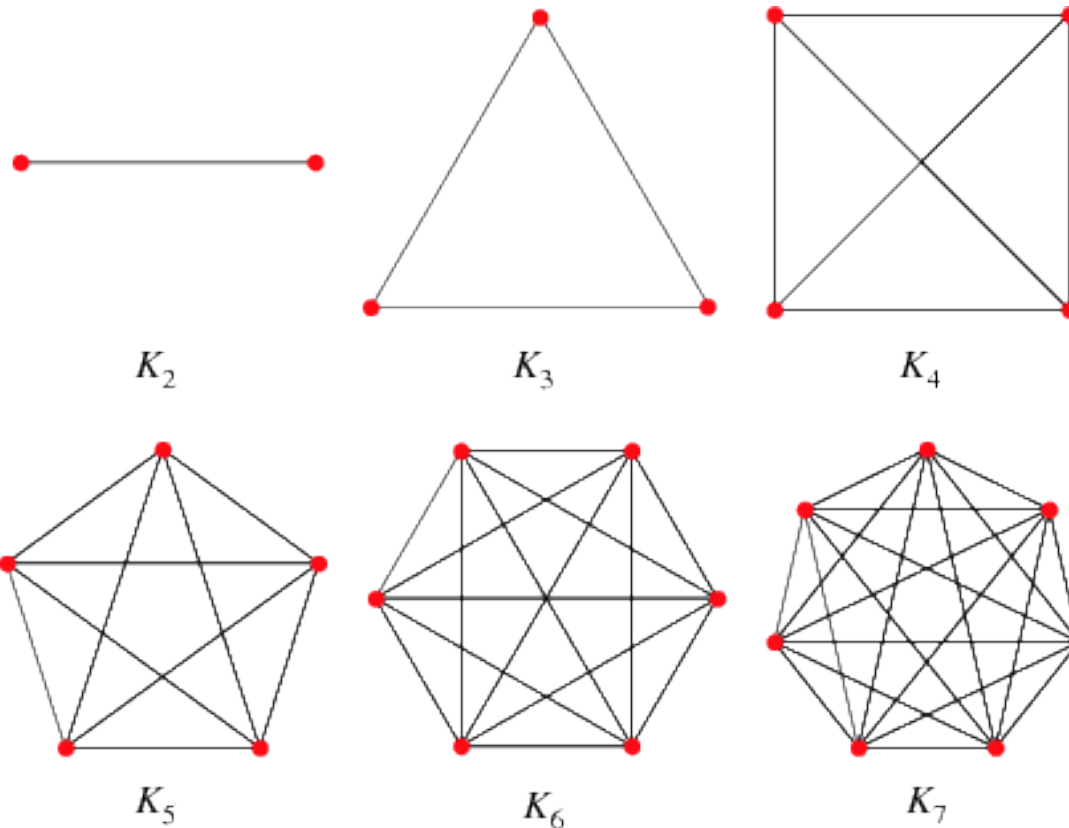


(c)

- Two graphs $G_1 = (V_1, E_1)$, $G_2 = (V_2, E_2)$ are **isomorphic** if there is a bijection $f: V_1 \rightarrow V_2$ s.t.
$$e = \{a, b\} \in E_1 \iff f(e) := \{f(a), f(b)\} \in E_2$$

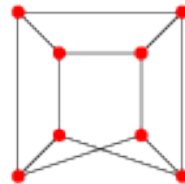
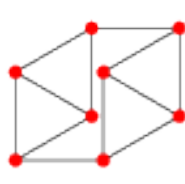
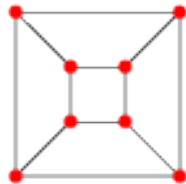
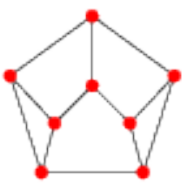
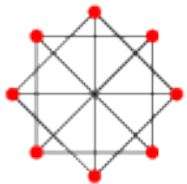
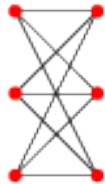
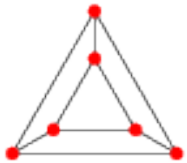
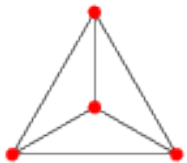
Example: Complete graphs

- There is an edge between every pair of vertices



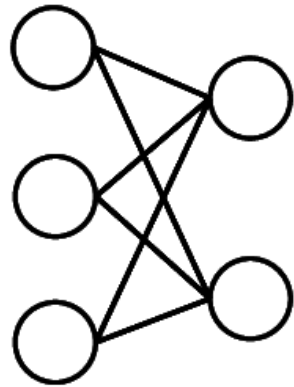
Example: Regular graphs

- Every vertex has the same degree

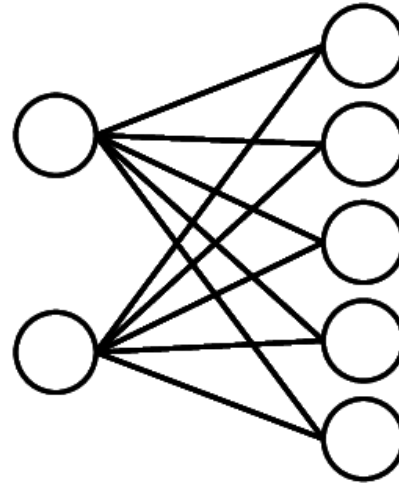


Example: Bipartite graphs

- The vertex set can be partitioned into two sets X and Y such that every edge in G has one end vertex in X and the other in Y
- Complete bipartite graphs



$K_{3,2}$



$K_{2,5}$

Example (1A, L): Peterson graph

- Show that the following two graphs are same/isomorphic

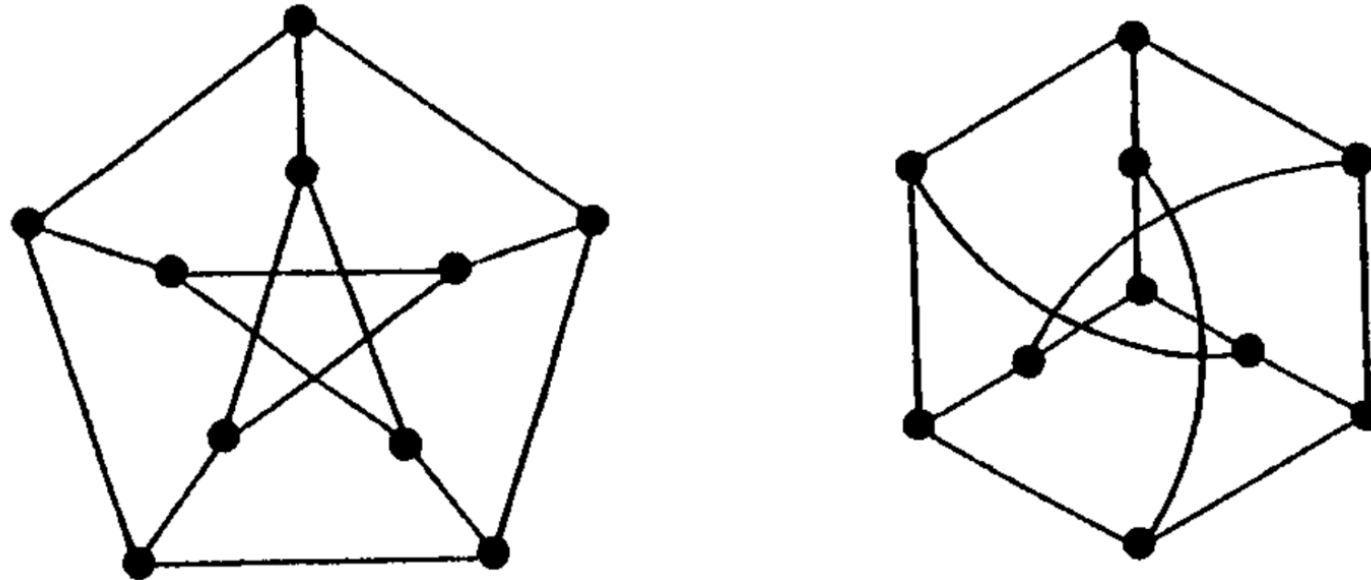
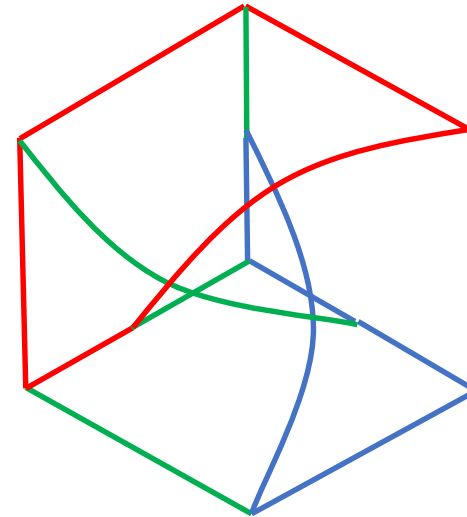
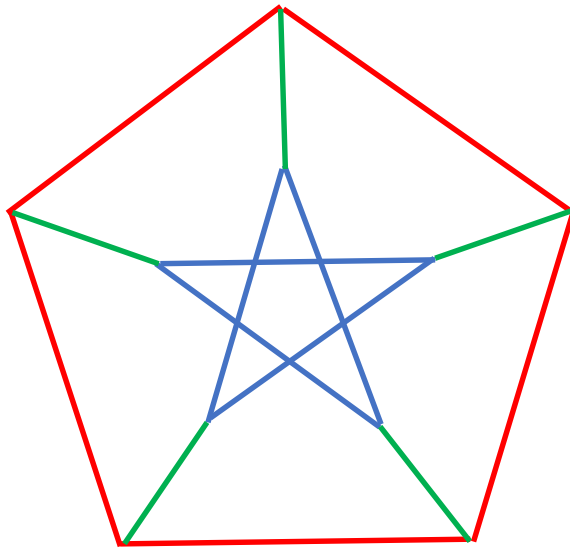


Figure 1.4

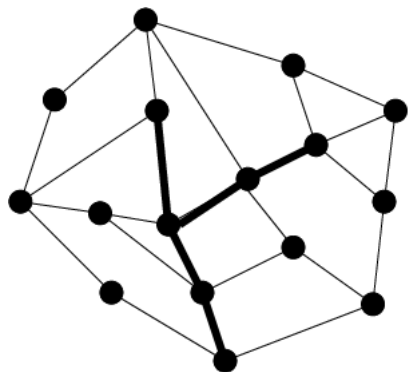
Example: Peterson graph (cont.)

- Show that the following two graphs are same/isomorphic

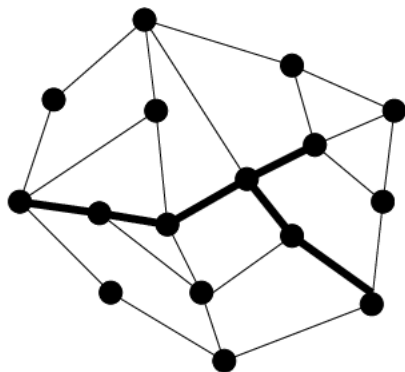


Subgraphs

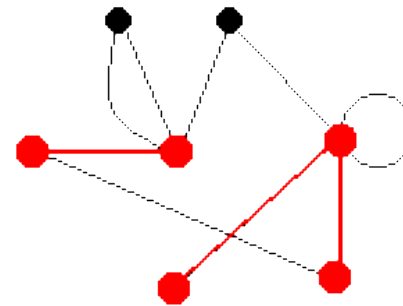
- A **subgraph** of a graph G is a graph H such that
$$V(H) \subseteq V(G), E(H) \subseteq E(G)$$
and the ends of an edge $e \in E(H)$ are the same as its ends in G
 - H is a **spanning subgraph** when $V(H) = V(G)$
 - The subgraph of G **induced** by a subset $S \subseteq V(G)$ is the subgraph whose vertex set is S and whose edges are all the edges of G with both ends in S



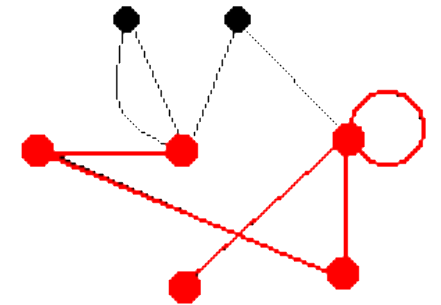
(a)



(b)



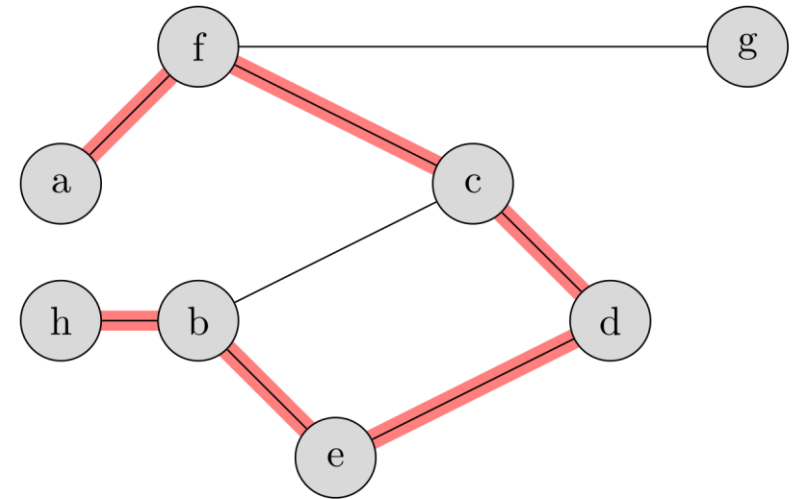
Subgraph (in red)



Induced Subgraph

Paths (路径)

- A **path** is a non-empty alternating sequence $v_0e_1v_1e_2 \dots e_kv_k$ where vertices are all **distinct**
 - Or it can be written as $v_0v_1 \dots v_k$ in simple graphs
- P^k : path of length k (the number of edges)

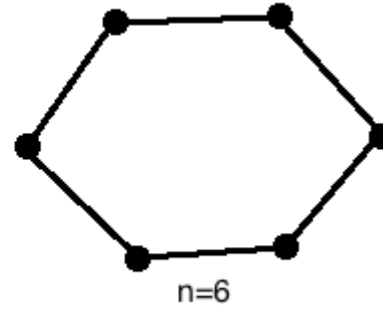
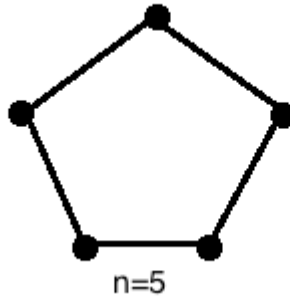
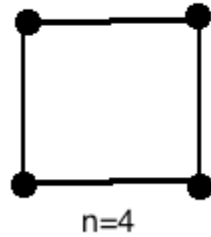


Walk (游走)

- A **walk** is a non-empty alternating sequence $v_0e_1v_1e_2 \dots e_kv_k$
 - The vertices not necessarily distinct
 - The length = the number of edges
- **Proposition** (1.2.5, W) Every u - v walk contains a u - v path

Cycles (环)

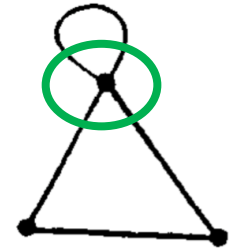
- If $P = x_0x_1 \dots x_{k-1}$ is a path and $k \geq 3$, then the graph $C := P + x_{k-1}x_0$ is called a **cycle**
- C^k : cycle of length k (the number of edges/vertices)



- **Proposition** (1.2.15, W) Every closed odd walk contains an odd cycle

Neighbors and degree

- Two vertices $a \neq b$ are called **adjacent** if they are joined by an edge
 - $N(x)$: set of all vertices adjacent to x
 - **neighbors** of x
 - A vertex is **isolated** vertex if it has no neighbors
- The number of edges incident with a vertex x is called the **degree** of x
 - A **loop** contributes **2** to the degree

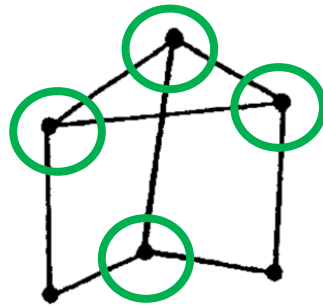


graph with loop

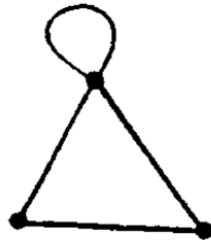
- A graph is **finite** when both $E(G)$ and $V(G)$ are finite sets

Handshaking Theorem (Euler 1736)

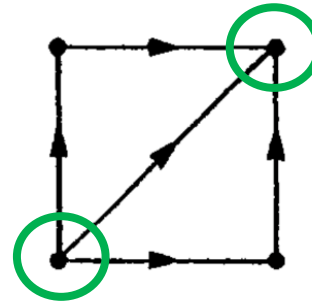
- **Theorem** A finite graph G has an even number of vertices with odd degree



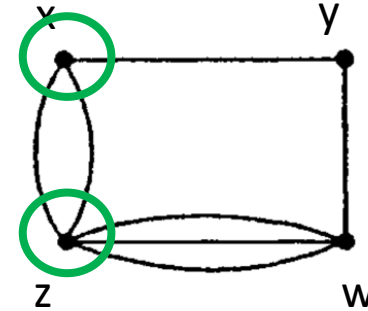
(i) graph



(ii) graph with loop



(iii) digraph



(iv) multiple edges

Figure 1.2

Proof

- **Theorem** A finite graph G has an even number of vertices with odd degree.
- **Proof** The degree of x is the number of times it appears in the right column. Thus

$$\sum_{x \in V(G)} \deg(x) = 2|E(G)|$$

edge	ends
a	x, z
b	y, w
c	x, z
d	z, w
e	z, w
f	x, y
g	z, w

Figure 1.1

Degree

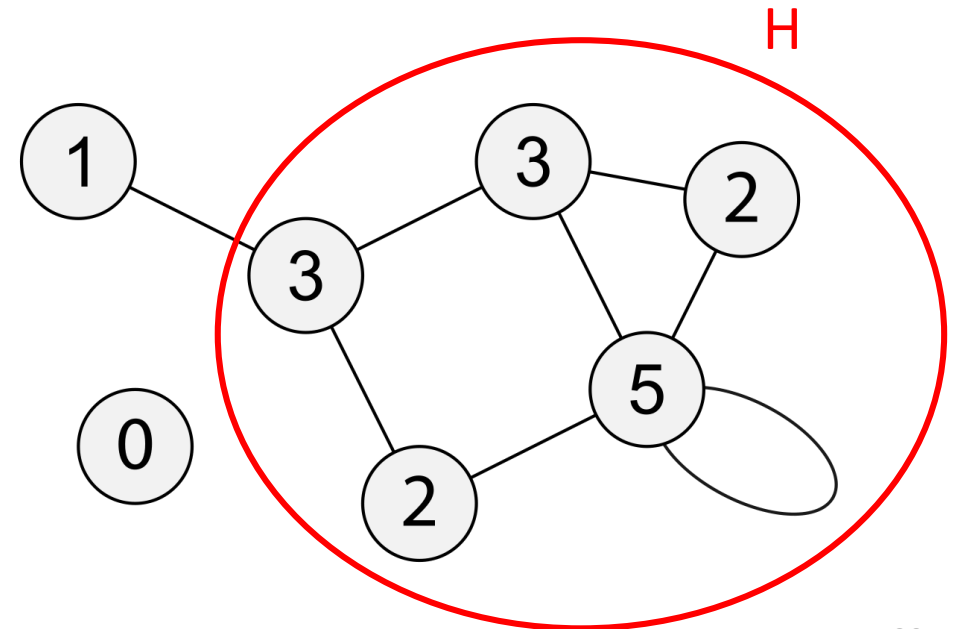
- **Minimal** degree of G : $\delta(G) = \min\{d(v) : v \in V\}$
- **Maximal** degree of G : $\Delta(G) = \max\{d(v) : v \in V\}$
- **Average** degree of G : $d(G) = \frac{1}{|V|} \sum_{v \in V} d(v) = \frac{2|E|}{|V|}$
- All measure the 'density' of a graph
- $d(G) \geq \delta(G)$

Degree (global to local)

- **Proposition** (1.2.2, D) Every graph G with at least one edge has a subgraph H with

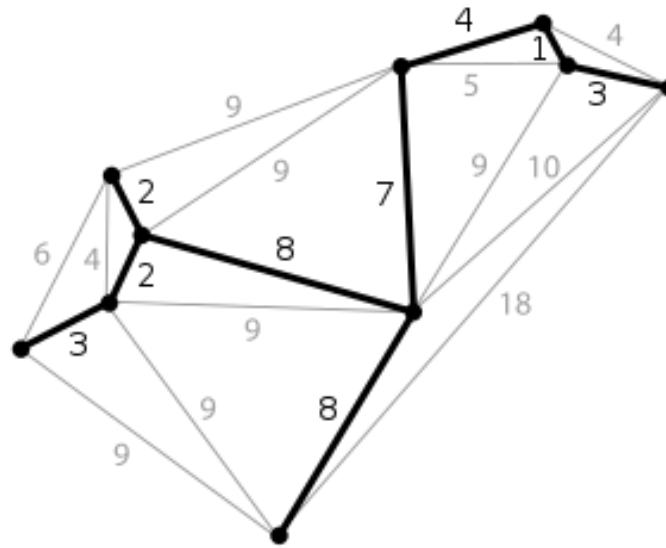
$$\delta(H) > \frac{1}{2} d(H) \geq \frac{1}{2} d(G)$$

- Example: $|G| = 7, d(G) = \frac{16}{7}$
- $\delta(H) = 2, d(H) = \frac{14}{5}$



Minimal degree guarantees long paths and cycles

- **Proposition** (1.3.1, D) Every graph G contains a path of length $\delta(G)$ and a cycle of length at least $\delta(G) + 1$, provided $\delta(G) \geq 2$.



Distance and diameter

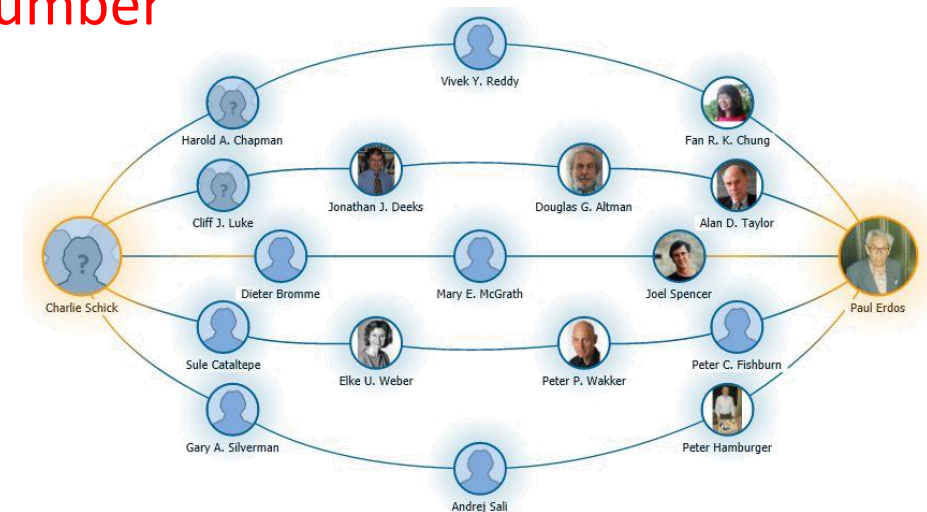
- The **distance** $d_G(x, y)$ in G of two vertices x, y is the length of a shortest $x \sim y$ path
 - if no such path exists, we set $d(x, y) := \infty$
- The greatest distance between any two vertices in G is the **diameter** of G

$$\text{diam}(G) = \max_{x, y \in V} d(x, y)$$

Example -- Erdős number



- A well-known graph
 - vertices: mathematicians of the world
 - Two vertices are adjacent if and only if they have published a joint paper
 - The distance in this graph from some mathematician to the vertex Paul Erdős is known as his or her **Erdős number**

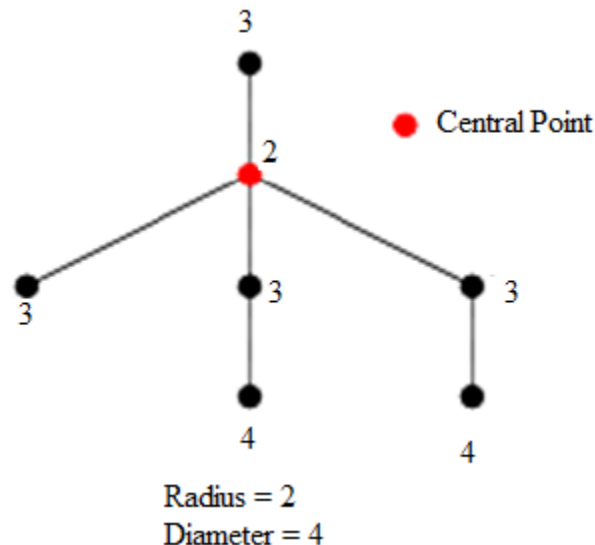


Radius and diameter

- A vertex is **central** in G if its greatest distance from other vertex is smallest, such greatest distance is the **radius** of G

$$\text{rad}(G) := \min_{x \in V} \max_{y \in V} d(x, y)$$

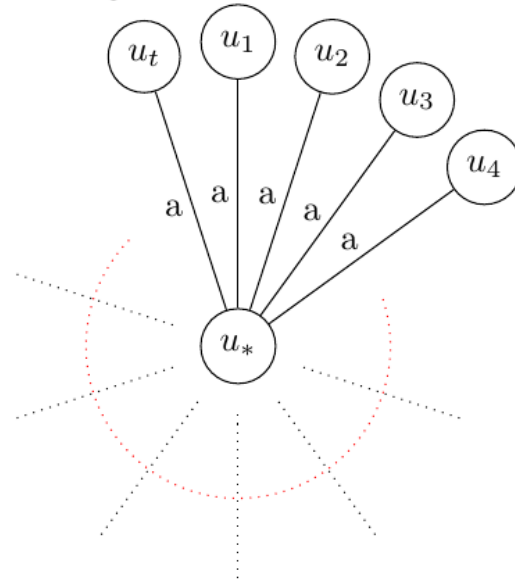
- Proposition** (1.4, H; Ex1.6, D) $\text{rad}(G) \leq \text{diam}(G) \leq 2 \text{ rad}(G)$



Radius and maximum degree control graph size

- **Proposition** (1.3.3, D) A graph G with radius at most r and maximum degree at most $\Delta \geq 3$ has fewer than $\frac{\Delta}{\Delta-2} (\Delta - 1)^r$.

Figure 1: Star Graph



Summary

Shuai Li

<https://shuaili8.github.io>

- Motivation and applications
- Basic concepts:
 - graph, isomorphism, subgraphs, paths, walks, cycles,
 - Neighbors, degree, distance, diameter, radius
- Examples:
 - Complete/regular/bipartite graphs, Peterson graph
- Theorems:
 - Handshaking
 - Large average degree guarantees dense subgraphs
 - Large minimal degree guarantees long paths and cycles
 - Radius and maximum degree control graph size

Questions?