Discrete Mathematics: Lecture 16

proposition, truth value, propositional constant/variable, negation, truth table, conjunction, disjunction, implication, bi-implication, formula

Xuming He*
Associate Professor

*School of Information Science and Technology, ShanghaiTech University

Spring Semester, 2022

Overview

- *Combinatorics: complexity analysis, etc*
- *Number theory: cryptography*
- Logic: software engineering, artificial intelligence, database theory, programming language, etc
- Graph theory: software engineering, theoretical computer science
- ...

Textbook: Discrete Mathematics and Its Applications (7th edition)
Kenneth H. Rosen, William C Brown Pub, 2011.

Logic: the study of reasoning, the basis of all mathematical reasoning.

Mathematical logic: the mathematical study of reasoning and the study of mathematical reasoning //foundation of mathematics

- Leibniz: introduced the idea of mathematical logic in "Dissertation on the Art of Combinations" in 1666
- Universal system of reasoning: reasoning based on symbols+calculations
- **Contributors:** Boole, De Morgan, Frege, Peano, Russell, Hilbert, Gödel,...
- **Areas:** (1) set theory, (2) proof theory, (3) recursion theory, (4) model theory, and their foundation (5) propositional logic and predicate logic

Our focus: propositional logic and predicate logic, (naive) set theory

Proposition

Definition: A **proposition** is a declarative sentence(that is, a sentence that declares a fact) that is either true or false.

- Lower-case letters represent propositions: $p, q, r, \ldots$
- **Truth value**: The truth value of p is true (**T**) if p is a true proposition. The truth value of p is false (**F**) if p is a false proposition.

Example:

- Washington, D.C, is the capital of the United States of America.
  (T)
- $1 + 1 = 3$
  (F)
- $(x^2)' = 2x$
  (T)

Example:

- Every even integer $n > 2$ is the sum of two primes.
    - Proposition?:
      Yes!
    - Goldbach's conjecture
    - A proposition whose truth value is not known now
- What time is it?
    - Proposition?:
      No!. It's not declarative.
- Do not smoke!
    - Proposition?:
      No!. It's not declarative.
- $x + 1 = 2$.
    - Proposition?:
      No!. It's neither true nor false.

Proposition

Simple Proposition:cannot be broken into 2 or more propositions

- $\sqrt{2}$ is irrational.

Compound Proposition: not simple

- 2 is rational and $\sqrt{2}$ is irrational.

Propositional Constant: a concrete proposition (truth value fixed)

- Every even integer $n > 2$ is the sum of two primes.

Propositional variables:a variable that represents any proposition

- Lower-case letters denote proposition variables: $p, q, r, s, \ldots$
- Truth value is not determined until it is assigned a concrete proposition

Propositional Logic: the area of logic that deals with propositions

Negation: $\neg$

Definition: Let p be any proposition.

- The **negation** of p is the statement "It is not the case that p"
- Notation: $\neg p$; read as "not p"
- **True table:**

p	$\neg p$
T	**F**
F	**T**

Negation: $\neg$

Example:

- $p =$ "Snow is black "
  - $\neg p =$ "It is not the case that snow is black."
  - $\neg p =$ "Snow is not black."
  - $\neg p \neq$ "Snow is white."

- $p =$ "Amy's smartphone has at least 32 GB of memory. "
  - $\neg p =$ "It is not the case that Amy's smartphone has at least 32 GB of memory. "
  - $\neg p =$ "Amy's smartphone does not have at least 32 GB."
  - $\neg p =$ "Amy's smartphone has less than 32 GB."

Conjunction: $\wedge$

Definition: Let p, q be any propositions.

- The **conjunction** of p and q is the statement "p and q"
- Notation: $p \wedge q$; read as "p and q"
- True table:

p	q	$p \wedge q$
T	**T**	**T**
T	**F**	**F**
F	**T**	**F**
F	**F**	**F**

Example:

- $p =$ "2<3"; $q =$ "$2^2 < 3^3$"
  - $p \wedge q =$ "$2 < 3$ and $2^2 < 3^3$."
    (T)
- $p =$ "Dog can fly"; $q =$ "Eagle can fly"
  - $p \wedge q =$ "Dog can fly and Eagle can fly."
    (F)

Disjunction: $\vee$

Definition: Let p, q be any propositions.

- The **disjunction** of p and q is the statement "p or q"
- Notation: $p \vee q$; read as "p or q"
- True table:

p	q	$p \vee q$
T	**T**	**T**
T	**F**	**T**
F	**T**	**T**
F	**F**	**F**

Example:

- $p =$ "2>3"; $q =$ "$2^2 > 3^3$"
  - $p \vee q =$ "$2 > 3$ or $2^2 > 3^3$."
    (F)
- $p =$ "Dog can fly"; $q =$ "Eagle can fly"
  - $p \vee q =$ "Dog can fly or Eagle can fly."
    (T)

Implication: $\rightarrow$

Definition: Let p, q be any propositions.

- The **conditional statement** $p \rightarrow q$ is the proposition "if p, then q."
  - p: hypothesis; q: conclusion; read as "p implies q", or "if p, then q"
- True table:

p	q	$p \rightarrow q$
T	**T**	**T**
T	**F**	**F**
F	**T**	**T**
F	**F**	**T**

Example:

- $p =$ "you get 100 on the final"; $q =$ "you will receive $A+$"
  - $p \rightarrow q =$ "If you get 100 on the final, you will receive $A+$." **(T)**
  - It is false when you get 100 on the final but don't receive $A+$, which is "when p is true but q is false."**(F)**

Bi-Implication: $\leftrightarrow$

Definition: Let p, q be any propositions.

- The **biconditional statement** $p \leftrightarrow q$ is the proposition "p if and only if q."
  - read as"p if and only if q"
- True table:

p	q	$p \leftrightarrow q$
T	**T**	**T**
T	**F**	**F**
F	**T**	**F**
F	**F**	**T**

Example:

- $p =$ "you can take the flight"; $q =$ "you buy a ticket"
  - $p \leftrightarrow q =$ "You can take the flight if and only if you buy a ticket."
  - False when $(p, q) =$(**T**, **F**) or (**F**, **T**)

Well-Formed Formulas

Definition: recursive definition of **well-formed formulas (WFFs)**

1. propositional constants (**T**, **F**) and propositional variables are WFFs
2. If A is a WFF, then $\neg A$ is a WFF.
3. If A, B are WFFs, then $(A \wedge B), (A \vee B), (A \rightarrow B), (A \leftrightarrow B)$ are WFFs
4. WFFs are results of finitely many applications of $1, 2, 3$.

Remark: well-formed formulas = propositional formulas = formulas

Use tree structure to check

- $\neg(p \wedge q) \rightarrow (r \wedge s)$
  (T)
- $(p \wedge q)\neg r$
  (F)
- $m \leftrightarrow ((p \wedge q) \rightarrow (\neg r \wedge s))$
  (T)

Summary

Proposition: a declarative sentence that is either true or false.

- simple, compound, propositional constant/variable

Logical Connectives: $\neg$ (unary), $\wedge, \vee, \rightarrow, \leftrightarrow$ (binary)

- Truth table
- Example 14 (Textbook Page 11)

Well-Formed Formulas: formulas

- propositional constant, variables
- $\neg A, (A \wedge B), (A \vee B), (A \leftarrow B), (A \leftrightarrow B)$
- Finite

Precedence of Logical Operators

Precedence (priority): $\neg, \wedge, \vee, \rightarrow, \leftrightarrow$

- formulas inside () are computed firstly
- different connectives: $\neg, \wedge, \vee, \rightarrow, \leftrightarrow$ (Decreasing Precedence)
- same connectives: from left to the right

- Example 1: $\neg p \wedge q$: $(\neg p) \wedge q$
- Example 2:$\neg(p \wedge q)$: First (.), then $\neg$.
- Example 3: $p \vee q \wedge r$: $p \vee (q \wedge r)$
- Example 4:

$$\underline{\underline{\underline{(p \rightarrow q)} \wedge \underline{(q \rightarrow r)}} \leftrightarrow \underline{(p \rightarrow r)}}$$

From Natural Language to WFFs

The Method of Translation:

- Introduce symbols to represent simple propositions
- Connect the symbols with logical connectives to obtain WFFs

Example:

- "It is not the case that snow is black."
  - p: "Snow is black"
  - Translation: $\neg p$
  - **Remark:** it is better to choose the simple proposition to be affirmative sentence.
- "π and e are both irrational"
  - p : "π is irrational"; q : "e. is irrational"
  - Translation: $p \wedge q$
- "If π is irrational, then 2π is irrational"
  - p : π is irrational; q : 2π is irrationals
  - Translation: $p \rightarrow q$

Example:

- "$e^{\pi} > \pi^{e}$ if and only if $\pi > e \ln \pi$"
  - $p : e^{\pi} > \pi^{e}$; $q : \pi > e \ln \pi$
  - Translation: $p \leftrightarrow q$
- **“ $(\sqrt{2})^{\sqrt{2}}$ is rational or irrational.”** (ambiguity in natural language)
  - $p =$ “$(\sqrt{2})^{\sqrt{2}}$ is rational ”; $q =$ “$(\sqrt{2})^{\sqrt{2}}$ is irrational"
  - Explanation 1: $(\sqrt{2})^{\sqrt{2}}$ cannot be neither rational nor irrational.
  - Emphasis: $(\sqrt{2})^{\sqrt{2}}$ is a real number, only two possibility
  - Translation 1: $p \vee q$ (by default, this is the translation of “or”)

  - Explanation 2: $(\sqrt{2})^{\sqrt{2}}$ cannot be both rational or irrational
  - It is obvious that $(\sqrt{2})^{\sqrt{2}}$ is real number. Emphasis: not both
  - Translation 2: $(p \wedge \neg q) \vee (\neg p \wedge q)$ (not both)
- The specific translations remove the ambiguity.

Discrete Mathematics: Lecture 17

translation, precedence, truth table, tautology, contradiction, contingency, satisfiable, rule of substitution, logically equivalent, rule of replacement

Xuming He
Associate Professor

School of Information Science and Technology
ShanghaiTech University

Spring Semester, 2022

Notes by Prof. Liangfeng Zhang

Review

Proposition: a declarative sentence that is either true or false.

- simple, compound, propositional constant/variable

Logical Connectives: $\neg$ (unary), $\wedge, \vee, \rightarrow, \leftrightarrow$ (binary)

- Truth table

Well-Formed Formulas (WFFs): formulas

- propositional constants (T, F) and propositional variables are WFFs
- $\neg A, (A \wedge B), (A \vee B), (A \leftarrow B), (A \leftrightarrow B)$
- Finite

Review: Proposition

Simple Proposition: cannot be broken into 2 or more propositions

- $\sqrt{2}$ is irrational.

Compound Proposition: not simple

- 2 is rational and $\sqrt{2}$ is irrational.

Propositional Constant: a concrete proposition (truth value fixed)

- Every even integer $n > 2$ is the sum of two primes.

Propositional variables: a variable that represents any proposition

- Lower-case letters denote proposition variables: $p, q, r, s, \ldots$
- Truth value is not determined until it is assigned a concrete proposition

Propositional Logic: the area of logic that deals with propositions

Review

Definition: Let p be any proposition.

- The **negation** of p is the statement "It is not the case that p"
- Notation: $\neg p$; read as "not p"
- **True table:**

p	$\neg p$
T	**F**
F	**T**

Definition: Let p, q be any propositions.

- The **conjunction** of p and q is the statement "p and q". Notation: $p \wedge q$
- The **disjunction** of p and q is the statement "p or q". Notation: $p \vee q$
- **True table:**

p	q	$p \wedge q$	$p \vee q$
T	**T**	**T**	**T**
T	**F**	**F**	**T**
F	**T**	**F**	**T**
F	**F**	**F**	**F**

Review

Definition: Let p, q be any propositions.

- The **conditional statement** $p \rightarrow q$ is the proposition "if p, then q."
  - p: hypothesis; q: conclusion; read as "p implies q", or "if p, then q"
- True table:

p	q	$p \rightarrow q$
T	**T**	**T**
T	**F**	**F**
F	**T**	**T**
F	**F**	**T**

- The **biconditional statement** $p \leftrightarrow q$ is the proposition "p if and only if q."
  - read as"p if and only if q"
- True table:

p	q	$p \leftrightarrow q$
T	**T**	**T**
T	**F**	**F**
F	**T**	**F**
F	**F**	**T**

Review: WFFs & Precedence

DEFINITION: recursive definition of **well-formed formulas** (**WFFs**)

① propositional constants (**T, F**) and propositional variables are WFFs

② If A is a WFF, then $\neg A$ is a WFF

③ If A, B are WFFs, then $(A \wedge B), (A \vee B), (A \to B), (A \leftrightarrow B)$ are WFFs

④ WFFs are results of finitely many applications of ①, ② , and ③

Precedence(优先级)**:** ¬, ∧, ∨, →, ↔

- formulas inside () are computed firstly
- different connectives: ¬, ∧, ∨, →, ↔
- same connectives: from left to the right
    - $\underline{\underline{\underline{(p \to q)} \wedge \underline{(q \to r)}} \leftrightarrow \underline{(p \to r)}}$

From Natural Language to WFFs

The Method of Translation:

- Introduce symbols p, q, r … to represent simple propositions
- Connect the symbols with logical connectives to obtain WFFs

EXAMPLE:

- "It is not the case that snow is black."
  - p:"Snow is black" Translation: $\neg p$
- "π and e are both irrational."
  - p:"π is irrational."; q:"e is irrational." Translation: $p \wedge q$
- "If π is irrational, then 2π is irrational"
  - p:"π is irrational"; q:"2π is irrational" Translation: $p \rightarrow q$
- "$e^{\pi} > \pi^{e}$ if and only if $\pi > e \ln \pi$."
  - p:"$e^{\pi} > \pi^{e}$"; q:"$\pi > e \ln \pi$" Translation: $p \leftrightarrow q$

Example

- $(\sqrt{2})^{\sqrt{2}}$ is rational or irrational. (ambiguity in natural language)
  - p: "$(\sqrt{2})^{\sqrt{2}}$ is rational"; q: "$(\sqrt{2})^{\sqrt{2}}$ is irrational"
  - **Explanation 1**: $(\sqrt{2})^{\sqrt{2}}$ cannot be neither rational nor irrational.
    - Translation 1: $p \vee q$
      - We agree that $p \vee q$ is the correct translation of "$(\sqrt{2})^{\sqrt{2}}$ is rational or irrational ."
  - **Explanation 2**: $(\sqrt{2})^{\sqrt{2}}$ cannot be both rational and irrational.
    - Translation 2: $(p \wedge \neg q) \vee (q \wedge \neg p)$
      - We agree that $(p \wedge \neg q) \vee (q \wedge \neg p)$ is the translation of "$(\sqrt{2})^{\sqrt{2}}$ is rational or irrational, but not both."

Example

- You are eligible to be President of the U.S.A. only if you are at least 35 years old, were born in the U.S.A, or at the time of your birth both of your parents were citizens, and you have lived at least 14 years in the country.
  - e: "You are eligible to be President of the U.S.A.,"
  - a: "You are at least 35 years old,"
  - b: "You were born in the U.S.A,"
  - p: "At the time of your birth, both of your parents were citizens,"
  - r: "You have lived at least 14 years in the U.S.A."
    - Translation: $e \rightarrow (a \wedge (b \vee p) \wedge r)$

Example

- A comes to the party if and only if B doesn't come, but, if B comes, then C doesn't come and D comes.
- A sufficient condition for A coming to the party is that, if B does not come, then at least one of C and D must come.
  - a: "A comes to the party."
  - b: "B comes to the party."
  - c: "C comes to the party."
  - d: "D comes to the party."
    - Translation 1: $(a \leftrightarrow \neg b) \wedge \big(b \rightarrow (\neg c \wedge d)\big)$
    - Translation 2: $(\neg b \rightarrow (c \vee d)) \rightarrow a$

Example

- **System Specifications**: Determine if there is a system that satisfy all of the following requirements.
    1. The diagnostic message is stored in the buffer or it is retransmitted.
    2. The diagnostic message is not stored in the buffer.
    3. If the diagnostic message is stored in the buffer, then it's retransmitted.
        - s: "The diagnostic message is stored in the buffer"
        - r: "The diagnostic message is retransmitted"
            - $s \vee r; \neg s; s \rightarrow r$
                - There is a system that satisfies 1, 2 and 3. ($s = \mathbf{F}, r = \mathbf{T}$)
    4. Add one more requirement "The diagnostic message is not retransmitted"
        - $s \vee r; \neg s; s \rightarrow r; \neg r$
            - There is no system that satisfies 1, 2, 3 and 4.

Truth Table

DEFINITION: Let F be a WFF of $p_1, \ldots, p_n$, n propositional variables

- A **truth assignment** (真值指派) for F is a map $\alpha: \{p_1, \ldots, p_n\} \to \{\mathbf{T}, \mathbf{F}\}$.
  - There are 2^n different truth assignments.

p_1	p_2	$\cdots$	p_n	F
T	**T**	$\cdots$	**T**	$\cdot$
T	**T**	$\cdots$	**F**	$\cdot$
$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$
F	**F**	$\cdots$	**F**	$\cdot$

EXAMPLE: Truth tables of $A = p \vee \neg p, B = p \wedge \neg p, C = p \to \neg p$

p	$\neg p$	A
T	**F**	
F	**T**	

p	$\neg p$	B
T	**F**	
F	**T**	

p	$\neg p$	C
T	**F**	
F	**T**	

Truth Table

DEFINITION: Let F be a WFF of $p_1, \dots, p_n$, n propositional variables

- A **truth assignment** (真值指派) for F is a map $\alpha: \{p_1, \dots, p_n\} \to \{\mathbf{T}, \mathbf{F}\}$.
  - There are 2^n different truth assignments.

p_1	p_2	$\cdots$	p_n	F
T	**T**	$\cdots$	**T**	$\cdot$
T	**T**	$\cdots$	**F**	$\cdot$
$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$
F	**F**	$\cdots$	**F**	$\cdot$

EXAMPLE: Truth tables of $A = p \vee \neg p, B = p \wedge \neg p, C = p \to \neg p$

p	$\neg p$	A
T	**F**	**T**
F	**T**	**T**

p	$\neg p$	B
T	**F**	**F**
F	**T**	**F**

p	$\neg p$	C
T	**F**	**F**
F	**T**	**T**

Truth Table

EXAMPLE: Truth table of $F = (p \to q) \land (q \to r) \leftrightarrow (p \to r)$

- $A = p \to q$; $B = q \to r$; $C = p \to r$
- $F = A \land B \leftrightarrow C$

p	q	r	A	B	C	$A \land B$	F
T	**T**	**T**					
T	**T**	**F**					
T	**F**	**T**					
T	**F**	**F**					
F	**T**	**T**					
F	**T**	**F**					
F	**F**	**T**					
F	**F**	**F**					

Truth Table

EXAMPLE: Truth table of $F = (p \to q) \land (q \to r) \leftrightarrow (p \to r)$

- $A = p \to q$; $B = q \to r$; $C = p \to r$
- $F = A \land B \leftrightarrow C$

p	q	r	A	B	C	$A \land B$	F
T	**T**	**T**	**T**	**T**	**T**	**T**	**T**
T	**T**	**F**	**T**	**F**	**F**	**F**	**T**
T	**F**	**T**	**F**	**T**	**T**	**F**	**F**
T	**F**	**F**	**F**	**T**	**F**	**F**	**T**
F	**T**	**T**	**T**	**T**	**T**	**T**	**T**
F	**T**	**F**	**T**	**F**	**T**	**F**	**F**
F	**F**	**T**	**T**	**T**	**T**	**T**	**T**
F	**F**	**F**	**T**	**T**	**T**	**T**	**T**

Types of WFFs

Tautology(重言式)**:** a WFF whose truth value is **T** for all truth assignment

- $p \vee \neg p$ is a tautology

Contradiction(矛盾式)**:** a WFF whose truth value is **F** for all truth assignment

- $p \wedge \neg p$ is a contradiction

Contingency(可能式)**:** neither tautology nor contradiction

- $p \rightarrow \neg p$ is a contingency

Satisfiable(可满足的):a WFF is satisfiable if it is true for at least one truth assignment

Rule of Substitution: (代入规则) Let B be a formula obtained from a tautology A by substituting a propositional variable in A with an arbitrary formula. Then B must be a tautology.

- $p \vee \neg p$ is a tautology: $(q \wedge r) \vee \neg(q \wedge r)$ is a tautology as well.

Logically Equivalent

DEFINITION: Let A and B be WFFs in propositional variables $p_1, \dots, p_n$.

- A and B are **logically equivalent** (等值) if they always have the same truth value for every truth assignment (of $p_1, \dots, p_n$)
  - Notation: $A \equiv B$

THEOREM: $A \equiv B$ if and only if $A \leftrightarrow B$ is a tautology.

- $A \equiv B$
- iff for any truth assignment, A, B take the same truth values
- iff for any truth assignment, $A \leftrightarrow B$ is true
- iff $A \leftrightarrow B$ is a tautology

THEOREM: $A \equiv A$; If $A \equiv B$, then $B \equiv A$; If $A \equiv B, B \equiv C$, then $A \equiv C$

QUESTION: How to prove $A \equiv B$?

Proving $A \equiv B$

EXAMPLE: $P \wedge (Q \vee R) \equiv (P \wedge Q) \vee (P \wedge R)$ **//distributive law**

- Idea: Show that A, B have the same truth table.

P	Q	R	$Q \vee R$	$P \wedge (Q \vee R)$	$P \wedge Q$	$P \wedge R$	$(P \wedge Q) \vee (P \wedge R)$
T	**T**	**T**					
T	**T**	**F**					
T	**F**	**T**					
T	**F**	**F**					
F	**T**	**T**					
F	**T**	**F**					
F	**F**	**T**					
F	**F**	**F**					

Proving $A \equiv B$

EXAMPLE: $P \wedge (Q \vee R) \equiv (P \wedge Q) \vee (P \wedge R)$ **//distributive law**

- Idea: Show that A, B have the same truth table.

P	Q	R	$Q \vee R$	$P \wedge (Q \vee R)$	$P \wedge Q$	$P \wedge R$	$(P \wedge Q) \vee (P \wedge R)$
T	**T**	**T**	**T**	**T**	**T**	**T**	**T**
T	**T**	**F**	**T**	**T**	**T**	**F**	**T**
T	**F**	**T**	**T**	**T**	**F**	**T**	**T**
T	**F**	**F**	**F**	**F**	**F**	**F**	**F**
F	**T**	**T**	**T**	**F**	**F**	**F**	**F**
F	**T**	**F**	**T**	**F**	**F**	**F**	**F**
F	**F**	**T**	**T**	**F**	**F**	**F**	**F**
F	**F**	**F**	**F**	**F**	**F**	**F**	**F**

REMARK: $P \vee (Q \wedge R) \equiv (P \vee Q) \wedge (P \vee R)$ can be shown similarly.

Logical Equivalences

Name	**Logical Equivalences**	**NO.**
Double Negation Law 双重否定律	$\neg(\neg P) \equiv P$	1
Identity Laws 同一律	$P \wedge \mathbf{T} \equiv P$	2
	$P \vee \mathbf{F} \equiv P$	3
Idempotent Laws 等幂律	$P \vee P \equiv P$	4
	$P \wedge P \equiv P$	5
Domination Laws 零律	$P \vee \mathbf{T} \equiv \mathbf{T}$	6
	$P \wedge \mathbf{F} \equiv \mathbf{F}$	7
Negation Laws 补余律	$P \vee \neg P \equiv \mathbf{T}$	8
	$P \wedge \neg P \equiv \mathbf{F}$	9

Logical Equivalences

Name	Logical Equivalences	NO.
Commutative Laws 交换律	$P \vee Q \equiv Q \vee P$	10
	$P \wedge Q \equiv Q \wedge P$	11
Associative Laws 结合律	$P \vee (Q \vee R) \equiv (P \vee Q) \vee R$	12
	$P \wedge (Q \wedge R) \equiv (P \wedge Q) \wedge R$	13
Distributive Laws 分配律	$P \wedge (Q \vee R) \equiv (P \wedge Q) \vee (P \wedge R)$	14
	$P \vee (Q \wedge R) \equiv (P \vee Q) \wedge (P \vee R)$	15
De Morgan's Laws 摩根律	$\neg(P \wedge Q) \equiv (\neg P) \vee (\neg Q)$	16
	$\neg(P \vee Q) \equiv (\neg P) \wedge (\neg Q)$	17
Absorption Laws 吸收律	$P \vee (P \wedge Q) \equiv P$	18
	$P \wedge (P \vee Q) \equiv P$	19

Logical Equivalences

Name	Logical Equivalences	NO.
Laws Involving Implication $\rightarrow$	$P \rightarrow Q \equiv \neg P \vee Q$	20
	$P \rightarrow Q \equiv \neg Q \rightarrow \neg P$	21
	$(P \rightarrow R) \wedge (Q \rightarrow R) \equiv (P \vee Q) \rightarrow R$	22
	$P \rightarrow (Q \rightarrow R) \equiv (P \wedge Q) \rightarrow R$	23
	$P \rightarrow (Q \rightarrow R) \equiv Q \rightarrow (P \rightarrow R)$	24
Laws Involving Bi-Implication $\leftrightarrow$	$P \leftrightarrow Q \equiv (P \rightarrow Q) \wedge (Q \rightarrow P)$	25
	$P \leftrightarrow Q \equiv (\neg P \vee Q) \wedge (P \vee \neg Q)$	26
	$P \leftrightarrow Q \equiv (P \wedge Q) \vee (\neg P \wedge \neg Q)$	27
	$P \leftrightarrow Q \equiv \neg P \leftrightarrow \neg Q$	28

Proving $A \equiv B$

<u>Rule of Replacement:</u> (替换规则) Replacing a sub-formula in a formula F with a logically equivalent sub-formula gives a formula logically equivalent to the formula F.

EXAMPLE: $P \to Q \equiv \neg Q \to \neg P$

$$P \to Q \equiv \neg P \lor Q \equiv Q \lor \neg P \equiv \neg(\neg Q) \lor \neg P \equiv \neg Q \to \neg P$$

EXAMPLE: $P \leftrightarrow Q \equiv (\neg P \lor Q) \land (P \lor \neg Q)$

$$P \leftrightarrow Q \equiv (P \to Q) \land (Q \to P) \equiv (\neg P \lor Q) \land (\neg Q \lor P)$$
$$\equiv (\neg P \lor Q) \land (P \lor \neg Q)$$

EXAMPLE: $P \to (Q \to R) \equiv (P \land Q) \to R$

$$P \to (Q \to R) \equiv \neg P \lor (\neg Q \lor R) \equiv (\neg P \lor \neg Q) \lor R \equiv \neg(P \land Q) \lor R$$
$$\equiv (P \land Q) \to R$$

Discrete Mathematics: Lecture 18

logically equivalent, rule of replacement, tautological implications

Xuming He
Associate Professor

School of Information Science and Technology
ShanghaiTech University

Spring Semester, 2022

Notes by Prof. Liangfeng Zhang

Logically Equivalent

DEFINITION: Let A and B be WFFs in propositional variables $p_1, \ldots, p_n$.

- A and B are **logically equivalent** (等值) if they always have the same truth value for every truth assignment (of $p_1, \ldots, p_n$)
  - Notation: $A \equiv B$

THEOREM: $A \equiv B$ if and only if $A \leftrightarrow B$ is a tautology.

- $A \equiv B$
- iff for any truth assignment, A, B take the same truth values
- iff for any truth assignment, $A \leftrightarrow B$ is true
- iff $A \leftrightarrow B$ is a tautology

THEOREM: $A \equiv A$; If $A \equiv B$, then $B \equiv A$; If $A \equiv B, B \equiv C$, then $A \equiv C$

QUESTION: How to prove $A \equiv B$?

Proving $A \equiv B$

EXAMPLE: $P \wedge (Q \vee R) \equiv (P \wedge Q) \vee (P \wedge R)$ **//distributive law**

- Idea: Show that A, B have the same truth table.

P	Q	R	$Q \vee R$	$P \wedge (Q \vee R)$	$P \wedge Q$	$P \wedge R$	$(P \wedge Q) \vee (P \wedge R)$
T	**T**	**T**	**T**	**T**	**T**	**T**	**T**
T	**T**	**F**	**T**	**T**	**T**	**F**	**T**
T	**F**	**T**	**T**	**T**	**F**	**T**	**T**
T	**F**	**F**	**F**	**F**	**F**	**F**	**F**
F	**T**	**T**	**T**	**F**	**F**	**F**	**F**
F	**T**	**F**	**T**	**F**	**F**	**F**	**F**
F	**F**	**T**	**T**	**F**	**F**	**F**	**F**
F	**F**	**F**	**F**	**F**	**F**	**F**	**F**

REMARK: $P \vee (Q \wedge R) \equiv (P \vee Q) \wedge (P \vee R)$ can be shown similarly.

Logical Equivalences

Name	Logical Equivalences	NO.
Double Negation Law 双重否定律	$\neg(\neg P) \equiv P$	1
Identity Laws 同一律	$P \wedge \mathbf{T} \equiv P$	2
	$P \vee \mathbf{F} \equiv P$	3
Idempotent Laws 等幂律	$P \vee P \equiv P$	4
	$P \wedge P \equiv P$	5
Domination Laws 零律	$P \vee \mathbf{T} \equiv \mathbf{T}$	6
	$P \wedge \mathbf{F} \equiv \mathbf{F}$	7
Negation Laws 补余律	$P \vee \neg P \equiv \mathbf{T}$	8
	$P \wedge \neg P \equiv \mathbf{F}$	9

Logical Equivalences

Name	Logical Equivalences	NO.
Commutative Laws 交换律	$P \vee Q \equiv Q \vee P$	10
	$P \wedge Q \equiv Q \wedge P$	11
Associative Laws 结合律	$P \vee (Q \vee R) \equiv (P \vee Q) \vee R$	12
	$P \wedge (Q \wedge R) \equiv (P \wedge Q) \wedge R$	13
Distributive Laws 分配律	$P \wedge (Q \vee R) \equiv (P \wedge Q) \vee (P \wedge R)$	14
	$P \vee (Q \wedge R) \equiv (P \vee Q) \wedge (P \vee R)$	15
De Morgan's Laws 摩根律	$\neg(P \wedge Q) \equiv (\neg P) \vee (\neg Q)$	16
	$\neg(P \vee Q) \equiv (\neg P) \wedge (\neg Q)$	17
Absorption Laws 吸收律	$P \vee (P \wedge Q) \equiv P$	18
	$P \wedge (P \vee Q) \equiv P$	19

Logical Equivalences

Name	Logical Equivalences	NO.
Laws Involving Implication $\rightarrow$	$P \rightarrow Q \equiv \neg P \vee Q$	20
	$P \rightarrow Q \equiv \neg Q \rightarrow \neg P$	21
	$(P \rightarrow R) \wedge (Q \rightarrow R) \equiv (P \vee Q) \rightarrow R$	22
	$P \rightarrow (Q \rightarrow R) \equiv (P \wedge Q) \rightarrow R$	23
	$P \rightarrow (Q \rightarrow R) \equiv Q \rightarrow (P \rightarrow R)$	24
Laws Involving Bi-Implication $\leftrightarrow$	$P \leftrightarrow Q \equiv (P \rightarrow Q) \wedge (Q \rightarrow P)$	25
	$P \leftrightarrow Q \equiv (\neg P \vee Q) \wedge (P \vee \neg Q)$	26
	$P \leftrightarrow Q \equiv (P \wedge Q) \vee (\neg P \wedge \neg Q)$	27
	$P \leftrightarrow Q \equiv \neg P \leftrightarrow \neg Q$	28

Proving $A \equiv B$

<u>Rule of Replacement:</u> (替换规则) Replacing a sub-formula in a formula F with a logically equivalent sub-formula gives a formula logically equivalent to the formula F.

EXAMPLE: $P \rightarrow Q \equiv \neg Q \rightarrow \neg P$

$$P \rightarrow Q \equiv \neg P \vee Q \equiv Q \vee \neg P \equiv \neg(\neg Q) \vee \neg P \equiv \neg Q \rightarrow \neg P$$

EXAMPLE: $P \leftrightarrow Q \equiv (\neg P \vee Q) \wedge (P \vee \neg Q)$

$$P \leftrightarrow Q \equiv (P \rightarrow Q) \wedge (Q \rightarrow P) \equiv (\neg P \vee Q) \wedge (\neg Q \vee P)$$
$$\equiv (\neg P \vee Q) \wedge (P \vee \neg Q)$$

EXAMPLE: $P \rightarrow (Q \rightarrow R) \equiv (P \wedge Q) \rightarrow R$

$$P \rightarrow (Q \rightarrow R) \equiv \neg P \vee (\neg Q \vee R) \equiv (\neg P \vee \neg Q) \vee R \equiv \neg(P \wedge Q) \vee R$$
$$\equiv (P \wedge Q) \rightarrow R$$

Logically Equivalent

THEOREM: Let $A^{-1}(\mathbf{T})$ be the set of truth assignments such that A is true. Then $A \equiv B$ if and only if $A^{-1}(\mathbf{T}) = B^{-1}(\mathbf{T})$.

- $A \equiv B$ if and only if $A^{-1}(\mathbf{F}) = B^{-1}(\mathbf{F})$

Proving $A \equiv B$

EXAMPLE: $P \wedge Q \equiv Q \wedge P$ **//commutative law**

- Idea: Show that $A^{-1}(\mathbf{T}) = B^{-1}(\mathbf{T})$.
- $A = P \wedge Q; B = Q \wedge P$
  - $A = \mathbf{T}$ if and only if $(P, Q) = (\mathbf{T}, \mathbf{T})$
    - $A^{-1}(\mathbf{T}) = \{(\mathbf{T}, \mathbf{T})\}$
  - $B = \mathbf{T}$ if and only if $(Q, P) = (\mathbf{T}, \mathbf{T})$
    - $B^{-1}(\mathbf{T}) = \{(\mathbf{T}, \mathbf{T})\}$
- $A^{-1}(\mathbf{T}) = B^{-1}(\mathbf{T})$
- $A \equiv B$

REMARK: $P \wedge (Q \wedge R) \equiv (P \wedge Q) \wedge R$ can be shown similarly.

- **Associative law**

Proving $A \equiv B$

EXAMPLE: $P \vee Q \equiv Q \vee P$ **//commutative law**

- Idea: Show that $A^{-1}(\mathbf{F}) = B^{-1}(\mathbf{F})$.
- $A = P \vee Q$; $B = Q \vee P$
  - $A = \mathbf{F}$ if and only if $(P, Q) = (\mathbf{F}, \mathbf{F})$
    - $A^{-1}(\mathbf{F}) = \{(\mathbf{F}, \mathbf{F})\}$
  - $B = \mathbf{F}$ if and only if $(Q, P) = (\mathbf{F}, \mathbf{F})$
    - $B^{-1}(\mathbf{F}) = \{(\mathbf{F}, \mathbf{F})\}$
- $A^{-1}(\mathbf{F}) = B^{-1}(\mathbf{F})$
- $A \equiv B$

REMARK: $P \vee (Q \vee R) \equiv (P \vee Q) \vee R$ can be shown similarly.

- **Associative law**

Tautological Implications

DEFINITION: Let A and B be WFFs in propositional variables $p_1, \ldots, p_n$.

- A **tautologically implies** (重言蕴涵) B if every truth assignment that causes A to be true causes B to be true.
  - Notation: $A \Rightarrow B$, called a **tautological implication**
  - $A^{-1}(\mathbf{T}) \subseteq B^{-1}(\mathbf{T})$; $B^{-1}(\mathbf{F}) \subseteq A^{-1}(\mathbf{F})$

THEOREM: $A \Rightarrow B$ iff $A \rightarrow B$ is a tautology.

- $A \Rightarrow B$ iff $A^{-1}(\mathbf{T}) \subseteq B^{-1}(\mathbf{T})$ iff $A \rightarrow B$ is a tautology

THEOREM: $A \Rightarrow B$ iff $A \wedge \neg B$ is a contradiction.

- $A \rightarrow B \equiv \neg A \vee B \equiv \neg(A \wedge \neg B)$

Proving $\boldsymbol{A \Rightarrow B}$: (1) $A^{-1}(\mathbf{T}) \subseteq B^{-1}(\mathbf{T})$; (2) $B^{-1}(\mathbf{F}) \subseteq A^{-1}(\mathbf{F})$;

(3) $A \rightarrow B$ is a tautology; (4) $A \wedge \neg B$ is a contradiction

Proving $A \Rightarrow B$

EXAMPLE: Show the tautological implication "$p \wedge (p \to q) \Rightarrow q$".

- Let $A = p \wedge (p \to q)$; $B = q$. Need to show that "$A \Rightarrow B$"
- $A^{-1}(\mathbf{T}) = \{(\mathbf{T}, \mathbf{T})\}$; $B^{-1}(\mathbf{T}) = \{(\mathbf{T}, \mathbf{T}), (\mathbf{F}, \mathbf{T})\}$: $A^{-1}(\mathbf{T}) \subseteq B^{-1}(\mathbf{T})$.

p	q	$p \to q$	A	B
T	**T**	**T**	**<u>T</u>**	**<u>T</u>**
T	**F**	**F**	**F**	**F**
F	**T**	**T**	**F**	**T**
F	**F**	**T**	**F**	**F**

- $$\begin{aligned} A \to B &\equiv \neg\big(p \wedge (p \to q)\big) \vee q \\ &\equiv \big(\neg p \vee \neg(p \to q)\big) \vee q \\ &\equiv (\neg p \vee q) \vee \neg(p \to q) \\ &\equiv (p \to q) \vee \neg(p \to q) \\ &\equiv \mathbf{T} \end{aligned}$$
- $$\begin{aligned} A \wedge \neg B &\equiv \big(p \wedge (p \to q)\big) \wedge \neg q \\ &\equiv (\neg q \wedge p) \wedge (p \to q) \\ &\equiv \neg(p \to q) \wedge (p \to q) \\ &\equiv \mathbf{F} \end{aligned}$$

Tautological Implications

Name	**Tautological Implication**	**NO.**
Conjunction(合取)	$(P) \wedge (Q) \Rightarrow P \wedge Q$	1
Simplification(化简)	$P \wedge Q \Rightarrow P$	2
Addition(附加)	$P \Rightarrow P \vee Q$	3
Modus ponens(假言推理)	$P \wedge (P \rightarrow Q) \Rightarrow Q$	4
Modus tollens(拒取)	$\neg Q \wedge (P \rightarrow Q) \Rightarrow \neg P$	5
Disjunctive syllogism(析取三段论)	$\neg P \wedge (P \vee Q) \Rightarrow Q$	6
Hypothetical syllogism(假言三段论)	$(P \rightarrow Q) \wedge (Q \rightarrow R) \Rightarrow (P \rightarrow R)$	7
Resolution (归结)	$(P \vee Q) \wedge (\neg P \vee R) \Rightarrow Q \vee R$	8

Proofs for 5 and 6

EXAMPLE: $\neg Q \wedge (P \rightarrow Q) \Rightarrow \neg P$

- $A = \neg Q \wedge (P \rightarrow Q), B = \neg P.$
- $A \rightarrow B \equiv \neg\big(\neg Q \wedge (P \rightarrow Q)\big) \vee \neg P$

$$\equiv \big(Q \vee \neg(P \rightarrow Q)\big) \vee \neg P$$

$$\equiv (\neg P \vee Q) \vee \neg(P \rightarrow Q)$$

$$\equiv \mathbf{T}$$

EXAMPLE: $\neg P \wedge (P \vee Q) \Rightarrow Q$

- $A = \neg P \wedge (P \vee Q), B = Q.$
- $A \rightarrow B \equiv \neg\big(\neg P \wedge (P \vee Q)\big) \vee Q$

$$\equiv \big(P \vee \neg(P \vee Q)\big) \vee Q$$

$$\equiv (\neg(P \vee Q) \vee P) \vee Q$$

$$\equiv \neg(P \vee Q) \vee (P \vee Q)$$

$$\equiv \mathbf{T}$$

Proofs for 7 and 8

EXAMPLE: $(P \to Q) \land (Q \to R) \Rightarrow P \to R$

- $A = (P \to Q) \land (Q \to R);\ B = (P \to R).$
- $A \land \neg B \equiv (\neg P \lor Q) \land (\neg Q \lor R) \land (P \land \neg R)$

$$\equiv \big((\neg P \lor Q) \land P\big) \land \big((\neg Q \lor R) \land \neg R\big)$$

$$\equiv \big((\neg P \land P) \lor (Q \land P)\big) \land \big((\neg Q \land \neg R) \lor (R \land \neg R)\big)$$

$$\equiv (Q \land P) \land (\neg Q \land \neg R)$$

$$\equiv \mathbf{F}$$

EXAMPLE: $(P \lor Q) \land (\neg P \lor R) \Rightarrow Q \lor R$

- $A = (P \lor Q) \land (\neg P \lor R); B = (Q \lor R).$
- $A \land \neg B \equiv (P \lor Q) \land (\neg P \lor R) \land (\neg Q \land \neg R)$

$$\equiv \big((P \lor Q) \land \neg Q\big) \land ((\neg P \lor R) \land \neg R)$$

$$\equiv (P \land \neg Q) \land (\neg P \land \neg R)$$

$$\equiv \mathbf{F}$$

More Examples

EXAMPLE: $(P \leftrightarrow Q) \wedge (Q \leftrightarrow R) \Rightarrow (P \leftrightarrow R)$

- $A = (P \leftrightarrow Q) \wedge (Q \leftrightarrow R); B = (P \leftrightarrow R).$
- $A = \mathbf{T}$ iff $(P \leftrightarrow Q) = \mathbf{T}$ and $(Q \leftrightarrow R) = \mathbf{T}$ iff $P = Q$ and $Q = R$
  - $A^{-1}(\mathbf{T}) = \{(\mathbf{T}, \mathbf{T}, \mathbf{T}), (\mathbf{F}, \mathbf{F}, \mathbf{F})\}$
- $B = \mathbf{T}$ iff $P = R$
  - $B^{-1}(\mathbf{T}) = \{(\mathbf{T}, \mathbf{T}, \mathbf{T}), (\mathbf{T}, \mathbf{F}, \mathbf{T}), (\mathbf{F}, \mathbf{T}, \mathbf{F}), (\mathbf{F}, \mathbf{F}, \mathbf{F})\}$
- $A^{-1}(\mathbf{T}) \subseteq B^{-1}(\mathbf{T})$

EXAMPLE: $(Q \rightarrow R) \Rightarrow ((P \vee Q) \rightarrow (P \vee R))$

- $A = Q \rightarrow R; B = ((P \vee Q) \rightarrow (P \vee R)).$
- $A = \mathbf{F}$ iff $(Q, R) = (\mathbf{T}, \mathbf{F})$
  - $A^{-1}(\mathbf{F}) = \{(\mathbf{T}, \mathbf{T}, \mathbf{F}), (\mathbf{F}, \mathbf{T}, \mathbf{F})\}$
- $B = \mathbf{F}$ iff $(P \vee Q, P \vee R) = (\mathbf{T}, \mathbf{F})$ iff $(P, Q) \neq (\mathbf{F}, \mathbf{F})$ and $(P, R) = (\mathbf{F}, \mathbf{F})$
  - $B^{-1}(\mathbf{F}) = \{(\mathbf{F}, \mathbf{T}, \mathbf{F})\}$
- $A^{-1}(\mathbf{F}) \supseteq B^{-1}(\mathbf{F})$

More Examples

EXAMPLE: $(P \rightarrow R) \wedge (Q \rightarrow S) \wedge (P \vee Q) \Rightarrow R \vee S$

- $A = (P \rightarrow R) \wedge (Q \rightarrow S) \wedge (P \vee Q); B = R \vee S$
- $A \wedge \neg B \equiv (P \rightarrow R) \wedge (Q \rightarrow S) \wedge (P \vee Q) \wedge \neg(R \vee S)$

$$\begin{aligned}
&\equiv (\neg P \vee R) \wedge (\neg Q \vee S) \wedge (P \vee Q) \wedge (\neg R \wedge \neg S) \\
&\equiv ((\neg P \vee R) \wedge \neg R)) \wedge ((\neg Q \vee S) \wedge \neg S) \wedge (P \vee Q) \\
&\equiv ((\neg P \wedge \neg R) \vee (R \wedge \neg R)) \wedge ((\neg Q \wedge \neg S) \vee (S \wedge \neg S)) \wedge (P \vee Q) \\
&\equiv ((\neg P \wedge \neg R) \vee \mathbf{F}) \wedge ((\neg Q \wedge \neg S) \vee \mathbf{F}) \wedge (P \vee Q) \\
&\equiv (\neg P \wedge \neg R) \wedge (\neg Q \wedge \neg S) \wedge (P \vee Q) \\
&\equiv \neg R \wedge (\neg Q \wedge \neg S) \wedge (\neg P \wedge (P \vee Q)) \\
&\equiv \neg R \wedge (\neg Q \wedge \neg S) \wedge ((\neg P \wedge P) \vee (\neg P \wedge Q)) \\
&\equiv \neg R \wedge (\neg Q \wedge \neg S) \wedge (\mathbf{F} \vee (\neg P \wedge Q)) \\
&\equiv \neg R \wedge (\neg Q \wedge \neg S) \wedge (\neg P \wedge Q) \\
&\equiv \neg R \wedge \neg S \wedge \neg P \wedge (\neg Q \wedge Q) \\
&\equiv \neg R \wedge \neg S \wedge \neg P \wedge \mathbf{F} \\
&\equiv \mathbf{F}
\end{aligned}$$

Discrete Mathematics: Lecture 19

tautological implications, argument

Xuming He
Associate Professor

School of Information Science and Technology
ShanghaiTech University

Spring Semester, 2022

Notes by Prof. Liangfeng Zhang

Review: Types of WFFs

Tautology(重言式)**:** a WFF whose truth value is **T** for all truth assignment

- $p \vee \neg p$ is a tautology

Contradiction(矛盾式)**:** a WFF whose truth value is **F** for all truth assignment

- $p \wedge \neg p$ is a contradiction

Contingency(可能式)**:** neither tautology nor contradiction

- $p \rightarrow \neg p$ is a contingency

Satisfiable(可满足的):a WFF is satisfiable if it is true for at least one truth assignment

Rule of Substitution: (代入规则) Let B be a formula obtained from a tautology A by substituting a propositional variable in A with an arbitrary formula. Then B must be a tautology.

- $p \vee \neg p$ is a tautology: $(q \wedge r) \vee \neg(q \wedge r)$ is a tautology as well.

Review: Proving $A \equiv B$

Rule of Replacement: **(替换规则)** Replacing a sub-formula in a formula F with a logically equivalent sub-formula gives a formula logically equivalent to the formula F.

EXAMPLE: $P \to Q \equiv \neg Q \to \neg P$

$$P \to Q \equiv \neg P \vee Q \equiv Q \vee \neg P \equiv \neg(\neg Q) \vee \neg P \equiv \neg Q \to \neg P$$

EXAMPLE: $P \leftrightarrow Q \equiv (\neg P \vee Q) \wedge (P \vee \neg Q)$

$$P \leftrightarrow Q \equiv (P \to Q) \wedge (Q \to P) \equiv (\neg P \vee Q) \wedge (\neg Q \vee P)$$
$$\equiv (\neg P \vee Q) \wedge (P \vee \neg Q)$$

EXAMPLE: $P \to (Q \to R) \equiv (P \wedge Q) \to R$

$$P \to (Q \to R) \equiv \neg P \vee (\neg Q \vee R) \equiv (\neg P \vee \neg Q) \vee R \equiv \neg(P \wedge Q) \vee R$$
$$\equiv (P \wedge Q) \to R$$

Tautological Implications

DEFINITION: Let A and B be WFFs in propositional variables $p_1, \dots, p_n$.

- A **tautologically implies** (重言蕴涵) B if every truth assignment that causes A to be true causes B to be true.
  - Notation: $A \Rightarrow B$, called a **tautological implication**
  - $A^{-1}(\mathbf{T}) \subseteq B^{-1}(\mathbf{T})$; $B^{-1}(\mathbf{F}) \subseteq A^{-1}(\mathbf{F})$

THEOREM: $A \Rightarrow B$ iff $A \rightarrow B$ is a tautology.

- $A \Rightarrow B$ iff $A^{-1}(\mathbf{T}) \subseteq B^{-1}(\mathbf{T})$ iff $A \rightarrow B$ is a tautology

THEOREM: $A \Rightarrow B$ iff $A \land \neg B$ is a contradiction.

- $A \rightarrow B \equiv \neg A \lor B \equiv \neg(A \land \neg B)$

Proving $\boldsymbol{A} \Rightarrow \boldsymbol{B}$: (1) $A^{-1}(\mathbf{T}) \subseteq B^{-1}(\mathbf{T})$; (2) $B^{-1}(\mathbf{F}) \subseteq A^{-1}(\mathbf{F})$;

(3) $A \rightarrow B$ is a tautology; (4) $A \land \neg B$ is a contradiction

Proving $A \Rightarrow B$

EXAMPLE: Show the tautological implication "$p \land (p \to q) \Rightarrow q$".

- Let $A = p \land (p \to q)$; $B = q$. Need to show that "$A \Rightarrow B$"
- $A^{-1}(\mathbf{T}) = \{(\mathbf{T}, \mathbf{T})\}$; $B^{-1}(\mathbf{T}) = \{(\mathbf{T}, \mathbf{T}), (\mathbf{F}, \mathbf{T})\}$: $A^{-1}(\mathbf{T}) \subseteq B^{-1}(\mathbf{T})$.

p	q	$p \to q$	A	B
T	**T**	**T**	<u>**T**</u>	<u>**T**</u>
T	**F**	**F**	**F**	**F**
F	**T**	**T**	**F**	**T**
F	**F**	**T**	**F**	**F**

- $$\begin{aligned} A \to B &\equiv \neg\big(p \land (p \to q)\big) \lor q \\ &\equiv \big(\neg p \lor \neg(p \to q)\big) \lor q \\ &\equiv (\neg p \lor q) \lor \neg(p \to q) \\ &\equiv (p \to q) \lor \neg(p \to q) \\ &\equiv \mathbf{T} \end{aligned}$$
- $$\begin{aligned} A \land \neg B &\equiv \big(p \land (p \to q)\big) \land \neg q \\ &\equiv (\neg q \land p) \land (p \to q) \\ &\equiv \neg(p \to q) \land (p \to q) \\ &\equiv \mathbf{F} \end{aligned}$$

Tautological Implications

Name	Tautological Implication	NO.
Conjunction(合取)	$(P) \wedge (Q) \Rightarrow P \wedge Q$	1
Simplification(化简)	$P \wedge Q \Rightarrow P$	2
Addition(附加)	$P \Rightarrow P \vee Q$	3
Modus ponens(假言推理)	$P \wedge (P \rightarrow Q) \Rightarrow Q$	4
Modus tollens(拒取)	$\neg Q \wedge (P \rightarrow Q) \Rightarrow \neg P$	5
Disjunctive syllogism(析取三段论)	$\neg P \wedge (P \vee Q) \Rightarrow Q$	6
Hypothetical syllogism(假言三段论)	$(P \rightarrow Q) \wedge (Q \rightarrow R) \Rightarrow (P \rightarrow R)$	7
Resolution (归结)	$(P \vee Q) \wedge (\neg P \vee R) \Rightarrow Q \vee R$	8

Proofs for 5 and 6

EXAMPLE: $\neg Q \land (P \to Q) \Rightarrow \neg P$

- $A = \neg Q \land (P \to Q), B = \neg P.$
- $A \to B \equiv \neg\big(\neg Q \land (P \to Q)\big) \lor \neg P$

$$\equiv \big(Q \lor \neg(P \to Q)\big) \lor \neg P$$

$$\equiv (\neg P \lor Q) \lor \neg(P \to Q)$$

$$\equiv \mathbf{T}$$

EXAMPLE: $\neg P \land (P \lor Q) \Rightarrow Q$

- $A = \neg P \land (P \lor Q), B = Q.$
- $A \to B \equiv \neg\big(\neg P \land (P \lor Q)\big) \lor Q$

$$\equiv \big(P \lor \neg(P \lor Q)\big) \lor Q$$

$$\equiv (\neg(P \lor Q) \lor P) \lor Q$$

$$\equiv \neg(P \lor Q) \lor (P \lor Q)$$

$$\equiv \mathbf{T}$$

Proofs for 7 and 8

EXAMPLE: $(P \to Q) \wedge (Q \to R) \Rightarrow P \to R$

- $A = (P \to Q) \wedge (Q \to R);\ B = (P \to R).$
- $A \wedge \neg B \equiv (\neg P \vee Q) \wedge (\neg Q \vee R) \wedge (P \wedge \neg R)$

$$\equiv \big((\neg P \vee Q) \wedge P\big) \wedge \big((\neg Q \vee R) \wedge \neg R\big)$$

$$\equiv \big((\neg P \wedge P) \vee (Q \wedge P)\big) \wedge \big((\neg Q \wedge \neg\, R) \vee (R \wedge \neg R)\big)$$

$$\equiv (Q \wedge P) \wedge (\neg Q \wedge \neg R)$$

$$\equiv \mathbf{F}$$

EXAMPLE: $(P \vee Q) \wedge (\neg P \vee R) \Rightarrow Q \vee R$

- $A = (P \vee Q) \wedge (\neg P \vee R); B = (Q \vee R).$
- $A \wedge \neg B \equiv (P \vee Q) \wedge (\neg P \vee R) \wedge (\neg Q \wedge \neg R)$

$$\equiv \big((P \vee Q) \wedge \neg Q\big) \wedge ((\neg P \vee R) \wedge \neg R)$$

$$\equiv (P \wedge \neg Q) \wedge (\neg P \wedge \neg R)$$

$$\equiv \mathbf{F}$$

More Examples

EXAMPLE: $(P \leftrightarrow Q) \wedge (Q \leftrightarrow R) \Rightarrow (P \leftrightarrow R)$

- $A = (P \leftrightarrow Q) \wedge (Q \leftrightarrow R); B = (P \leftrightarrow R).$
- $A = \mathbf{T}$ iff $(P \leftrightarrow Q) = \mathbf{T}$ and $(Q \leftrightarrow R) = \mathbf{T}$ iff $P = Q$ and $Q = R$
  - $A^{-1}(\mathbf{T}) = \{(\mathbf{T}, \mathbf{T}, \mathbf{T}), (\mathbf{F}, \mathbf{F}, \mathbf{F})\}$
- $B = \mathbf{T}$ iff $P = R$
  - $B^{-1}(\mathbf{T}) = \{(\mathbf{T}, \mathbf{T}, \mathbf{T}), (\mathbf{T}, \mathbf{F}, \mathbf{T}), (\mathbf{F}, \mathbf{T}, \mathbf{F}), (\mathbf{F}, \mathbf{F}, \mathbf{F})\}$
- $A^{-1}(\mathbf{T}) \subseteq B^{-1}(\mathbf{T})$

EXAMPLE: $(Q \rightarrow R) \Rightarrow ((P \vee Q) \rightarrow (P \vee R))$

- $A = Q \rightarrow R; B = ((P \vee Q) \rightarrow (P \vee R)).$
- $A = \mathbf{F}$ iff $(Q, R) = (\mathbf{T}, \mathbf{F})$
  - $A^{-1}(\mathbf{F}) = \{(\mathbf{T}, \mathbf{T}, \mathbf{F}), (\mathbf{F}, \mathbf{T}, \mathbf{F})\}$
- $B = \mathbf{F}$ iff $(P \vee Q, P \vee R) = (\mathbf{T}, \mathbf{F})$ iff $(P, Q) \neq (\mathbf{F}, \mathbf{F})$ and $(P, R) = (\mathbf{F}, \mathbf{F})$
  - $B^{-1}(\mathbf{F}) = \{(\mathbf{F}, \mathbf{T}, \mathbf{F})\}$
- $A^{-1}(\mathbf{F}) \supseteq B^{-1}(\mathbf{F})$

More Examples

EXAMPLE: $(P \to R) \land (Q \to S) \land (P \lor Q) \Rightarrow R \lor S$

- $A = (P \to R) \land (Q \to S) \land (P \lor Q); B = R \lor S$
- $A \land \neg B \equiv (P \to R) \land (Q \to S) \land (P \lor Q) \land \neg(R \lor S)$

$$
\begin{aligned}
&\equiv (\neg P \lor R) \land (\neg Q \lor S) \land (P \lor Q) \land (\neg R \land \neg S) \\
&\equiv ((\neg P \lor R) \land \neg R)) \land ((\neg Q \lor S) \land \neg S) \land (P \lor Q) \\
&\equiv ((\neg P \land \neg R) \lor (R \land \neg R)) \land ((\neg Q \land \neg S) \lor (S \land \neg S)) \land (P \lor Q) \\
&\equiv ((\neg P \land \neg R) \lor \mathbf{F}) \land ((\neg Q \land \neg S) \lor \mathbf{F}) \land (P \lor Q) \\
&\equiv (\neg P \land \neg R) \land (\neg Q \land \neg S) \land (P \lor Q) \\
&\equiv \neg R \land (\neg Q \land \neg S) \land (\neg P \land (P \lor Q)) \\
&\equiv \neg R \land (\neg Q \land \neg S) \land ((\neg P \land P) \lor (\neg P \land Q)) \\
&\equiv \neg R \land (\neg Q \land \neg S) \land (\mathbf{F} \lor (\neg P \land Q)) \\
&\equiv \neg R \land (\neg Q \land \neg S) \land (\neg P \land Q) \\
&\equiv \neg R \land \neg S \land \neg P \land (\neg Q \land Q) \\
&\equiv \neg R \land \neg S \land \neg P \land \mathbf{F} \\
&\equiv \mathbf{F}
\end{aligned}
$$

Argument

DEFINITION: An **argument** (论证) is a sequence of propositions

- **Conclusion**(结论): the final proposition
- **Premises**(假设)**:** all the other propositions
- **Valid**(有效): the truth of premises implies that of the conclusion
- **Proof** (证明): a valid argument that establishes the truth of a conclusion

EXAMPLE: a valid argument, a proof

- If $\{2^{-n}\}$ is convergent, then $\{2^{-n}\}$ has a convergent subsequence.
- $\{2^{-n}\}$ is convergent.
- $\{2^{-n}\}$ has a convergent subsequence.

Argument Form

DEFINITION: An **argument form**(论证形式) is a sequence of formulas.

- **Valid**(有效): no matter which propositions are substituted for the propositional variables, the truth of conclusion follows from the truth of premises
  - **Rules of inference**(推理规则): valid argument forms (relatively simple)

EXAMPLE: a valid argument form and an invalid argument form

$p \to q$

p

q

valid

p: $\{(-1)^n\}$ is convergent.

q: $\{(-1)^n\}$ has a convergent subsequence.

$p \to q$: If $\{(-1)^n\}$ is convergent, then $\{(-1)^n\}$ has a convergent subsequence.

$p \to q$

$\neg p$

$\neg q$

invalid

$\neg p$: $\{(-1)^n\}$ is not convergent.

$\neg q$: $\{(-1)^n\}$ does not have a convergent subsequence.

The truth of $\neg p$ and $p \to q$ does not imply that of $\neg q$

Building Arguments

QUESTION: Given the premises $P_1, \dots, P_n$, show a conclusion Q, that is, show that $P_1 \wedge \cdots \wedge P_n \Rightarrow Q$.

Name	Operations
Premise	Introduce the given formulas $P_1, \dots, P_n$ in the process of constructing proofs.
Conclusion	Quote the intermediate formula that have been deducted.
Rule of replacement	Replace a formula with a logically equivalent formula.
Rules of Inference	Deduct a new formula with a tautological implication.
Rule of substitution	Deduct a formula from a tautology.

Review: Tautological Implications

Name	Tautological Implication	NO.
Conjunction(合取)	$(P) \wedge (Q) \Rightarrow P \wedge Q$	1
Simplification(化简)	$P \wedge Q \Rightarrow P$	2
Addition(附加)	$P \Rightarrow P \vee Q$	3
Modus ponens(假言推理)	$P \wedge (P \to Q) \Rightarrow Q$	4
Modus tollens(拒取)	$\neg Q \wedge (P \to Q) \Rightarrow \neg P$	5
Disjunctive syllogism(析取三段论)	$\neg P \wedge (P \vee Q) \Rightarrow Q$	6
Hypothetical syllogism(假言三段论)	$(P \to Q) \wedge (Q \to R) \Rightarrow (P \to R)$	7
Resolution (归结)	$(P \vee Q) \wedge (\neg P \vee R) \Rightarrow Q \vee R$	8

Building Arguments

EXAMPLE: Show that the premises 1, 2, 3, and 4 lead to conclusion 5.

1. "It is not sunny this afternoon and it is colder than yesterday,"
2. "We will go swimming only if it is sunny,"
3. "If we do not go swimming, then we will take a canoe trip,"
4. "If we take a canoe trip, then we will be home by sunset"
5. "We will be home by sunset."

- **Translating the premises and the conclusion into formulas. Let**
  - p: "It is sunny this afternoon"
  - q: "It is colder than yesterday"
  - r: "We will go swimming"
  - s: "We will take a canoe trip"
  - t: "We will be home by sunset"
    - The premises are $\neg p \land q, r \rightarrow p, \neg r \rightarrow s$, and $s \rightarrow t$.
    - The conclusion is t.
- **Question: ?**$(\neg p \land q) \land (r \rightarrow p) \land (\neg r \rightarrow s) \land (s \rightarrow t) \Rightarrow t$
  - Can be proven with truth table. 32 rows!

Building Arguments

EXAMPLE: Show that the premises 1, 2, 3, and 4 lead to conclusion 5.

1. "It is not sunny this afternoon and it is colder than yesterday,"
2. "We will go swimming only if it is sunny,"
3. "If we do not go swimming, then we will take a canoe trip,"
4. "If we take a canoe trip, then we will be home by sunset"
5. "We will be home by sunset."

- **Show that** $(\neg p \wedge q) \wedge (r \rightarrow p) \wedge (\neg r \rightarrow s) \wedge (s \rightarrow t) \Rightarrow t$

(1)	$\neg p \wedge q$	Premise
(2)	$\neg p$	Simplification using (1)
(3)	$r \rightarrow p$	Premise
(4)	$\neg r$	Modus tollens using (2) and (3)
(5)	$\neg r \rightarrow s$	Premise
(6)	s	Modus ponens using (4) and (5)
(7)	$s \rightarrow t$	Premise
(8)	t	Modus ponens using (6) and (7)

Building Arguments

EXAMPLE: Show that $(P \vee Q) \wedge (P \rightarrow R) \wedge (Q \rightarrow S) \Rightarrow S \vee R$

(1)	$P \vee Q$	Premise
(2)	$\neg P \rightarrow Q$	Rule of replacement applied to (1)
(3)	$Q \rightarrow S$	Premise
(4)	$\neg P \rightarrow S$	Hypothetical syllogism applied to (2) and (3)
(5)	$\neg S \rightarrow P$	Rule of replacement applied to (4)
(6)	$P \rightarrow R$	Premise
(7)	$\neg S \rightarrow R$	Hypothetical syllogism applied to (5) and (6)
(8)	$S \vee R$	Rule of replacement applied to (7)

Discrete Mathematics: Lecture 20

argument (proposition), building arguments, predicate logic, quantifiers, WFFs

Xuming He
Associate Professor

School of Information Science and Technology
ShanghaiTech University

Spring Semester, 2022

Notes by Prof. Liangfeng Zhang

Argument

DEFINITION: An **argument** **(论证)** is a sequence of propositions

- **Conclusion(结论)**: the final proposition
- **Premises(假设):** all the other propositions
- **Valid(有效)**: the truth of premises implies that of the conclusion
- **Proof** **(证明)**: a valid argument that establishes the truth of a conclusion

EXAMPLE: a valid argument, a proof

- If $\{2^{-n}\}$ is convergent, then $\{2^{-n}\}$ has a convergent subsequence.
- $\{2^{-n}\}$ is convergent.
- $\{2^{-n}\}$ has a convergent subsequence.

Argument Form

DEFINITION: An **argument form**(论证形式) is a sequence of formulas.

- **Valid**(有效): no matter which propositions are substituted for the propositional variables, the truth of conclusion follows from the truth of premises
  - **Rules of inference**(推理规则): valid argument forms (relatively simple)

EXAMPLE: a valid argument form and an invalid argument form

$p \to q$

p

q

valid

p: $\{(-1)^n\}$ is convergent.

q: $\{(-1)^n\}$ has a convergent subsequence.

$p \to q$: If $\{(-1)^n\}$ is convergent, then $\{(-1)^n\}$ has a convergent subsequence.

$p \to q$

$\neg p$

$\neg q$

invalid

$\neg p$: $\{(-1)^n\}$ is not convergent.

$\neg q$: $\{(-1)^n\}$ does not have a convergent subsequence.

The truth of $\neg p$ and $p \to q$ does not imply that of $\neg q$

Building Arguments

QUESTION: Given the premises $P_1, \dots, P_n$, show a conclusion Q, that is, show that $P_1 \wedge \cdots \wedge P_n \Rightarrow Q$.

Name	Operations
Premise	Introduce the given formulas $P_1, \dots, P_n$ in the process of constructing proofs.
Conclusion	Quote the intermediate formula that have been deducted.
Rule of replacement	Replace a formula with a logically equivalent formula.
Rules of Inference	Deduct a new formula with a tautological implication.
Rule of substitution	Deduct a formula from a tautology.

Review: Tautological Implications

Name	Tautological Implication	NO.
Conjunction(合取)	$(P) \wedge (Q) \Rightarrow P \wedge Q$	1
Simplification(化简)	$P \wedge Q \Rightarrow P$	2
Addition(附加)	$P \Rightarrow P \vee Q$	3
Modus ponens(假言推理)	$P \wedge (P \rightarrow Q) \Rightarrow Q$	4
Modus tollens(拒取)	$\neg Q \wedge (P \rightarrow Q) \Rightarrow \neg P$	5
Disjunctive syllogism(析取三段论)	$\neg P \wedge (P \vee Q) \Rightarrow Q$	6
Hypothetical syllogism(假言三段论)	$(P \rightarrow Q) \wedge (Q \rightarrow R) \Rightarrow (P \rightarrow R)$	7
Resolution (归结)	$(P \vee Q) \wedge (\neg P \vee R) \Rightarrow Q \vee R$	8

Building Arguments

EXAMPLE: Show that the premises 1, 2, 3, and 4 lead to conclusion 5.

1. "It is not sunny this afternoon and it is colder than yesterday,"
2. "We will go swimming only if it is sunny,"
3. "If we do not go swimming, then we will take a canoe trip,"
4. "If we take a canoe trip, then we will be home by sunset"
5. "We will be home by sunset."

- **Translating the premises and the conclusion into formulas. Let**
  - p: "It is sunny this afternoon"
  - q: "It is colder than yesterday"
  - r: "We will go swimming"
  - s: "We will take a canoe trip"
  - t: "We will be home by sunset"
    - The premises are $\neg p \wedge q, r \rightarrow p, \neg r \rightarrow s$, and $s \rightarrow t$.
    - The conclusion is t.
- **Question: ?**$(\neg p \wedge q) \wedge (r \rightarrow p) \wedge (\neg r \rightarrow s) \wedge (s \rightarrow t) \Rightarrow t$
  - Can be proven with truth table. 32 rows!

Building Arguments

EXAMPLE: Show that the premises 1, 2, 3, and 4 lead to conclusion 5.

1. "It is not sunny this afternoon and it is colder than yesterday,"
2. "We will go swimming only if it is sunny,"
3. "If we do not go swimming, then we will take a canoe trip,"
4. "If we take a canoe trip, then we will be home by sunset"
5. "We will be home by sunset."

- **Show that** $(\neg p \wedge q) \wedge (r \rightarrow p) \wedge (\neg r \rightarrow s) \wedge (s \rightarrow t) \Rightarrow t$

(1)	$\neg p \wedge q$	Premise
(2)	$\neg p$	Simplification using (1)
(3)	$r \rightarrow p$	Premise
(4)	$\neg r$	Modus tollens using (2) and (3)
(5)	$\neg r \rightarrow s$	Premise
(6)	s	Modus ponens using (4) and (5)
(7)	$s \rightarrow t$	Premise
(8)	t	Modus ponens using (6) and (7)

Building Arguments

EXAMPLE: Show that $(P \vee Q) \wedge (P \rightarrow R) \wedge (Q \rightarrow S) \Rightarrow S \vee R$

(1)	$P \vee Q$	Premise
(2)	$\neg P \rightarrow Q$	Rule of replacement applied to (1)
(3)	$Q \rightarrow S$	Premise
(4)	$\neg P \rightarrow S$	Hypothetical syllogism applied to (2) and (3)
(5)	$\neg S \rightarrow P$	Rule of replacement applied to (4)
(6)	$P \rightarrow R$	Premise
(7)	$\neg S \rightarrow R$	Hypothetical syllogism applied to (5) and (6)
(8)	$S \vee R$	Rule of replacement applied to (7)

Limitation of Propositional Logic

EXAMPLE: What is the underlying tautological implication in the following proof?

- If 1/3 is a rational number, then 1/3 is a real number.
- 1/3 is a rational number.
- 1/3 is a real number.
    - $q \rightarrow r$:"If 1/3 is a rational number, then 1/3 is a real number.
    - q:"1/3 is a rational number"
    - r:"1/3 is a real number"
        - What is the underlying tautological implication?
            - $(q \rightarrow r) \wedge q \Rightarrow r$
                - YES. This is a tautological implication.

Limitation of Propositional Logic

EXAMPLE: What is the underlying tautological implication in the following proof?

- All rational numbers are real numbers
- 1/3 is a rational number
- 1/3 is a real number
  - p:“All rational numbers are real numbers”
  - q:“1/3 is a rational number”
  - r:“1/3 is a real number”
    - What is the underlying tautological implication?
      - $p \wedge q \Rightarrow r$?
        - NO. $p \wedge q \rightarrow r$ is not a tautology.
          - Why is this a proof?
            - We need **predicate logic**.

Predicate and Individual

Predicate谓词: describe the property of the subject term (in a sentence)

- A predicate is a function from a domain of individuals to $\{\mathbf{T}, \mathbf{F}\}$
- $\boldsymbol{n}$**-ary predicate**$_{n}$**元谓词**: a predicate on n individuals
  - I: "is an integer" // unary
  - G: "is greater than" //binary
- **Predicate constant**谓词常项: a concrete predicate // I, G
- **Predicate variable**谓词变项: a symbol that represents any predicate

Individual个体词: the object you are considering (in a sentence)

- "$\sqrt{1+2\sqrt{1+3\sqrt{1+\cdots}}}$ is an integer"
- "e^{π} is greater than π^{e}"
  - **Individual Constant**个体常项: $\sqrt{1+2\sqrt{1+3\sqrt{1+\cdots}}}$, e^{π}, π^{e}
  - **Individual Variable**个体变项: x, y, z
  - **Domain**个体域: the set of all individuals in consideration

From Predicates to Propositions

Propositional function命题函数: $P(x_1, \dots, x_n)$, where P is an n-ary predicate

- $P(x, y)$:“x is greater than y”
- $P(x, y)$ gives a proposition when we assign values to x, y
  - $P(e^{\pi}, \pi^{e})$ is a proposition (a true proposition)
- $P(x, y)$ is not a proposition

EXAMPLE: p:“Alice’s father is a doctor”; q:“Bob’s father is a doctor”

- Individuals: Alice’s father, Bob’s father; Predicate D: “is a doctor”
- $p = D(\text{Alice's father})$, $q = D(\text{Bob's father})$

Function of Individuals: a map on the domain of individuals

- $f(x) = x$’s father
- $p = D\big(f(\text{Alice})\big); q = D(f(\text{Bob}))$

Universal Quantifier

DEFINITION: Let $P(x)$ be a propositional function. The **universal quantification**全称量化 of $P(x)$ is "$P(x)$ for all x in the domain".

- notation: $\forall x\ P(x)$; read as "for all x $P(x)$" or "for every x $P(x)$"
  - "$\forall$" is called the **universal quantifier**全称量词
  - "$\forall x\ P(x)$" is true iff $P(x)$ is true for every x in the domain
  - "$\forall x\ P(x)$" is false iff there is an x_0 in the domain such that $P(x_0)$ is false
    - **Counterexample**反例: an x_0 such that $P(x_0)$ is false

EXAMPLE: $P(n)$:"$n^2 + n + 41$ is a prime"

- When domain = natural numbers, "$\forall n P(n)$" is "for every natural number n, $n^2 + n + 41$ is a prime"
- When domain is $D = \{0,1, \dots, 39\}$, "$\forall n P(n)$" is "for every $n \in D$, $n^2 + n + 41$ is a prime"

REMARK: <u>If the domain is empty, then "$\forall x\ P(x)$" is true for any P.</u>

Existential Quantifier

DEFINITION: Let $P(x)$ be a propositional function. The **existential quantification**存在量化 of $P(x)$ is "there is an x in the domain such that $P(x)$"

- notation: $\exists x\ P(x)$; read as "for some x $P(x)$" or "there is an x $\mathrm{s.t.}\ P(x)$"
  - "$\exists$" is called the **existential quantifier**存在量词
  - "$\exists x\ P(x)$" is true iff there is an x in the domain such that $P(x)$ is true
  - "$\exists x\ P(x)$" is false iff $P(x)$ is false for every x in the domain

EXAMPLE: $P(x)$:"$x^2 - x + 1 = 0$"

- "$\exists x\ P(x)$" is false when $D = \mathbb{R}$ and is true when $D = \mathbb{C}$

REMARK: <u>If the domain is empty, then "$\exists x\ P(x)$" is false for any P.</u>

REMARK: <u>if not stated, the individual can be anything.</u>

Binding Variables and Scope

DEFINITION: An individual variable x is **bound**约束的 if a quantifier $(\forall, \exists)$ is used on x; otherwise, x is said to be **free**自由的**.**

- $\exists x(x + y = 1)$
  - x is bound and y is free
- **scope**辖域 of a quantifier: the part of a formula to which a quantifier is used
  - the scope of $\exists x$ in $\exists x(x + y = 1)$ is $(x + y = 1)$

Predicate Logic谓词逻辑**:** the area of logic that deals with predicates and quantifiers (a.k.a. **predicate calculus**)

- predicate logic is an extension of propositional logic

Well-Formed Formulas

Elements that may appear in Well-Formed Formulas合式公式**:**

- Propositional constants: **T,F**, $p, q, r, \ldots$
- Propositional variables: $p, q, r, \ldots$
- Logical Connectives: $\neg, \wedge, \vee, \rightarrow, \leftrightarrow$
- Parenthesis: (,)
- Individual constants: $a, b, c, \ldots$
- Individual variables: $x, y, z, \ldots$
- Predicate constants: $P, Q, R, \ldots$
- Predicate variables: $P, Q, R, \ldots$
- Quantifiers: $\forall, \exists$
- Functions of individuals: $f, g, \ldots$

Well-Formed Formulas

DEFINIITON: well-formed formulas合式公式/formulas

1) propositional constants, propositional variables, and propositional functions without connectives are WFFs
2) If A is a WFF, then $\neg A$ is also a WFF
3) If A, B are WFFs and there is no individual variable x which is bound in one of A, B but free in the other, then $(A \wedge B), (A \vee B), (A \rightarrow B), (A \leftrightarrow B)$ are WFFs.
4) If A is a WFF with a free individual variable x, then $\forall x\ A, \exists x\ A$ are WFFs.
5) WFFs can be constructed with 1)-4).
   - Example: $\forall x\ F(x) \vee G(x), \forall x P(y)$ are not WFFs
   - Example: $\exists x\ (A(x) \rightarrow \forall y\ B(x, y))$ is a WFF

Precedence: $\forall, \exists$ have higher precedence than $\neg, \wedge, \vee, \rightarrow, \leftrightarrow$

- $\forall x\ P(x) \rightarrow Q(y)$ means $\left(\forall x P(x)\right) \rightarrow Q(y)$, not $\forall x(P(x) \rightarrow Q(y))$

Discrete Mathematics: Lecture 21

predicate logic, WFFs, from NL to WFFs, logic equivalence, tautological implication

Xuming He
Associate Professor

School of Information Science and Technology
ShanghaiTech University

Spring Semester, 2022

Notes by Prof. Liangfeng Zhang

Well-Formed Formulas

DEFINIITON: well-formed formulas合式公式/formulas

1) propositional constants, propositional variables, and propositional functions without connectives are WFFs
2) If A is a WFF, then $\neg A$ is also a WFF
3) If A, B are WFFs and there is no individual variable x which is bound in one of A, B but free in the other, then $(A \land B), (A \lor B), (A \to B), (A \leftrightarrow B)$ are WFFs.
4) If A is a WFF with a free individual variable x, then $\forall x\ A, \exists x\ A$ are WFFs.
5) WFFs can be constructed with 1)-4).
   - Example: $\forall x\ F(x) \lor G(x), \forall x P(y)$ are not WFFs
   - Example: $\exists x\ (A(x) \to \forall y\ B(x, y))$ is a WFF

Precedence: $\forall, \exists$ have higher precedence than $\neg, \land, \lor, \to, \leftrightarrow$

- $\forall x\ P(x) \to Q(y)$ means $\big(\forall x P(x)\big) \to Q(y)$, not $\forall x(P(x) \to Q(y))$

From Natural Language to WFFs

The Method of Translation:

- Introduce symbols to represent propositional constants, propositional variables, individual constants, individual variables, predicate constants, predicate variables, functions of individuals
- Construct WFFs with 1)-4) such that WFFs reflect the real meaning of the natural language

EXAMPLE: All irrational numbers are real numbers.

- Every irrational number is a real number.
- For every x, if x is an irrational number, then x is a real number.
  - $I(x) =$"x is an irrational number"
  - $R(x) =$"x is a real number"
    - Translation: $\forall x\ (I(x) \rightarrow R(x))$

From Natural Language to WFFs

EXAMPLE: Some real numbers are irrational numbers.

- There is a real number which is also an irrational number.
- There is an x such that x is a real number and also an irrational number.
  - $I(x) =$"x is an irrational number"
  - $R(x) =$"x is a real number"
    - Translation: $\exists x\ (R(x) \land I(x))$

EXAMPLE: There is a symbol that can not be understood by any person's brain.

- There is a symbol such that any person's brain can not understand it.
- There is an x such that x is a symbol and any person's brain can not understand x.
  - $S(x)$: "x is a symbol"
  - Translation: $\exists x\ (S(x) \land (\cdots))$

From Natural Language to WFFs

EXAMPLE: There is a symbol that can not be understood by any person's brain.

- Any person's brain can not understand x.
- For any y, if y is a person, then y's brain cannot understand x.
  - $P(y)$: "y is a person"
  - Translation: $\forall y\ (P(y) \rightarrow (\cdots))$
- y's brain cannot understand x
  - $U(z,x)$: "z can understand x"
  - $b(y)$ =the brain of y
    - Translation: $\neg U(b(y),x)$
- Translation: $\exists x\ \Big(S(x) \wedge \forall y\ \big(P(y) \rightarrow \neg U(b(y),x)\big)\Big)$

Interpretation

DEFINITION: an **interpretation**解释 requires one to (remove all uncertainty)

- assign a concrete proposition to every **proposition variable**
- assign a concrete predicate to every **predicate variable**
- restrict the domain of every **bound individual variable**
- assign a concrete individual to every **free individual variable**
- choose a concrete **function**, if there is any

EXAMPLE: $\exists x P(x) \rightarrow q$

- Domain of x={Alice, Bob, Eve}
- $P(x) =$"x gets A+"
- $q =$"I get A+"
- If at least one of Alice, Bob, and Eve gets A+, then I get A+.

Types of WFFs

DEFINITION: A WFF is **logically valid**普遍有效 if it is **T** in every interpretation

- $\forall x\ (P(x) \vee \neg P(x))$ is logically valid

DEFINITION: A WFF is **unsatisfiable**不可满足 if it is **F** in every interpretation

- $\exists x\ (P(x) \wedge \neg P(x))$ is unsatisfiable

DEFINITION: A WFF is **satisfiable**可满足 if it is **T** in some interpretation

- $\forall x\ (x^2 > 0)$
  - true when domain= nonzero real numbers

THEOREM: Let A be any WFF. A is logically valid iff $\neg A$ is unsatisfiable.

Rule of Substitution: Let A be a tautology in propositional logic. If we substitute any propositional variable in A with an arbitrary WFF from predicate logic, then we get a logically valid WFF.

- $p \vee \neg p$ is a tautology; hence, $P(x) \vee \neg P(x)$ is logically valid

Logical Equivalence

DEFINITION: Two WFFs A,B are **logically equivalent**等值 if they always have the same truth value in every interpretation.

- notation: $A \equiv B$; example: $\forall x\, P(x) \wedge \forall x\, Q(x) \equiv \forall x\, (P(x) \wedge Q(x))$

THEOREM: $A \equiv B$ iff $A \leftrightarrow B$ is logically valid.

- $A \equiv B$
- iff A, B have the same truth value in every interpretation I
- iff $A \leftrightarrow B$ is true in every interpretation I
- iff $A \leftrightarrow B$ is logically valid

THEOREM: $A \equiv B$ iff $A \rightarrow B$ and $B \rightarrow A$ are both logically valid.

- $A \leftrightarrow B \equiv (A \rightarrow B) \wedge (B \rightarrow A)$

Rule of Substitution

METHOD*:* Applying the rule of substitution to the logical equivalences in propositional logic, we get logical equivalences in predicate logic.

$$P \vee Q \equiv Q \vee P \qquad A(x) \vee B(y) \equiv B(y) \vee A(x)$$

$$(P \wedge Q) \wedge R \equiv P \wedge (Q \wedge R) \qquad \big(A(x) \wedge B(y)\big) \wedge c \equiv A(x) \wedge (B(y) \wedge c)$$

$$P \wedge (Q \vee R) \equiv (P \wedge Q) \vee (P \wedge R) \qquad A(x) \wedge (B(y) \vee c) \equiv \big(A(x) \wedge B(y)\big) \vee (A(x) \wedge c)$$

$$P \wedge (P \vee Q) \equiv P \qquad A(x) \wedge \big(A(x) \vee B(y)\big) \equiv A(x)$$

$$\neg(P \wedge Q) \equiv \neg P \vee \neg Q \qquad \neg\big(A(x) \wedge B(y)\big) \equiv \neg A(x) \vee \neg B(y)$$

$$P \rightarrow Q \equiv \neg P \vee Q \qquad A(x) \rightarrow (\forall y\, B(y)) \equiv \neg A(x) \vee \big(\forall y\, B(y)\big)$$

$$P \leftrightarrow Q \equiv (P \rightarrow Q) \wedge (Q \rightarrow P) \qquad A(x) \leftrightarrow c \equiv (A(x) \rightarrow c) \wedge (c \rightarrow A(x))$$

De Morgan's Laws for Quantifiers

THEOREM: $\neg\forall x\ P(x) \equiv \exists x\ \neg P(x)$

- Show that $\neg\forall x\ P(x) \rightarrow \exists x\ \neg P(x)$ is logically valid
  - Suppose that $\neg\forall x\ P(x)$ is $\mathbf{T}$ in an interpretation I
  - $\forall x\ P(x)$ is $\mathbf{F}$ in I
  - There is an x_0 such that $P(x_0)$ is $\mathbf{F}$ in I
  - There is an x_0 such that $\neg P(x_0)$ is $\mathbf{T}$ in I
  - $\exists x\ \neg P(x)$ is $\mathbf{T}$ in I
- Show that $\exists x\ \neg P(x) \rightarrow \neg\forall x\ P(x)$ is logically valid
  - Suppose that $\exists x \neg P(x)$ is $\mathbf{T}$ in an interpretation I
  - There is an x_0 such that $\neg P(x_0)$ is $\mathbf{T}$ in I
  - There is an x_0 such that $P(x_0)$ is $\mathbf{F}$ in I
  - $\forall x P(x)$ is $\mathbf{F}$ in I
  - $\neg\forall x P(x)$ is $\mathbf{T}$ in I

THEOREM: $\neg\exists x\ P(x) \equiv \forall x\ \neg P(x)$.

De Morgan’s Laws for Quantifiers

EXAMPLE: $R(x)$:“x is a real number”; $Q(x)$:“x is a rational number”

- $\neg\forall x\ (R(x) \rightarrow Q(x))$
  - Not all real numbers are rational numbers
- Negation: $\exists x\ \neg\big(R(x) \rightarrow Q(x)\big) \equiv \exists x\ (R(x) \wedge \neg Q(x))$
  - There is a real number which is not rational

EXAMPLE: Let the domain be the set of all real numbers. Let $Q(x)$:“x is a rational number” and $I(x)$:“x is an irrational number”

- $\neg\exists x\ (Q(x) \wedge I\ (x))$
  - No real number is both rational and irrational.
- Negation: $\forall x\ \neg\big(Q(x) \wedge I(x)\big) \equiv \forall x\ (\neg Q(x) \vee \neg I(x))$
  - Any real number is either not rational or not irrational.

Distributive Laws for Quantifiers

THEOREM: $\forall x\, (P(x) \wedge Q(x)) \equiv \forall x\, P(x) \wedge \forall x\, Q(x)$

- Show that $\forall x\, (P(x) \wedge Q(x)) \rightarrow \forall x\, P(x) \wedge \forall x\, Q(x)$ is logically valid
    - Suppose that $\forall x\, (P(x) \wedge Q(x))$ is $\mathbf{T}$ in an interpretation I
    - $P(x) \wedge Q(x)$ is $\mathbf{T}$ for every x in I
    - $P(x)$ is $\mathbf{T}$ for every x in I and $Q(x)$ is $\mathbf{T}$ for every x in I
    - $\forall x\, P(x)$ is $\mathbf{T}$ in I and $\forall x\, Q(x)$ is $\mathbf{T}$ in I
    - $\forall x\, P(x) \wedge \forall x\, Q(x)$ is $\mathbf{T}$ in I
- Show that $\forall x\, P(x) \wedge \forall x\, Q(x) \rightarrow \forall x\, (P(x) \wedge Q(x))$ is logically valid.
    - Suppose that $\forall x\, P(x) \wedge \forall x\, Q(x)$ is $\mathbf{T}$ in an interpretation I
    - $\forall x\, P(x)$ is $\mathbf{T}$ in I and $\forall x\, Q(x)$ is $\mathbf{T}$ in I
    - $P(x)$ is $\mathbf{T}$ for every x in I and $Q(x)$ is $\mathbf{T}$ for every x in I
    - $P(x) \wedge Q(x)$ is $\mathbf{T}$ for every x in I
    - $\forall x\, (P(x) \wedge Q(x))$ is $\mathbf{T}$ in I

THEOREM: $\exists x\, (P(x) \vee Q(x)) \equiv \exists x\, P(x) \vee \exists x\, Q(x)$.

Tautological Implication

DEFINITION: Let A and B be WFFs in predicate logic. A **tautologically implies** (重言蕴涵) B if every interpretation that causes A to be true causes B to be true.

- notation: $A \Rightarrow B$, called a **tautological implication** (重言蕴涵)

THEOREM: $A \Rightarrow B$ iff $A \to B$ is logically valid.

- $A \Rightarrow B$
- iff every interpretation that causes A to be true causes B to be true
- iff there is no interpretation such that $(A, B) = (\mathbf{T}, \mathbf{F})$
- Iff $A \to B$ is true in every interpretation
- iff $A \to B$ is logically valid

THEOREM: $A \Rightarrow B$ iff $A \land \neg B$ is unsatisfiable.

- $A \to B \equiv \neg A \lor B \equiv \neg(A \land \neg B)$

Rule of Substitution

Name	Tautological Implication	NO.
Conjunction(合取)	$(P) \land (Q) \Rightarrow P \land Q$	1
Simplification(化简)	$P \land Q \Rightarrow P$	2
Addition(附加)	$P \Rightarrow P \lor Q$	3
Modus ponens(假言推理)	$P \land (P \to Q) \Rightarrow Q$	4
Modus tollens(拒取)	$\neg Q \land (P \to Q) \Rightarrow \neg P$	5
Disjunctive syllogism(析取三段论)	$\neg P \land (P \lor Q) \Rightarrow Q$	6
Hypothetical syllogism(假言三段论)	$(P \to Q) \land (Q \to R) \Rightarrow (P \to R)$	7
Resolution (归结)	$(P \lor Q) \land (\neg P \lor R) \Rightarrow Q \lor R$	8

EXAMPLE: $P \land (P \to Q) \Rightarrow Q$ is a TI in propositional logic.

- $A(x) \land \big(A(x) \to B(y)\big) \Rightarrow B(y)$ must be a TI in predicate logic.
  - Rule of substitution: let $P = A(x)$ and $Q = B(y)$

Tautological Implications

- $\forall x P(x) \vee \forall x\, Q(x) \Rightarrow \forall x\, (P(x) \vee Q(x))$
- $\exists x\big(P(x) \wedge Q(x)\big) \Rightarrow \exists x\, P(x) \wedge \exists x Q(x)$
- $\forall x\, \big(P(x) \rightarrow Q(x)\big) \Rightarrow \forall x P(x) \rightarrow \forall x\, Q(x)$
- $\forall x\, \big(P(x) \rightarrow Q(x)\big) \Rightarrow \exists x\, P(x) \rightarrow \exists x\, Q(x)$
- $\forall x\, \big(P(x) \leftrightarrow Q(x)\big) \Rightarrow \forall x\, P(x) \leftrightarrow \forall x\, Q(x)$
- $\forall x\, \big(P(x) \leftrightarrow Q(x)\big) \Rightarrow \exists x\, P(x) \leftrightarrow \exists x\, Q(x)$
- $\forall x\, \big(P(x) \rightarrow Q(x)\big) \wedge \forall x\, \big(Q(x) \rightarrow R(x)\big) \Rightarrow \forall x\, (P(x) \rightarrow R(x))$
- $\forall x\, \big(P(x) \rightarrow Q(x)\big) \wedge P(a) \Rightarrow Q(a)$

Examples

EXAMPLE: $\underline{\forall x \left(P(x) \to Q(x)\right) \land P(a)} \Rightarrow \underline{Q(a)}$

- Suppose that the left hand side is true in an interpretation I (domain=D)
  - $\forall x \left(P(x) \to Q(x)\right)$ is $\mathbf{T}$ and $P(a)$ is $\mathbf{T}$
    - $P(a) \to Q(a)$ is $\mathbf{T}$ and $P(a)$ is $\mathbf{T}$
      - $Q(a)$ is $\mathbf{T}$ in I.

EXAMPLE: Tautological implication in the following proof?

- All rational numbers are real numbers $\boxed{\forall x\left(P(x) \to Q(x)\right)}$
- 1/3 is a rational number $\boxed{P(1/3)}$
- 1/3 is a real number $\boxed{Q(1/3)}$
  - $P(x) =$"x is a rational number"
  - $Q(x) =$"x is a real number"
  - rule of inference: $\forall x \left(P(x) \to Q(x)\right) \land P(1/3) \Rightarrow Q(1/3)$

Examples

EXAMPLE: $\underline{\forall x \left(P(x) \to Q(x)\right) \wedge \forall x \left(Q(x) \to R(x)\right)} \Rightarrow \underline{\forall x \left(P(x) \to R(x)\right)}$

- Suppose that the left hand side is $\mathbf{T}$ in an interpretation I (domain=D)
  - $\forall x \left(P(x) \to Q(x)\right)$ is $\mathbf{T}$ and $\forall x \left(Q(x) \to R(x)\right)$ is $\mathbf{T}$
  - $P(x) \to Q(x)$ is $\mathbf{T}$ for all $x \in D$ and $Q(x) \to R(x)$ is $\mathbf{T}$ for all $x \in D$
    - $P(x) \to R(x)$ is $\mathbf{T}$ for all $x \in D$
    - $\forall x \left(P(x) \to R(x)\right)$ is $\mathbf{T}$ in I.

EXAMPLE: Tautological implication in the following proof?

- All integers are rational numbers. $\boxed{\forall x\left(P(x) \to Q(x)\right)}$
- All rational numbers are real numbers. $\boxed{\forall x\left(Q(x) \to R(x)\right)}$
- All integers are real numbers. $\boxed{\forall x\left(P(x) \to R(x)\right)}$
  - $P(x) =$"x is an integer"
  - $Q(x) =$"x is a rational number"
  - $R(x) =$"x is a real number"
  - rule of inference: $\forall x \left(P(x) \to Q(x)\right) \wedge \forall x \left(Q(x) \to R(x)\right) \Rightarrow \forall x \left(P(x) \to R(x)\right)$

Discrete Mathematics: Lecture 22 (I)

logic equivalence, tautological implication, building arguments

Xuming He
Associate Professor

School of Information Science and Technology
ShanghaiTech University

Spring Semester, 2022

Notes by Prof. Liangfeng Zhang

Review: Types of WFFs (Proposition)

Tautology(重言式)**:** a WFF whose truth value is **T** for all truth assignment

- $p \vee \neg p$ is a tautology

Contradiction(矛盾式)**:** a WFF whose truth value is **F** for all truth assignment

- $p \wedge \neg p$ is a contradiction

Contingency(可能式)**:** neither tautology nor contradiction

- $p \rightarrow \neg p$ is a contingency

Satisfiable(可满足的):a WFF is satisfiable if it is true for at least one truth assignment

Rule of Substitution: (代入规则) Let B be a formula obtained from a tautology A by substituting a propositional variable in A with an arbitrary formula. Then B must be a tautology.

- $p \vee \neg p$ is a tautology: $(q \wedge r) \vee \neg(q \wedge r)$ is a tautology as well.

Review: Types of WFFs (Predicate)

DEFINITION: A WFF is **logically valid**普遍有效 if it is **T** in every interpretation

- $\forall x\ (P(x) \vee \neg P(x))$ is logically valid

DEFINITION: A WFF is **unsatisfiable**不可满足 if it is **F** in every interpretation

- $\exists x\ (P(x) \wedge \neg P(x))$ is unsatisfiable

DEFINITION: A WFF is **satisfiable**可满足 if it is **T** in some interpretation

- $\forall x\ (x^2 > 0)$
  - true when domain= nonzero real numbers

THEOREM: Let A be any WFF. A is logically valid iff $\neg A$ is unsatisfiable.

Rule of Substitution: Let A be a tautology in propositional logic. If we substitute any propositional variable in A with an arbitrary WFF from predicate logic, then we get a logically valid WFF.

- $p \vee \neg p$ is a tautology; hence, $P(x) \vee \neg P(x)$ is logically valid

Review: Logically Equivalent (Proposition)

DEFINITION: Let A and B be WFFs in propositional variables $p_1, \dots, p_n$.

- A and B are **logically equivalent** (等值) if they always have the same truth value for every truth assignment (of $p_1, \dots, p_n$)
  - Notation: $A \equiv B$

THEOREM: $A \equiv B$ if and only if $A \leftrightarrow B$ is a tautology.

- $A \equiv B$
- iff for any truth assignment, A, B take the same truth values
- iff for any truth assignment, $A \leftrightarrow B$ is true
- iff $A \leftrightarrow B$ is a tautology

THEOREM: $A \equiv A$; If $A \equiv B$, then $B \equiv A$; If $A \equiv B, B \equiv C$, then $A \equiv C$

QUESTION: How to prove $A \equiv B$?

Review: Logical Equivalence (Predicate)

DEFINITION: Two WFFs A,B are **logically equivalent**等值 if they always have the same truth value in every interpretation.

- notation: $A \equiv B$; example: $\forall x\, P(x) \wedge \forall x\, Q(x) \equiv \forall x\, (P(x) \wedge Q(x))$

THEOREM: $A \equiv B$ iff $A \leftrightarrow B$ is logically valid.

- $A \equiv B$
- iff A, B have the same truth value in every interpretation I
- iff $A \leftrightarrow B$ is true in every interpretation I
- iff $A \leftrightarrow B$ is logically valid

THEOREM: $A \equiv B$ iff $A \rightarrow B$ and $B \rightarrow A$ are both logically valid.

- $A \leftrightarrow B \equiv (A \rightarrow B) \wedge (B \rightarrow A)$

Review: Tautological Implications (Proposition)

DEFINITION: Let A and B be WFFs in propositional variables $p_1, \ldots, p_n$.

- A **tautologically implies** (重言蕴涵) B if every truth assignment that causes A to be true causes B to be true.
  - Notation: $A \Rightarrow B$, called a **tautological implication**
  - $A^{-1}(\mathbf{T}) \subseteq B^{-1}(\mathbf{T})$; $B^{-1}(\mathbf{F}) \subseteq A^{-1}(\mathbf{F})$

THEOREM: $A \Rightarrow B$ iff $A \to B$ is a tautology.

- $A \Rightarrow B$ iff $A^{-1}(\mathbf{T}) \subseteq B^{-1}(\mathbf{T})$ iff $A \to B$ is a tautology

THEOREM: $A \Rightarrow B$ iff $A \land \neg B$ is a contradiction.

- $A \to B \equiv \neg A \lor B \equiv \neg(A \land \neg B)$

Proving $\boldsymbol{A} \Rightarrow \boldsymbol{B}$: (1) $A^{-1}(\mathbf{T}) \subseteq B^{-1}(\mathbf{T})$; (2) $B^{-1}(\mathbf{F}) \subseteq A^{-1}(\mathbf{F})$;

(3) $A \to B$ is a tautology; (4) $A \land \neg B$ is a contradiction

Tautological Implication (Predicate)

DEFINITION: Let A and B be WFFs in predicate logic. A **tautologically implies**(重言蕴涵) B if every interpretation that causes A to be true causes B to be true.

- notation: $A \Rightarrow B$, called a **tautological implication**(重言蕴涵)

THEOREM: $A \Rightarrow B$ iff $A \rightarrow B$ is logically valid.

- $A \Rightarrow B$
- iff every interpretation that causes A to be true causes B to be true
- iff there is no interpretation such that $(A, B) = (\mathbf{T}, \mathbf{F})$
- Iff $A \rightarrow B$ is true in every interpretation
- iff $A \rightarrow B$ is logically valid

THEOREM: $A \Rightarrow B$ iff $A \wedge \neg B$ is unsatisfiable.

- $A \rightarrow B \equiv \neg A \vee B \equiv \neg(A \wedge \neg B)$

Rule of Substitution

Name	Tautological Implication	NO.
Conjunction(合取)	$(P) \wedge (Q) \Rightarrow P \wedge Q$	1
Simplification(化简)	$P \wedge Q \Rightarrow P$	2
Addition(附加)	$P \Rightarrow P \vee Q$	3
Modus ponens(假言推理)	$P \wedge (P \rightarrow Q) \Rightarrow Q$	4
Modus tollens(拒取)	$\neg Q \wedge (P \rightarrow Q) \Rightarrow \neg P$	5
Disjunctive syllogism(析取三段论)	$\neg P \wedge (P \vee Q) \Rightarrow Q$	6
Hypothetical syllogism(假言三段论)	$(P \rightarrow Q) \wedge (Q \rightarrow R) \Rightarrow (P \rightarrow R)$	7
Resolution (归结)	$(P \vee Q) \wedge (\neg P \vee R) \Rightarrow Q \vee R$	8

EXAMPLE: $P \wedge (P \rightarrow Q) \Rightarrow Q$ is a TI in propositional logic.

- $A(x) \wedge \big(A(x) \rightarrow B(y)\big) \Rightarrow B(y)$ must be a TI in predicate logic.
  - Rule of substitution: let $P = A(x)$ and $Q = B(y)$

Tautological Implications

- $\forall x P(x) \lor \forall x\, Q(x) \Rightarrow \forall x\, (P(x) \lor Q(x))$
- $\exists x\big(P(x) \land Q(x)\big) \Rightarrow \exists x\, P(x) \land \exists x Q(x)$
- $\forall x\, \big(P(x) \to Q(x)\big) \Rightarrow \forall x P(x) \to \forall x\, Q(x)$
- $\forall x\, \big(P(x) \to Q(x)\big) \Rightarrow \exists x\, P(x) \to \exists x\, Q(x)$
- $\forall x\, \big(P(x) \leftrightarrow Q(x)\big) \Rightarrow \forall x\, P(x) \leftrightarrow \forall x\, Q(x)$
- $\forall x\, \big(P(x) \leftrightarrow Q(x)\big) \Rightarrow \exists x\, P(x) \leftrightarrow \exists x\, Q(x)$
- $\forall x\, \big(P(x) \to Q(x)\big) \land \forall x\, \big(Q(x) \to R(x)\big) \Rightarrow \forall x\, (P(x) \to R(x))$
- $\forall x\, \big(P(x) \to Q(x)\big) \land P(a) \Rightarrow Q(a)$

Examples

EXAMPLE: $\underline{\forall x\,(P(x) \rightarrow Q(x)) \land P(a)} \Rightarrow \underline{Q(a)}$

- Suppose that the left hand side is true in an interpretation I (domain=D)
  - $\forall x\,(P(x) \rightarrow Q(x))$ is $\mathbf{T}$ and $P(a)$ is $\mathbf{T}$
    - $P(a) \rightarrow Q(a)$ is $\mathbf{T}$ and $P(a)$ is $\mathbf{T}$
      - $Q(a)$ is $\mathbf{T}$ in I.

EXAMPLE: Tautological implication in the following proof?

- All rational numbers are real numbers $\boxed{\forall x(P(x) \rightarrow Q(x))}$
- 1/3 is a rational number $\boxed{P(1/3)}$
- 1/3 is a real number $\boxed{Q(1/3)}$
  - $P(x) =$"x is a rational number"
  - $Q(x) =$"x is a real number"
  - rule of inference: $\forall x\,(P(x) \rightarrow Q(x)) \land P(1/3) \Rightarrow Q(1/3)$

Examples

EXAMPLE: $\underline{\forall x \left(P(x) \rightarrow Q(x)\right) \wedge \forall x \left(Q(x) \rightarrow R(x)\right)} \Rightarrow \underline{\forall x \left(P(x) \rightarrow R(x)\right)}$

- Suppose that the left hand side is $\mathbf{T}$ in an interpretation I (domain=D)
  - $\forall x \left(P(x) \rightarrow Q(x)\right)$ is $\mathbf{T}$ and $\forall x \left(Q(x) \rightarrow R(x)\right)$ is $\mathbf{T}$
  - $P(x) \rightarrow Q(x)$ is $\mathbf{T}$ for all $x \in D$ and $Q(x) \rightarrow R(x)$ is $\mathbf{T}$ for all $x \in D$
    - $P(x) \rightarrow R(x)$ is $\mathbf{T}$ for all $x \in D$
    - $\forall x \left(P(x) \rightarrow R(x)\right)$ is $\mathbf{T}$ in I.

EXAMPLE: Tautological implication in the following proof?

- All integers are rational numbers. $\boxed{\forall x\left(P(x) \rightarrow Q(x)\right)}$
- All rational numbers are real numbers. $\boxed{\forall x\left(Q(x) \rightarrow R(x)\right)}$
- All integers are real numbers. $\boxed{\forall x\left(P(x) \rightarrow R(x)\right)}$
  - $P(x) =$"x is an integer"
  - $Q(x) =$"x is a rational number"
  - $R(x) =$"x is a real number"
  - rule of inference: $\forall x \left(P(x) \rightarrow Q(x)\right) \wedge \forall x \left(Q(x) \rightarrow R(x)\right) \Rightarrow \forall x \left(P(x) \rightarrow R(x)\right)$

Building Arguments

QUESTION: Given the premises $P_1, \ldots, P_n$, show a conclusion Q, that is, show that $P_1 \land \cdots \land P_n \Rightarrow Q$.

Name	Operations
Premise	Introduce the given formulas $P_1, \ldots, P_n$ in the process of constructing proofs.
Conclusion	Quote the intermediate formula that have been deducted.
Rule of replacement	Replace a formula with a logically equivalent formula.
Rules of Inference	Deduct a new formula with a tautological implication.
Rule of substitution	Deduct a formula from a tautology.

Rules of Inference for ∀, ∃

Name	Rules of Inference	NO.
Universal Instantiation 全称量词消去	$\forall\, x\, P(x) \Rightarrow P(a)$ a is any individual in the domain of x	1
Universal Generalization 全称量词引入	$P(a) \Rightarrow \forall x\, P(x)$ a takes any individual in the domain of x	2
Existential Instantiation 存在量词消去	$\exists\, x\, P(x) \Rightarrow P(a)$ a is a specific individual in the domain of x	3
Existential Generalization 存在量词引入	$P(a) \Rightarrow \exists x\, P(x)$ a is a specific individual in the domain of x	4

Building Arguments

EXAMPLE: Show that the following premises 1, 2 lead to conclusion 3.

1. "A student in this class has not read the book," $\exists x\big(C(x) \land \neg B(x)\big)$
2. "Everyone in this class passed the exam," $\forall x(C(x) \to P(x))$
3. "Someone who passed the exam has not read the book." $\exists x(P(x) \land \neg B(x))$

- **Translate the premises and the conclusion into formulas.**
  - $C(x)$: "x is in the class"; $B(x)$: "x has read the book"; $P(x)$: "x passed the exam"
- **?**$\exists x\big(C(x) \land \neg B(x)\big) \land \forall x(C(x) \to P(x)) \Rightarrow \exists x(P(x) \land \neg B(x))$

(1)	$\exists x\big(C(x) \land \neg B(x)\big)$	Premise
(2)	$C(a) \land \neg B(a)$	Existential instantiation from (1)
(3)	$C(a)$	Simplification from (2)
(4)	$\forall x(C(x) \to P(x))$	Premise
(5)	$C(a) \to P(a)$	Universal instantiation from (4)
(6)	$P(a)$	Modus ponens from (3) and (5)
(7)	$\neg B(a)$	Simplification from (2)
(8)	$P(a) \land \neg B(a)$	Conjunction from (6) and (7)
(9)	$\exists x(P(x) \land \neg B(x))$	Existential generalization from (8)

Discrete Mathematics: Lecture 23

graph, vertex, edge, endpoints, directed, undirected, multiple edge, loop,

complete graph, cycle, wheel, cube

Xuming He
Associate Professor

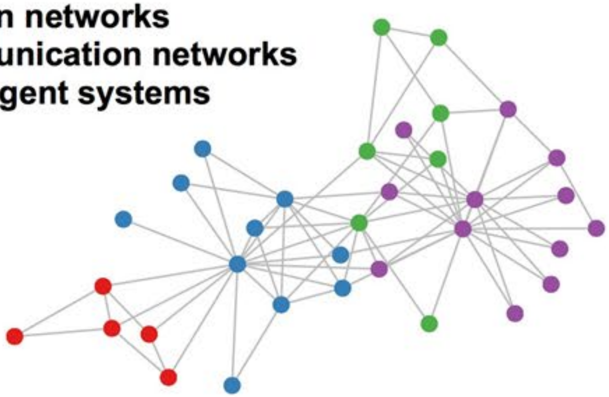
School of Information Science and Technology
ShanghaiTech University

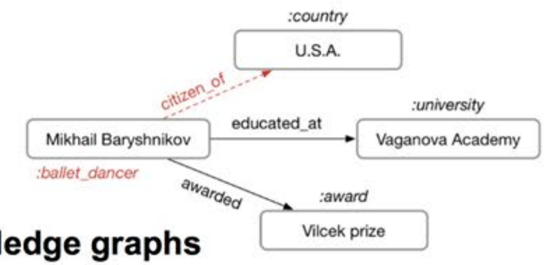
Spring Semester, 2022

Notes by Prof. Liangfeng Zhang

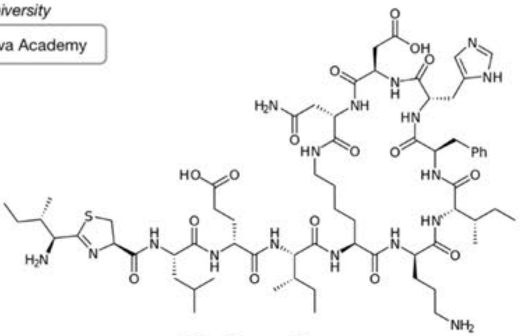
Real-world Graphs

Social networks
Citation networks
Communication networks
Multi-agent systems

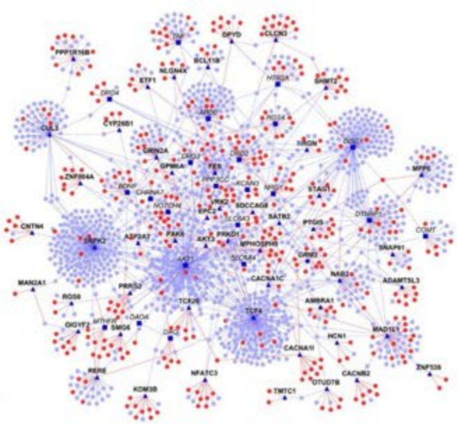




Knowledge graphs



Molecules



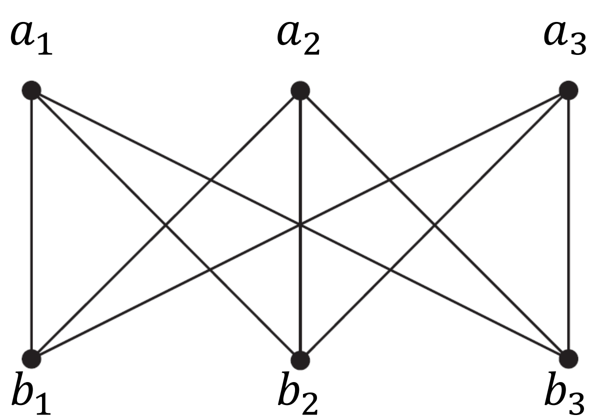
Protein interaction networks

Road maps

Graph

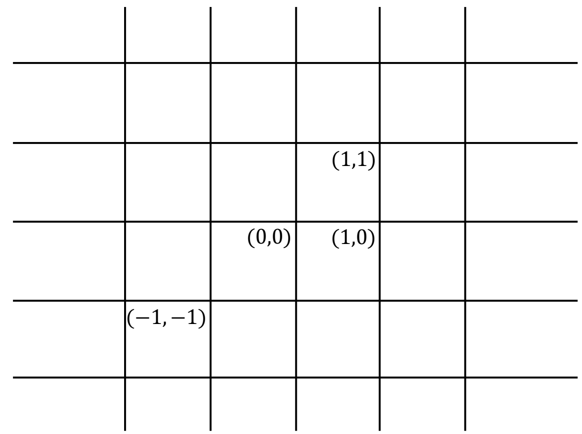
DEFINITION: A **graph** $G = (V, E)$ is defined by a nonempty set V of **vertices**顶点 and a set E of **edges**边, where each edge is associated with one or two vertices (called **endpoints**端点 of the edge).

- **Infinite Graph**无限图: $|V| = \infty$ or $|E| = \infty$
- **Finite Graph**有限图: $|V| < \infty$ and $|E| < \infty$; //$|V|$ is called the **order**阶数 of G



$V = \{a_1, a_2, a_3, b_1, b_2, b_3\}$

$E = \left\{\{a_i, b_j\}: i, j = 1,2,3\right\}$



$V = \{(i, j): i, j \in \mathbb{Z}\}$

$E = \{\{(a, b), (c, d)\}: |a - c| = 1 \text{ or } |b - d| = 1\}$

Graphs

- Loop & multiple edge

An edge with one endpoint is called a **loop**.
If there is more than one edge between two distinct vertices, it is called a **multiple edge**.

- Simple graph

A **simple graph** is a finite graph with no loops nor multiple edges.

- Weighted graph

A **weighted graph** is a graph $G = (V, E)$ such that each edge is assignated with a strictly positive number.

Graphs

- Directed graph

A **directed graph** $G = (V, E)$ consists of:

- V a non empty set of **vertices**,
- E a set of **directed edges**

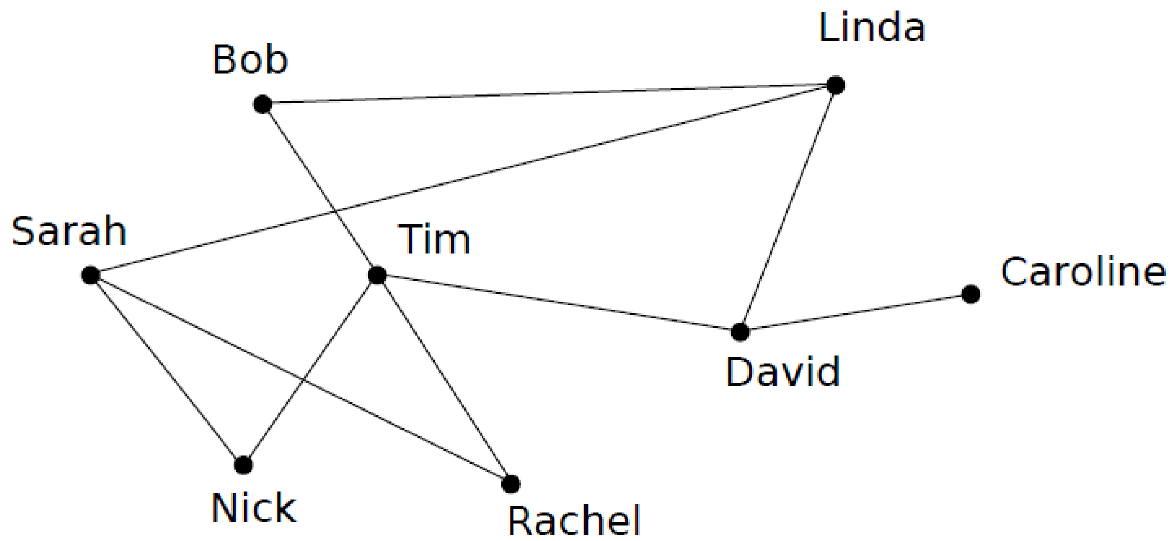
Each edge e is associated with an **ordered pair of vertices** (u, v), we say that e **starts at** u and **ends at** v.

- Subgraph

A **subgraph** of a graph $G = (V, E)$ is a graph $H = (W, F)$ where $W \subset V$, $F \subset E$. A subgraph H of G is a **proper subgraph** if $H \neq G$.

Graph Examples

Acquaintanceship Graph:



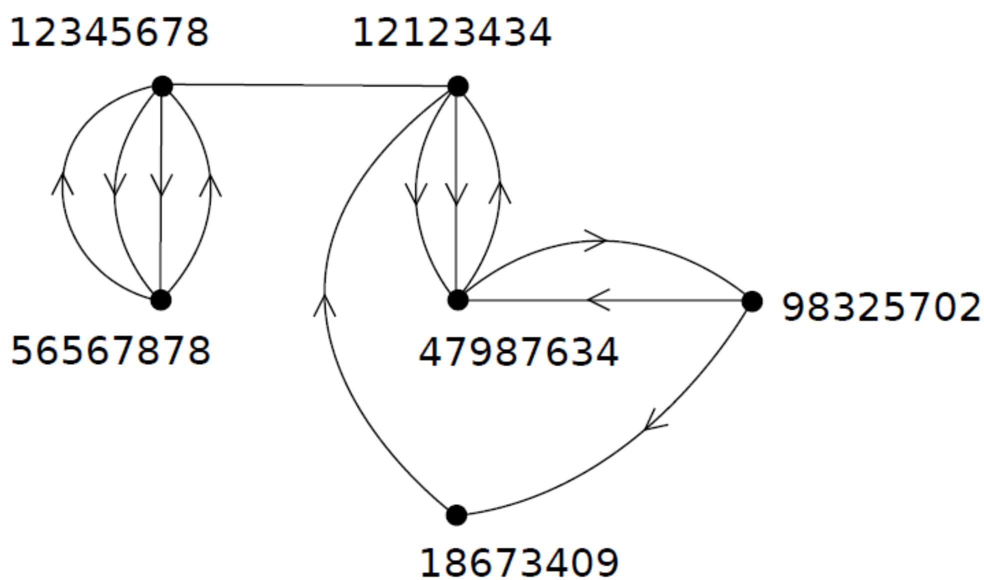
Tim knows Bob, David, Rachel and Nick. But Tim doesn't know Linda neither Caroline.

Simple graph, undirected

Graph Examples

Call Graphs: directed edges; the same edge may appear multiple times

- Vertices: telephone numbers
- Edges: there is an arc (u, v) if u called v
- AT&T experiment: calls during 20 days (290 million vertices and 4 billion edges)



Directed graph, multiple edges

Graph Examples

Precedence Graph

$S_1 \quad a := 0$

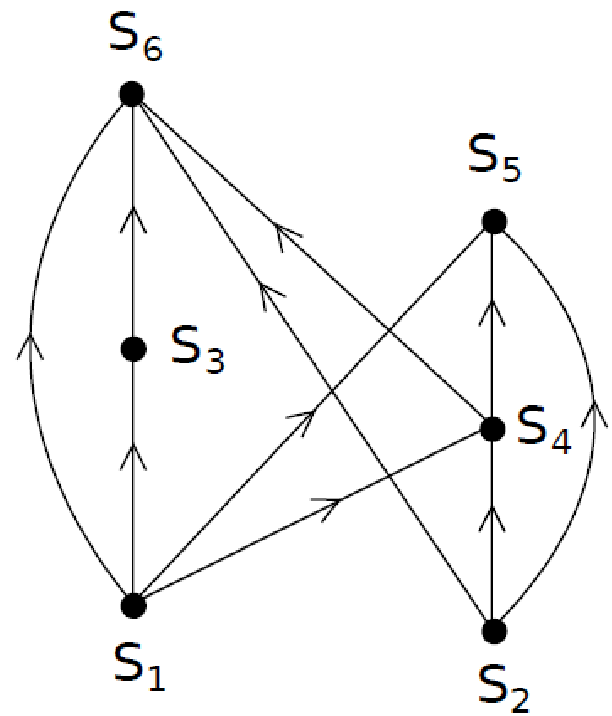
$S_2 \quad b := 1$

$S_3 \quad c := a + 1$

$S_4 \quad d := b + a$

$S_5 \quad e := d + 1$

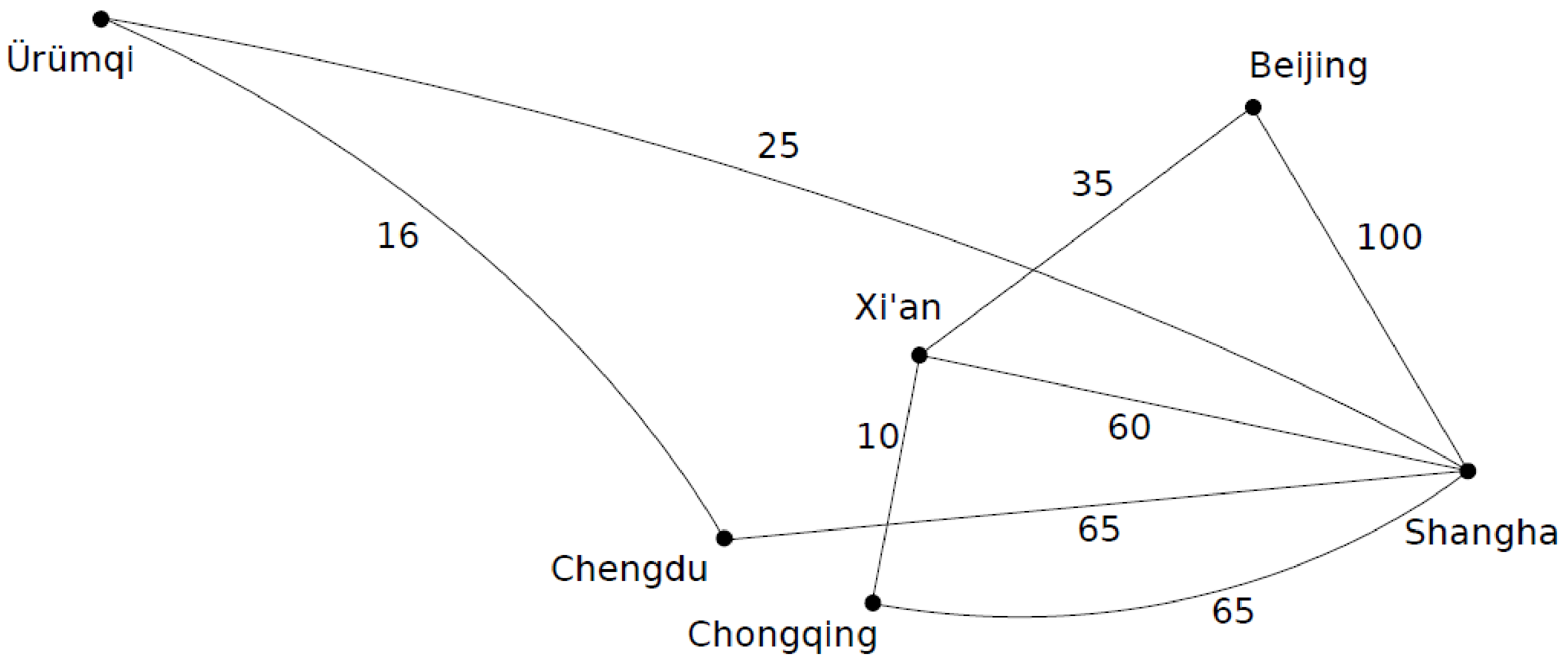
$S_6 \quad f := c + d$



Directed simple graph

Graph Examples

Flights



Weighted graph

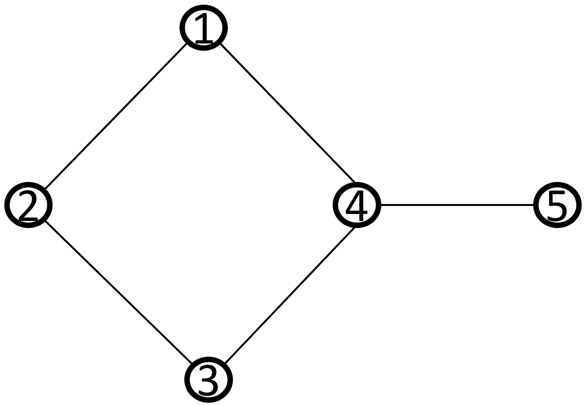
Types of Graphs

DEFINITION: Let $G = (V, E)$ be a graph with vertex set $V = \{v_1, \dots, v_n\}$.

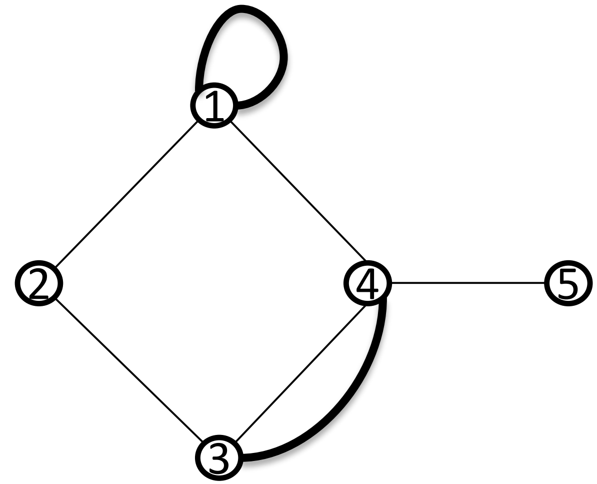
- **Question 1**: are the edges of G **directed**有向的?
  - No: G is an **undirected graph**无向图; the edge connecting v_i, v_j: $\{v_i, v_j\}$
  - Yes: G is a **directed graph**有向图; the edge starting at v_i and ending at v_j: (v_i, v_j)
- **Question 2**: are there **multiple edges**多重边connecting two different vertices v_i, v_j?
  - No: G is a **simple graph**简单图**;** Yes: G is a **multigraph**多重图
- **Question 3:** are there **loops**自环 connecting a vertex v_i to itself?
  - Yes: G is a **pseudograph**伪图

Type	Edges	Multiple Edges Allowed?	Loops Allowed?
Simple graph	undirected	No	No
Multigraph	undirected	Yes	No
Pseudograph	undirected	Yes	Yes
Simple directed graph	directed	No	No
Directed multigraph	directed	Yes	No
Mixed graph	undirected + directed	Yes	Yes

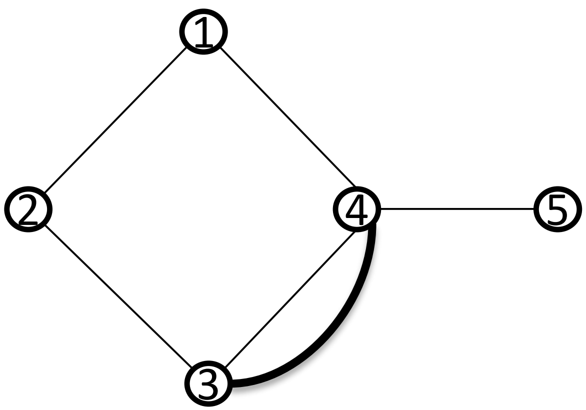
Types of Graphs



A Simple Graph (G_1)



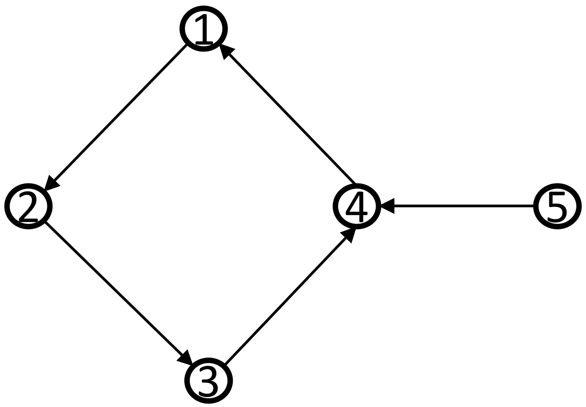
A Pseudograph (G_3)



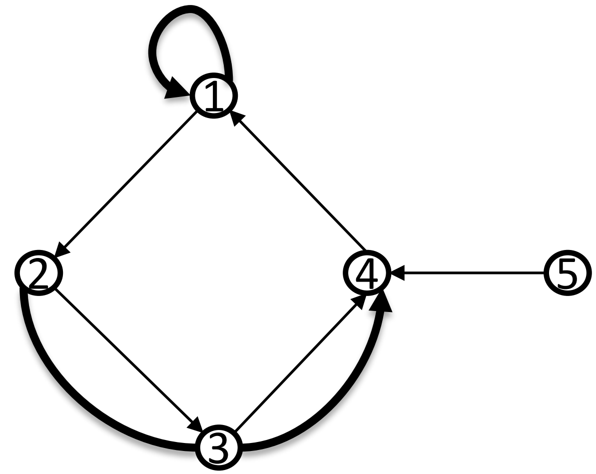
A Multigraph (G_2)

- Vertex set: $V = \{1,2,3,4,5\}$
- Edge set of G_1: $E = \{\{1,2\}, \{2,3\}, \{3,4\}, \{1,4\}, \{4,5\}\}$
- $\{4,5\}$ is an edge of the simple graph G_1
  - 4,5 are endpoints of the edge $\{4,5\}$
  - $\{4,5\}$ connects 4 and 5.
- $\{3,4\}$ is a multiple edge of the multigraph G_2
- There is a loop connecting 1 to itself in G_3

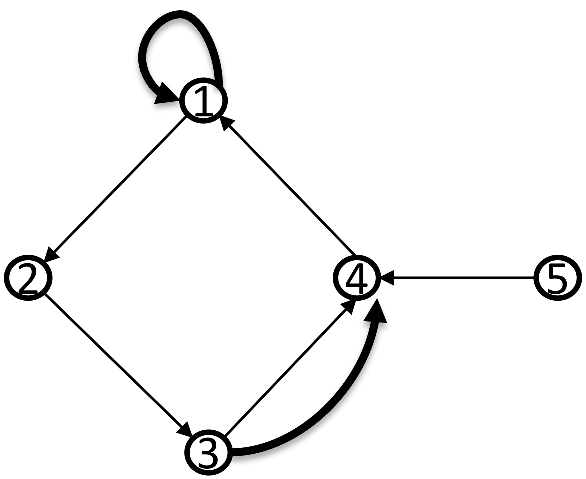
Types of Graphs



A Simple Directed Graph (G_4)



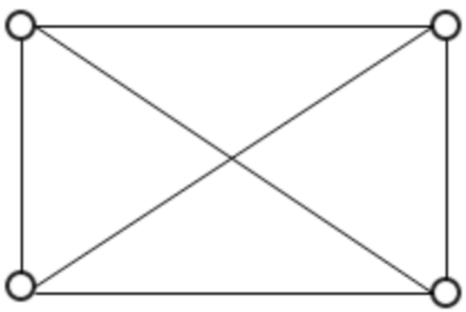
A Mixed Graph (G_6)



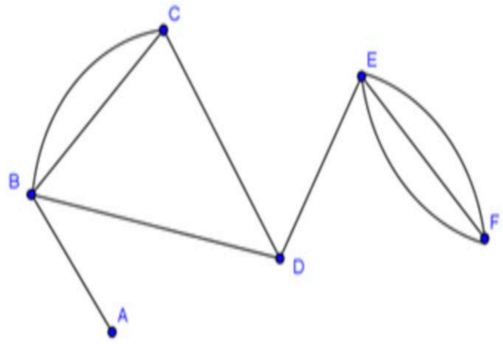
A Directed Pseudograph (G_5)

- Vertex set: $V = \{1,2,3,4,5\}$
- Edge set of G_4: $E = \{(1,2), (2,3), (3,4), (4,1), (5,4)\}$
  - $(5,4)$ is a directed edge
  - $(5,4)$ starts at 5 and ends at 4
- $(3,4)$ is a directed multiple edge in G_5
- There is a loop connecting 1 to itself in G_5

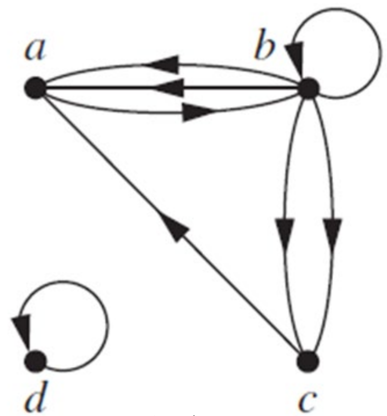
Bonus exercise



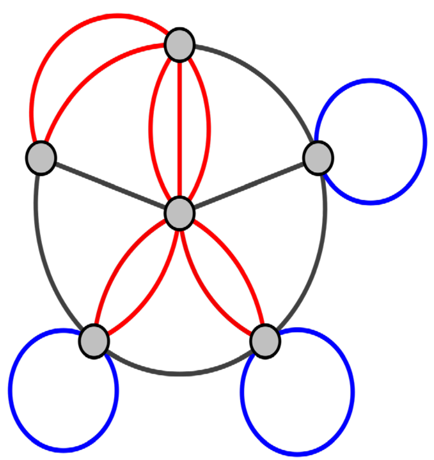
（1）



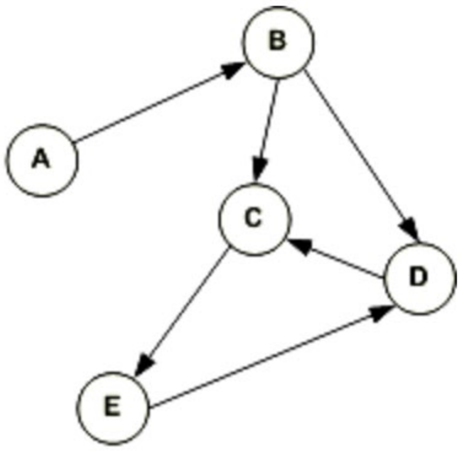
（3）



（5）

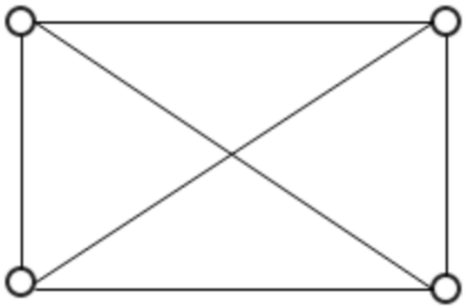


（2）

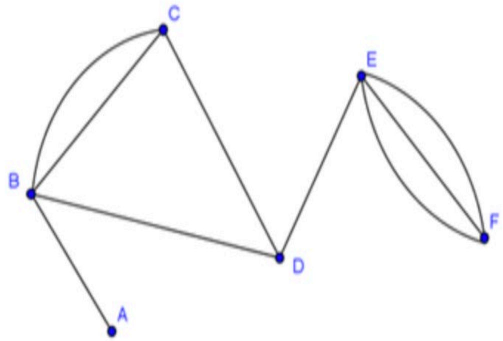


（4）

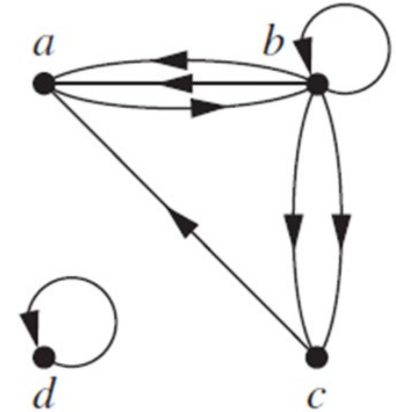
Bonus exercise



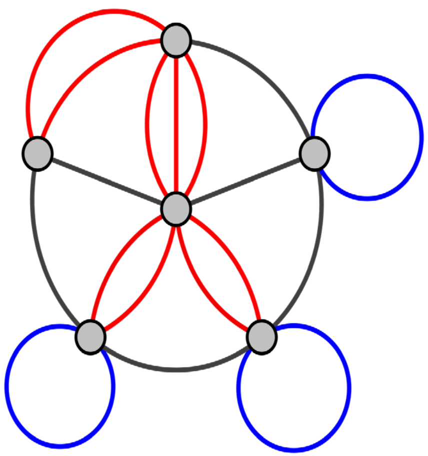
（1） simple graph



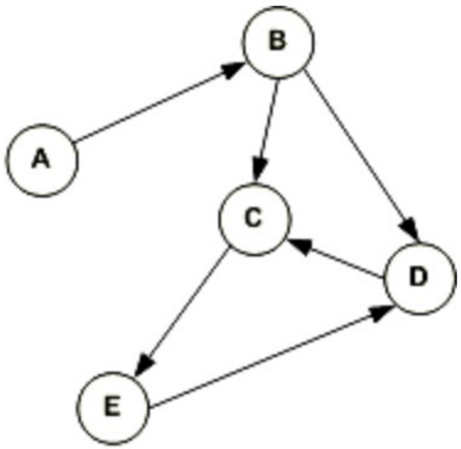
（3） multigraph



（5） directed pseudograph



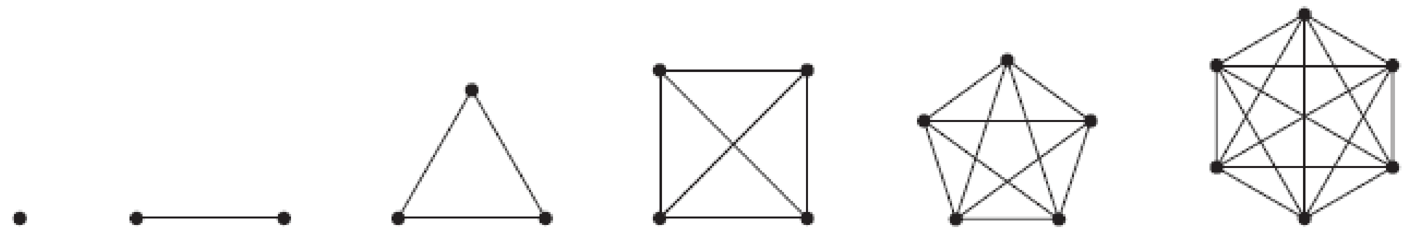
（2） pseudograph



（4） simple directed graph

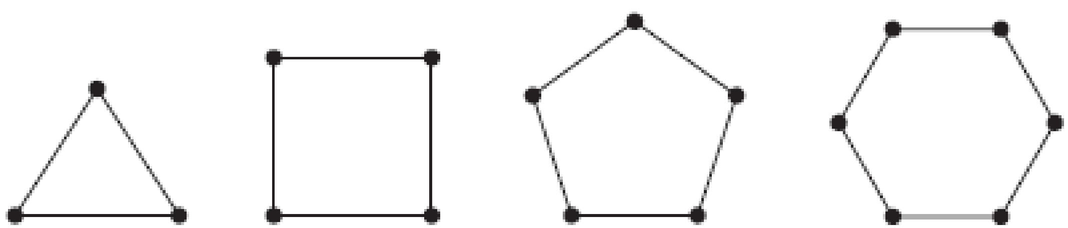
Special Simple Graphs

Complete Graph完全图 K_n: $V = \{v_1, \ldots, v_n\}$; $E = \left\{\{v_i, v_j\}: 1 \le i \ne j \le n\right\}$



K_1 K_2 K_3 K_4 K_5 K_6

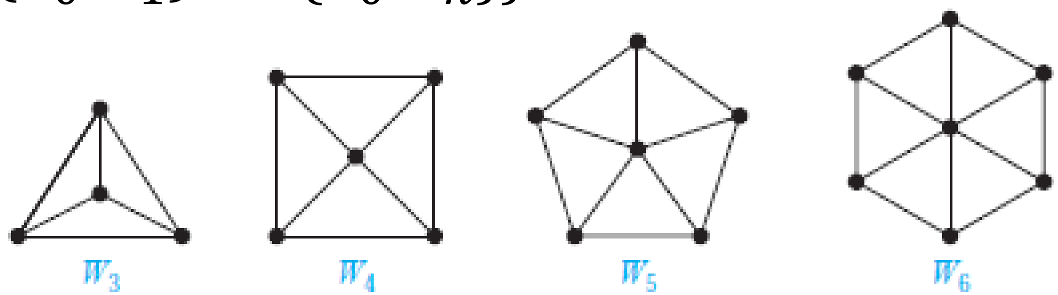
Cycle环,圈 C_n: $V = \{v_1, v_2, \ldots, v_n\}$; $E = \left\{\{v_1 v_2\}, \{v_2, v_3\}, \ldots, \{v_n, v_1\}\right\}$



C_3 C_4 C_5 C_6

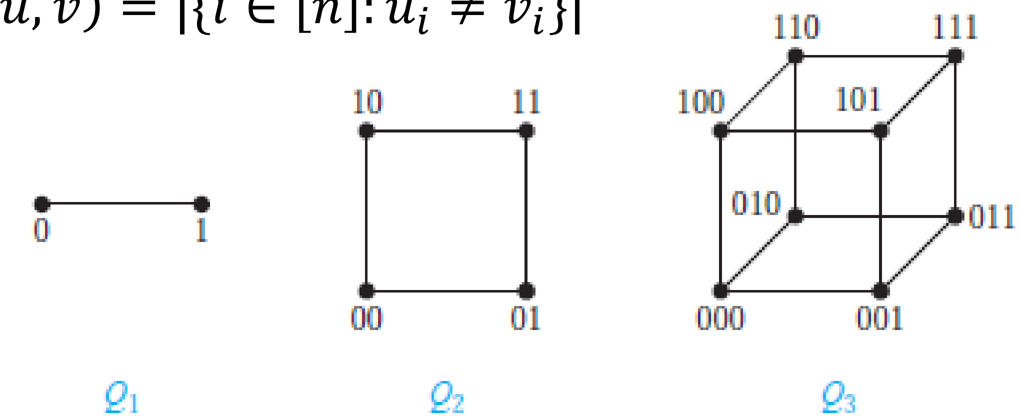
Special Simple Graphs

Wheel轮 W_n: $V = \{v_0, v_1, v_2, \dots, v_n\}$; $E = \{\{v_1, v_2\}, \dots, \{v_n, v_1\}\} \cup \{\{v_0, v_1\}, \dots, \{v_0, v_n\}\}$



n-**Cubes**方体 Q_n: $V = \{0,1\}^n$; $E = \{\{u, v\}: d(u, v) = 1\}$

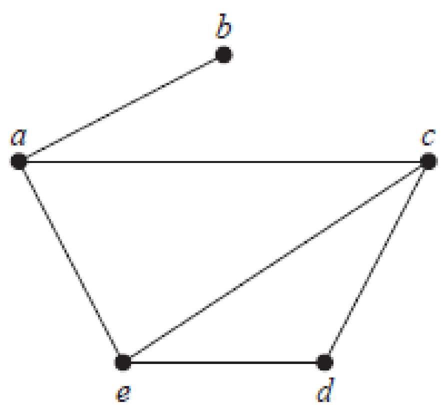
- $d(u, v) = |\{i \in [n]: u_i \neq v_i\}|$



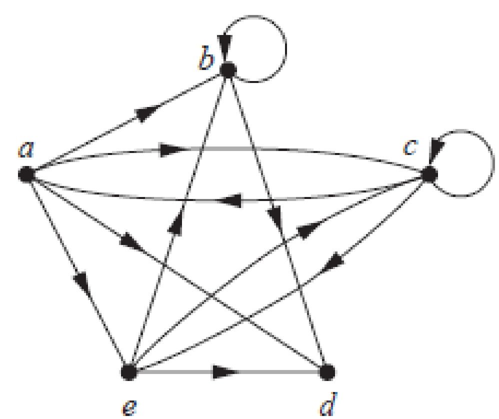
Adjacency List

DEFINITION: Let $G = (V, E)$ be a graph with no multiple edges. The **adjacency list**邻接表 of G is a list the vertices of the graph and all adjacent vertices

- $v_i, v_j \in V$ are **adjacent**相邻的 if $\{v_i, v_j\}$ or (v_i, v_j) is an edge



a	b, c, e
b	a
c	a, d, e
d	c, e
e	a, c, d

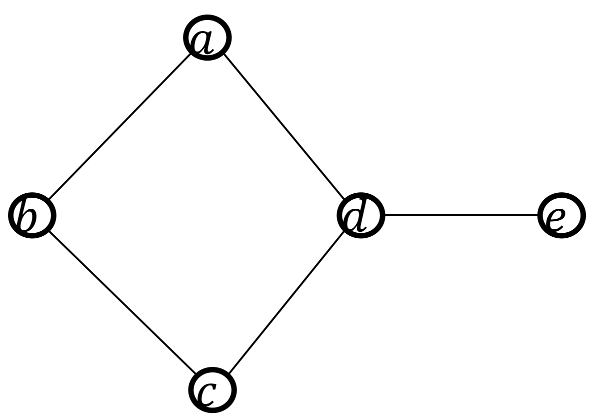


a	b, c, d, e
b	b, d
c	a, c, e
d	
e	b, c, d

Adjacency Matrix

DEFINITION: Let $G = (V = \{v_1, \dots, v_n\}, E)$ be a simple graph. The **adjacency matrix**邻接矩阵 of G is an $n \times n$ matrix $A = (a_{ij})$, where

$$a_{ij} = \begin{cases} 1 & \{v_i, v_j\} \in E \\ 0 & \{v_i, v_j\} \notin E \end{cases}$$

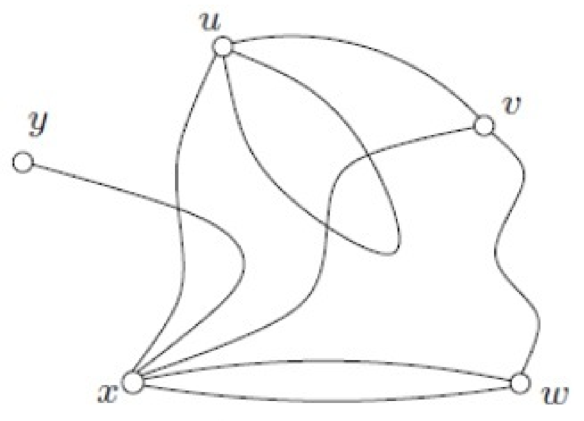


	a	b	c	d	e
a	0	1	0	1	0
b	1	0	1	0	0
c	0	1	0	1	0
d	1	0	1	0	1
e	0	0	0	1	0

Adjacency Matrix

DEFINITION: Let $G = (V = \{v_1, \ldots, v_n\}, E)$ be an <u>undirected graph</u>. The **adjacency matrix** of G is an $n \times n$ matrix $A = (a_{ij})$, where

- a_{ij} = **multiplicity**重数 of $\{v_i, v_j\}$ when $i \neq j$
- $a_{ii} = 1$ if $\exists$ a loop from v_i to itself; $a_{ii} = 0$, otherwise.

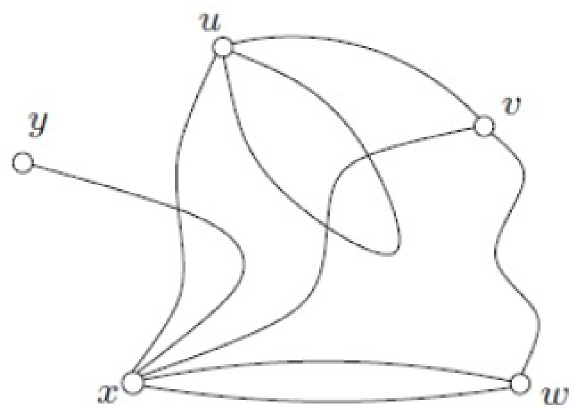


	u	v	w	x	y
u	$\boxed{1}$	1	0	1	0
v	1	0	1	1	0
w	0	1	0	2	0
x	1	1	2	0	1
y	0	0	0	1	0

Adjacency Matrix

DEFINITION: Let $G = (V = \{v_1, \ldots, v_n\}, E)$ be an <u>undirected graph</u>. The **adjacency matrix** of G is an $n \times n$ matrix $A = (a_{ij})$, where

- $a_{ij} =$ **multiplicity**重数 of $\{v_i, v_j\}$ when $i \neq j$
- $a_{ii} = 1$ if $\exists$ a loop from v_i to itself; $a_{ii} = 0$, otherwise.



	x	y	u	v	w
x	0	1	1	1	2
y	1	0	0	0	0
u	1	0	1	1	0
v	1	0	1	0	1
w	2	0	0	1	0

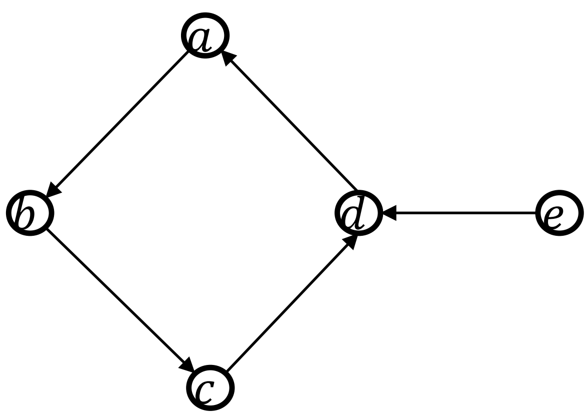
REMARKs: features of the adjacency matrices of undirected graphs

- The adjacency matrix depends on the ordering of the vertices
- The adjacency matrix of a simple graph is always symmetric
- The (i, j) entry counts the multiplicity of $\{v_i, v_j\}$, $i \neq j$

Adjacency Matrix

DEFINITION: Let $G = (V = \{v_1, \dots, v_n\}, E)$ be a simple directed graph. The **adjacency matrix** of G is an $n \times n$ matrix $A = (a_{ij})$, where

$$a_{ij} = \begin{cases} 1 & (v_i, v_j) \in E \\ 0 & (v_i, v_j) \notin E \end{cases}$$



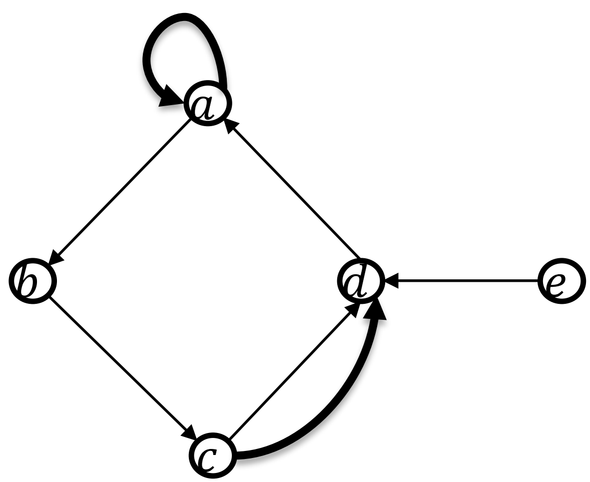
	a	b	c	d	e
a	0	1	0	0	0
b	0	0	1	0	0
c	0	0	0	1	0
d	1	0	0	0	0
e	0	0	0	1	0

REMARKS: The adjacency matrix is no longer symmetric

Adjacency Matrix

DEFINITION: Let $G = (V = \{v_1, \ldots, v_n\}, E)$ be a directed multigraph. The **adjacency matrix** of G is an $n \times n$ matrix $A = (a_{ij})$, where

$$a_{ij} = \begin{cases} \text{multiplicity of } (v_i, v_j) & (v_i, v_j) \in E \\ 0 & (v_i, v_j) \notin E \end{cases}$$



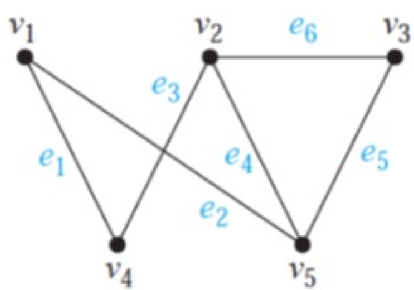
	a	b	c	d	e
a	1	1	0	0	0
b	0	0	1	0	0
c	0	0	0	2	0
d	1	0	0	0	0
e	0	0	0	1	0

Incidence Matrix

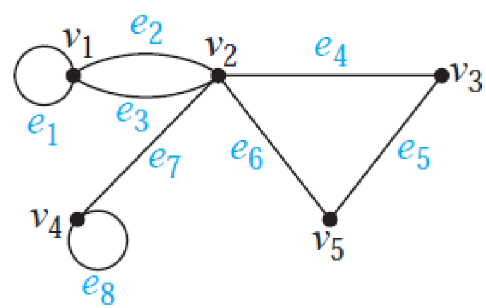
DEFINITION: Let $G = (V = \{v_1, \ldots, v_n\}, E = \{e_1\ , \ldots, e_m\})$ be <u>undirected</u>. The **incidence matrix**关联矩阵 of G is an $n \times m$ matrix $B = (b_{ij})$, where

$$b_{ij} = \begin{cases} 1 & \text{if } e_j \text{ is incident with } v_i \\ 0 & \text{otherwise} \end{cases}$$

- e_j **incident with** v_i: v_i is an endpoint of e_j



$$\begin{array}{cc} & \begin{array}{cccccc} e_1 & e_2 & e_3 & e_4 & e_5 & e_6 \end{array} \\ \begin{array}{c} v_1 \\ v_2 \\ v_3 \\ v_4 \\ v_5 \end{array} & \begin{bmatrix} 1 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 1 & 0 & 1 \\ 0 & 0 & 0 & 0 & 1 & 1 \\ 1 & 0 & 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 1 & 1 & 0 \end{bmatrix} \end{array}$$

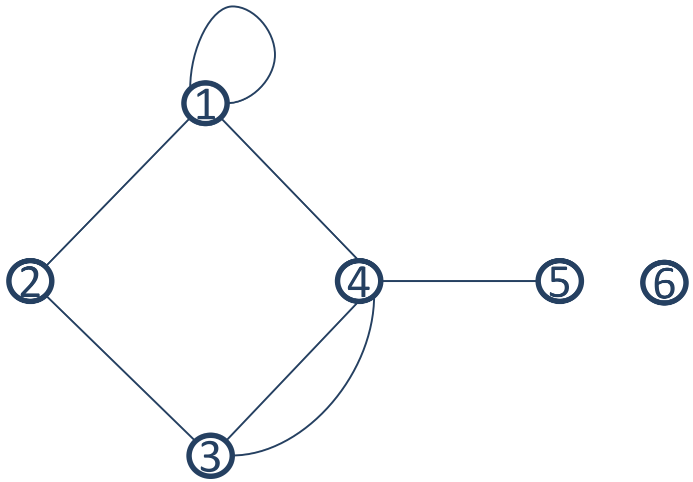


$$\begin{array}{cc} & \begin{array}{cccccccc} e_1 & e_2 & e_3 & e_4 & e_5 & e_6 & e_7 & e_8 \end{array} \\ \begin{array}{c} v_1 \\ v_2 \\ v_3 \\ v_4 \\ v_5 \end{array} & \begin{bmatrix} 1 & 1 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 1 & 1 & 0 & 1 & 1 & 0 \\ 0 & 0 & 0 & 1 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 1 \\ 0 & 0 & 0 & 0 & 1 & 1 & 0 & 0 \end{bmatrix} \end{array}$$

Degree

DEFINITION: Let $G = (V, E)$ be an <u>undirected</u> graph. We say that two vertices $u, v \in V$ are **adjacent**相邻的 (or **neighbors**邻居) if $\{u, v\} \in E$.

- **neighborhood**邻域 of v in G: $N(v) = \{u \in V : \{u, v\} \in E\}$
  - $N(A) = \cup_{v \in A} N(v)$ for $A \subseteq V$
- the **degree**度 $\deg(v)$ of $v \in V$ in G, is the number of edges incident with v
  - every loop from v to v contributes 2 to $\deg(v)$
- v is **isolated**孤立的 if $\deg(v) = 0$; v is **pendant**悬挂的 if $\deg(v) = 1$

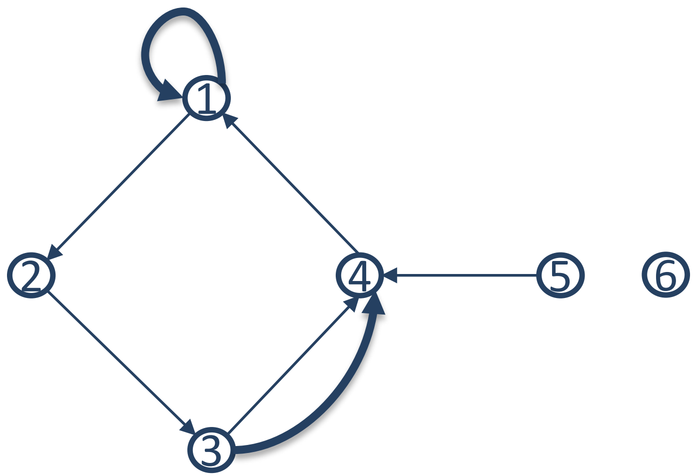


- 4 and 5 are adjacent
- $\{4,5\}$ is incident with 4 and 5
- $N(4) = \{1,3,5\}$; $N(\{1,4\}) = \{1,2,3,4,5\}$
- $\deg(1) = 4, \deg(2) = 2, \deg(3) = 3, \deg(4) = 4, \deg(5) = 1$
- 6 is isolated; 5 is pendant

Degree

DEFINITION: Let $G = (V, E)$ be a <u>directed</u> graph. If $(u, v) \in E$, we say that u is **adjacent to** v and v is **adjacent from** u.

- u is the **initial vertex**起始点 of (u, v); v is the **terminal vertex**终点 of (u, v)
  - $u = v$: u is the initial vertex and the terminal vertex
- **in-degree**入度 $\deg^{-}(v)$: the number of edges where v is the terminal vertex
- **out-degree**出度 $\deg^{+}(v)$: the number of edges where v is the initial vertex
  - $u = v$: the loop contributes 1 to $\deg^{-}(v)$ and 1 to $\deg^{+}(v)$



- 5 is adjacent to 4; 4 is adjacent from 5
- 5 is the initial vertex of (5,4)
- 4 is the terminal vertex of (5,4)
- 1 is the initial and terminal vertex of a loop
- $\deg^{-}(1) = 2\,; \deg^{+}(1) = 2$
- $\deg^{-}(4) = 3\,; \deg^{+}(4) = 1$

Handshaking Theorem

THEOREM: Let $G = (V, E)$ be an undirected graph. Then

$2|E| = \sum_{v \in V} \deg(v)$ and $|\{v \in V : \deg(v) \text{ is odd}\}|$ is even.

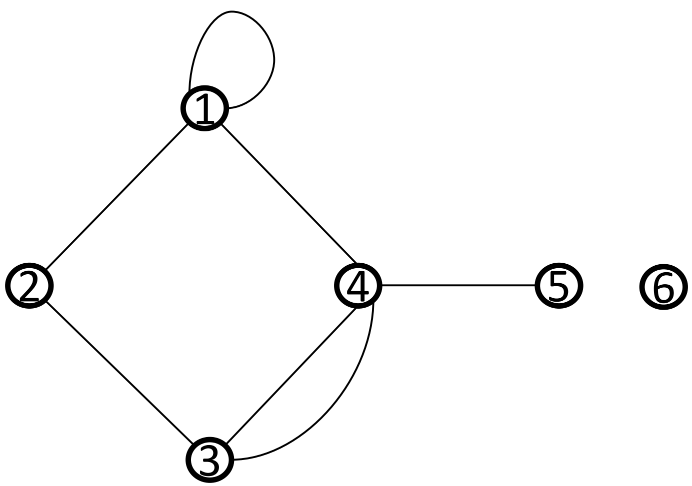
- Any edge $e \in E$ contribute 2 to the sum $\sum_{v \in V} \deg(v)$
  - $e = \{v_i, v_j\}$: e contributes 1 to $\deg(v_i)$ and 1 to $\deg(v_j)$
  - $e = \{v_i\}$: e contributes 2 to $\deg(v_i)$
- The m edges contribute $2|E|$ to $\sum_{v \in V} \deg(v)$.
  - Hence, $\sum_{v \in V} \deg(v) = 2|E|$
- $\sum_{v \in V} \deg(v) = \sum_{v \in V: 2 | \deg(v)} \deg(v) + \sum_{v \in V: 2 \nmid \deg(v)} \deg(v)$
  - $2 | \sum_{v \in V} \deg(v)$; $2 | \sum_{v \in V: 2 | \deg(v))} \deg(v)$
    - $2 | \sum_{v \in V: 2 \nmid \deg(v)} \deg(v)$
      - $|\{v \in V : \deg(v) \text{ is odd}\}|$ must be even

Handshaking Theorem

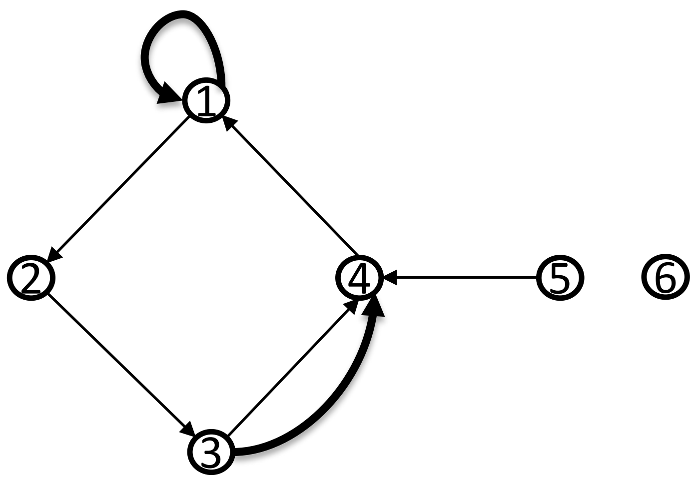
THEOREM: Let $G = (V, E)$ be a <u>directed</u> graph. Then

$$\sum_{v \in V} \deg^-(v) = \sum_{v \in V} \deg^+(v) = |E|.$$

- Every edge $e \in E$ contributes 1 to $\sum_{v \in V} \deg^-(v)$
  - $e = (v_i, v_j)$ contributes 1 to $\deg^-(v_i)$
- Hence, $\sum_{v \in V} \deg^-(v) = |E|$



v	1	2	3	4	5	6
$\deg(v)$	4	2	3	4	1	0



v	1	2	3	4	5	6
$\deg^-(v)$	2	1	1	3	0	0
$\deg^+(v)$	2	1	2	1	1	0

Discrete Mathematics: Lecture 24

Degree, Handshaking Theorem, Graph Transform, Graph Isomorphism, Bipartite Graph, Matching

Xuming He
Associate Professor

School of Information Science and Technology
ShanghaiTech University

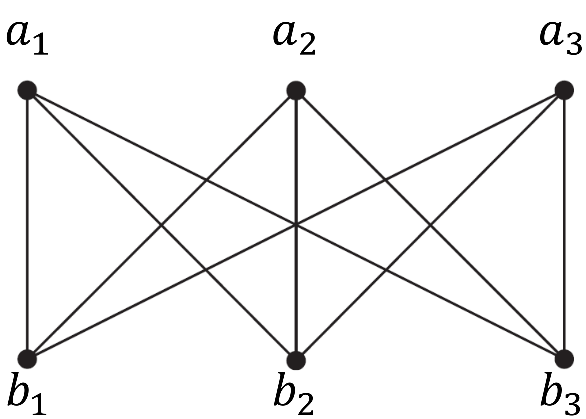
Spring Semester, 2022

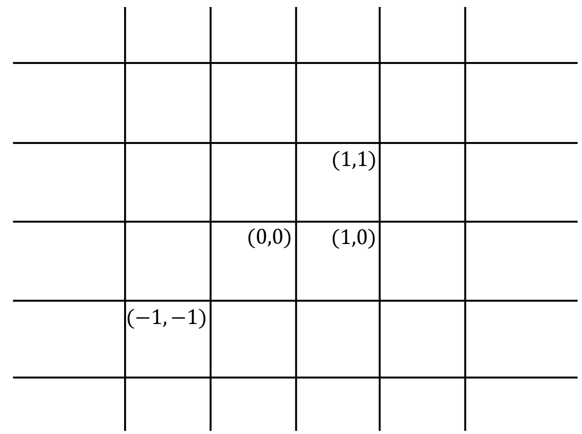
Notes by Prof. Liangfeng Zhang

Graph

DEFINITION: A **graph** $G = (V, E)$ is defined by a nonempty set V of **vertices**顶点 and a set E of **edges**边, where each edge is associated with one or two vertices (called **endpoints**端点 of the edge).

- **Infinite Graph**无限图: $|V| = \infty$ or $|E| = \infty$
- **Finite Graph**有限图: $|V| < \infty$ and $|E| < \infty$; //$|V|$ is called the **order**阶数 of G





$V = \{a_1, a_2, a_3, b_1, b_2, b_3\}$
$E = \left\{\{a_i, b_j\}: i, j = 1,2,3\right\}$

$V = \{(i, j): i, j \in \mathbb{Z}\}$
$E = \{\{(a, b), (c, d)\}: |a - c| = 1 \text{ or } |b - d| = 1\}$

Types of Graphs

DEFINITION: Let $G = (V, E)$ be a graph with vertex set $V = \{v_1, \dots, v_n\}$.

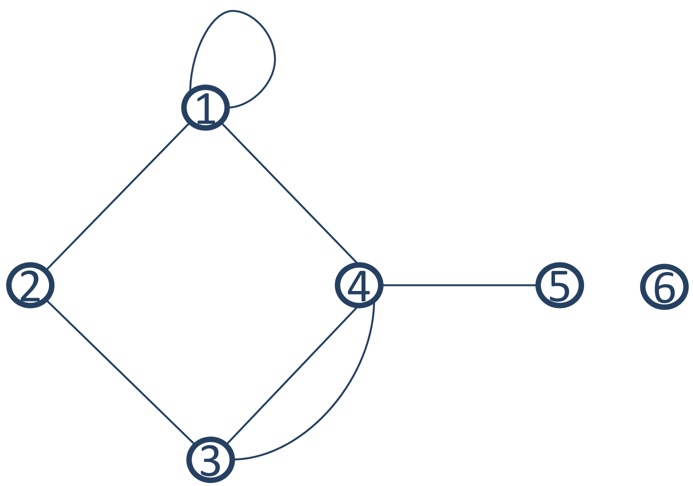
- **Question 1**: are the edges of G **directed**有向的?
  - No: G is an **undirected graph**无向图; the edge connecting v_i, v_j: $\{v_i, v_j\}$
  - Yes: G is a **directed graph**有向图; the edge starting at v_i and ending at v_j: (v_i, v_j)
- **Question 2**: are there **multiple edges**多重边connecting two different vertices v_i, v_j?
  - No: G is a **simple graph**简单图**;** Yes: G is a **multigraph**多重图
- **Question 3:** are there **loops**自环 connecting a vertex v_i to itself?
  - Yes: G is a **pseudograph**伪图

Type	Edges	Multiple Edges Allowed?	Loops Allowed?
Simple graph	undirected	No	No
Multigraph	undirected	Yes	No
Pseudograph	undirected	Yes	Yes
Simple directed graph	directed	No	No
Directed multigraph	directed	Yes	No
Mixed graph	undirected + directed	Yes	Yes

Degree

DEFINITION: Let $G=(V,E)$ be an undirected graph. We say that two vertices $u,v\in V$ are **adjacent**相邻的 (or **neighbors**邻居) if $\{u,v\}\in E$.

- **neighborhood**邻域 of v in G: $N(v)=\{u\in V:\{u,v\}\in E\}$
  - $N(A)=\cup_{v\in A}N(v)$ for $A\subseteq V$
- the **degree**度 $\deg(v)$ of $v\in V$ in G, is the number of edges incident with v
  - every loop from v to v contributes 2 to $\deg(v)$
- v is **isolated**孤立的 if $\deg(v)=0$; v is **pendant**悬挂的 if $\deg(v)=1$

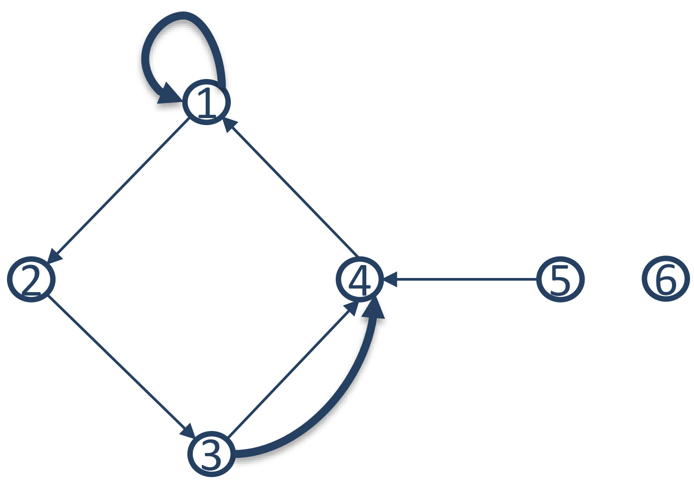


- 4 and 5 are adjacent
- $\{4{,}5\}$ is incident with 4 and 5
- $N(4)=\{1{,}3{,}5\}$; $N(\{1{,}4\})=\{1{,}2{,}3{,}4,5\}$
- $\deg(1)=4,\deg(2)=2,\deg(3)=3,\deg(4)=4,\deg(5)=1$
- 6 is isolated; 5 is pendant

Degree

DEFINITION: Let $G = (V, E)$ be a <u>directed</u> graph. If $(u, v) \in E$, we say that u is **adjacent to** v and v is **adjacent from** u.

- u is the **initial vertex**起始点 of (u, v); v is the **terminal vertex**终点 of (u, v)
  - $u = v$: u is the initial vertex and the terminal vertex
- **in-degree**入度 $\deg^{-}(v)$: the number of edges where v is the terminal vertex
- **out-degree**出度 $\deg^{+}(v)$: the number of edges where v is the initial vertex
  - $u = v$: the loop contributes 1 to $\deg^{-}(v)$ and 1 to $\deg^{+}(v)$



- 5 is adjacent to 4; 4 is adjacent from 5
- 5 is the initial vertex of (5,4)
- 4 is the terminal vertex of (5,4)
- 1 is the initial and terminal vertex of a loop
- $\deg^{-}(1) = 2\,; \deg^{+}(1) = 2$
- $\deg^{-}(4) = 3\,; \deg^{+}(4) = 1$

Handshaking Theorem

THEOREM: Let $G = (V, E)$ be an undirected graph. Then

$2|E| = \sum_{v \in V} \deg(v)$ and $|\{v \in V : \deg(v) \text{ is odd}\}|$ is even.

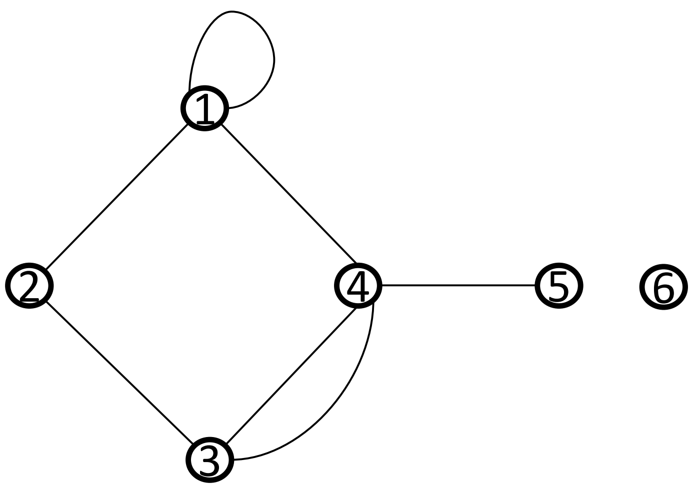
- Any edge $e \in E$ contribute 2 to the sum $\sum_{v \in V} \deg(v)$
  - $e = \{v_i, v_j\}$: e contributes 1 to $\deg(v_i)$ and 1 to $\deg(v_j)$
  - $e = \{v_i\}$: e contributes 2 to $\deg(v_i)$
- The m edges contribute $2|E|$ to $\sum_{v \in V} \deg(v)$.
  - Hence, $\sum_{v \in V} \deg(v) = 2|E|$
- $\sum_{v \in V} \deg(v) = \sum_{v \in V : 2 | \deg(v)} \deg(v) + \sum_{v \in V : 2 \nmid \deg(v)} \deg(v)$
  - $2 | \sum_{v \in V} \deg(v);\ 2 | \sum_{v \in V : 2 | \deg(v))} \deg(v)$
    - $2 | \sum_{v \in V : 2 \nmid \deg(v)} \deg(v)$
      - $|\{v \in V : \deg(v) \text{ is odd}\}|$ must be even

Handshaking Theorem

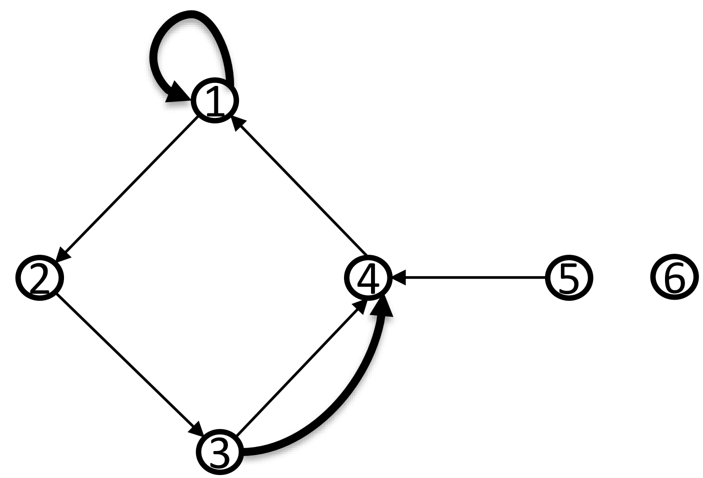
THEOREM: Let $G = (V, E)$ be a directed graph. Then

$$\sum_{v \in V} \deg^-(v) = \sum_{v \in V} \deg^+(v) = |E|.$$

- Every edge $e \in E$ contributes 1 to $\sum_{v \in V} \deg^-(v)$
  - $e = (v_i, v_j)$ contributes 1 to $\deg^-(v_i)$
- Hence, $\sum_{v \in V} \deg^-(v) = |E|$



v	1	2	3	4	5	6
$\deg(v)$	4	2	3	4	1	0



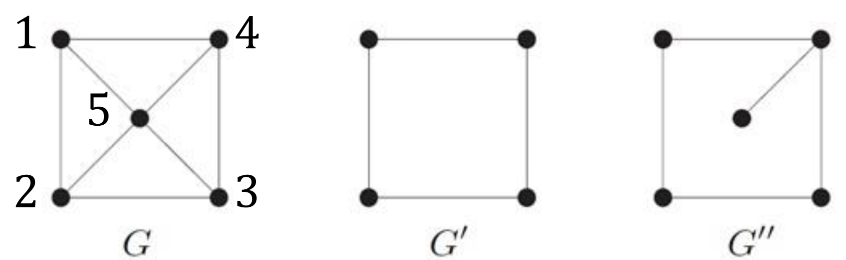
v	1	2	3	4	5	6
$\deg^-(v)$	2	1	1	3	0	0
$\deg^+(v)$	2	1	2	1	1	0

Subgraph

DEFINITION: Let $G = (V, E)$ be a simple graph. $H = (W, F)$ is a **subgraph**子图 of G if $W \subseteq V$ and $F \subseteq E$.

- **proper subgraph**真子图: H is a subgraph of G and $H \neq G$.
- The **subgraph induced**导出子图 by $W \subseteq V$ is (W, F), where $F = \{e: e \in E, e \subseteq W\}$.
  //Notation: $G[W]$
- The **subgraph induced**导出子图 by $F \subseteq E$ is (W, F), where $W = \{v: v \in V, v \in e \text{ for some } e \in F\}$. //Notation: $G[F]$

EXAMPLE: Let G, G', G'' be three graphs as below.

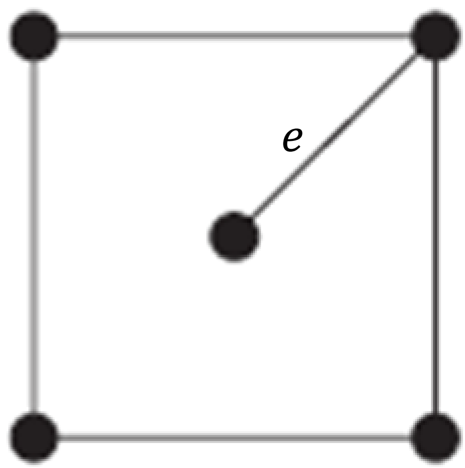


- G', G'' are subgraphs of G; G', G'' are proper subgraphs of G
- G' is a subgraph induced by $W = \{1,2,3,4\}$, i.e., $G' = G[W]$
- G'' is a subgraph induced by $F = \{\{1,2\}, \{2,3\}, \{3,4\}, \{4,1\}, \{4,5\}\}$, i.e., $G'' = G[F]$

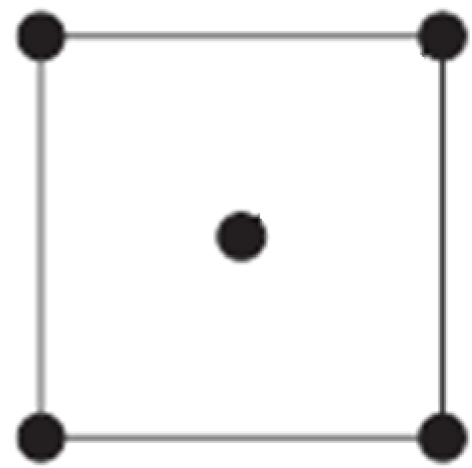
Removing An Edge

DEFINITION: Let $G = (V, E)$ be a simple graph and $e \in E$. Define

$$G - e = (V, E - \{e\})$$



$G = (V, E)$

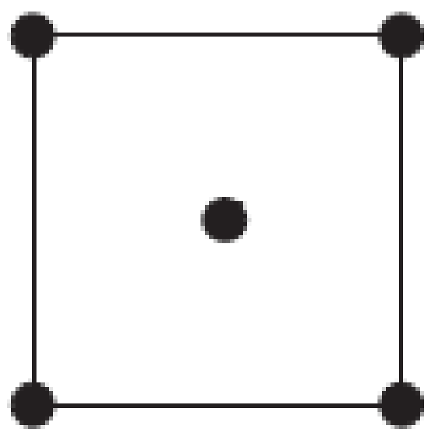


$G - e = (V, E - \{e\})$

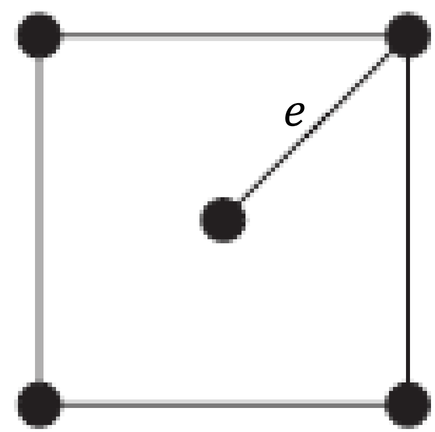
Adding An Edge

DEFINITION: Let $G = (V, E)$ be a simple graph and $e \notin E$. Define

$$G + e = (V, E \cup \{e\})$$



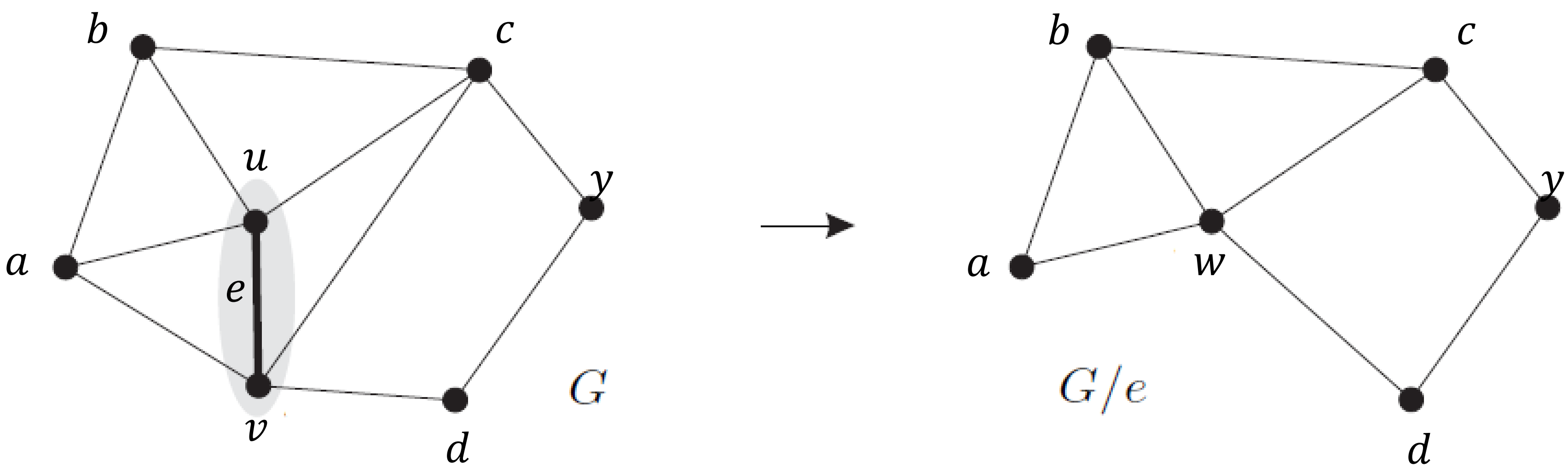
$G = (V, E)$



$G + e = (V, E \cup \{e\})$

Edge Contraction

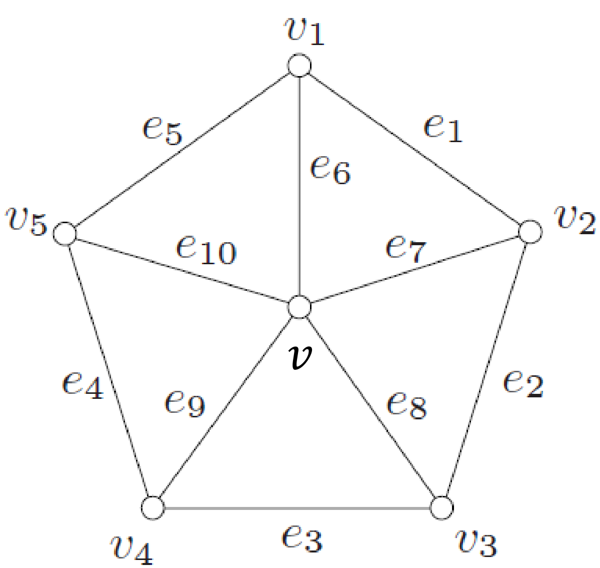
DEFINITION: Let $G = (V, E)$ be a simple graph and $e = \{u, v\} \in E$. Define $G/e = (V', E')$, where $V' = (V - \{u, v\}) \cup \{w\}$ and $E' = \{e' \in E: e' \cap e = \emptyset\} \cup \left\{\{w, x\}: \{u, x\} \in E \text{ or } \{v, x\} \in E\right\}$



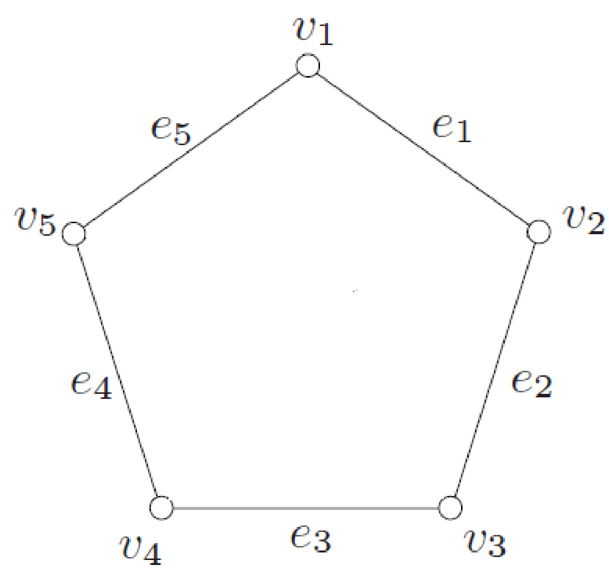
Removing A Vertex

DEFINITION: Let $G = (V, E)$ be a simple graph and let $v \in V$. Define

$$G - v = (V - \{v\}, E'), \text{ where } E' = \{e \in E : v \notin e\}$$



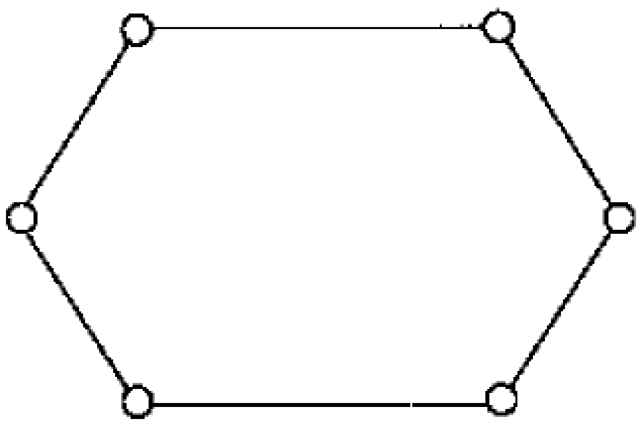
$G = (V, E)$



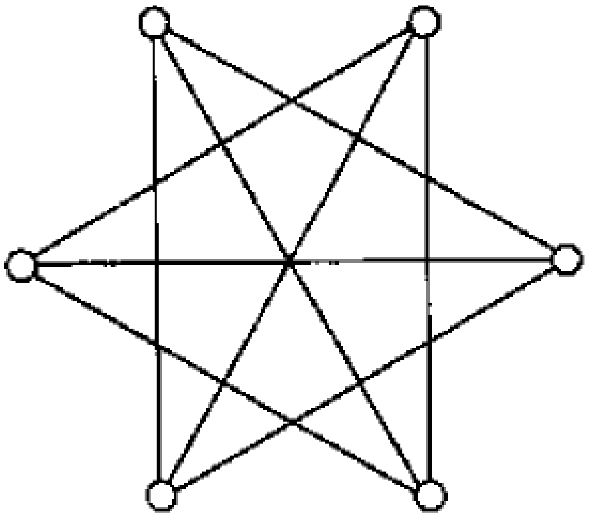
$G - v$

Complement

DEFINITION: Let $G = (V, E)$ be a simple graph of order n. Define the **complement graph**补图 of G as $\bar{G} = (V, E')$, where $E' = \{\{u, v\}: u, v \in V,\ u \neq v, \{u, v\} \notin E\}$



G

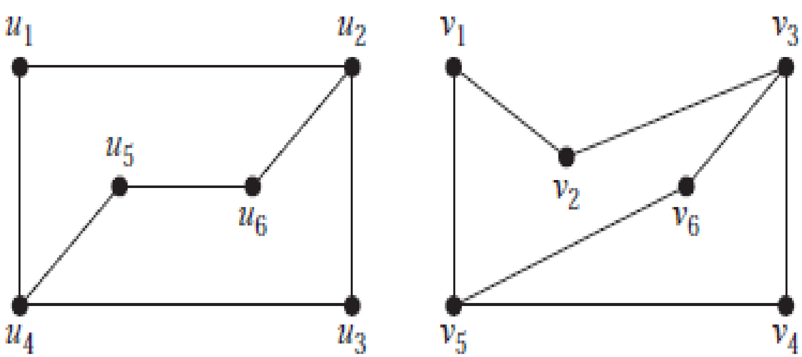


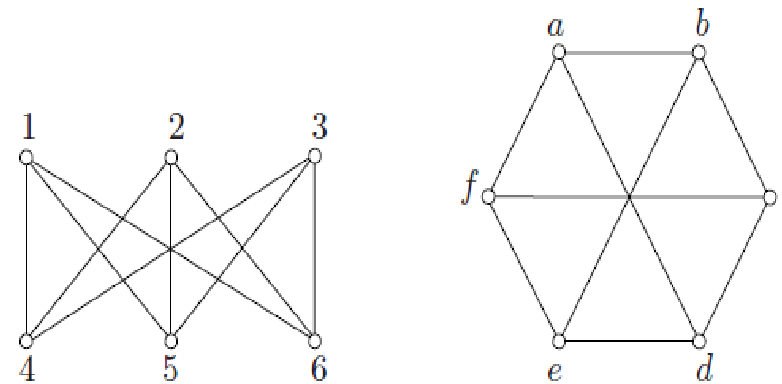
$\bar{G}$

Graph Isomorphism

DEFINITION: The simple graphs $G_1 = (V_1, E_1)$ and $G_2 = (V_2, E_2)$ are **isomorphic**同构 if there is a bijection $\sigma: V_1 \to V_2$ such that $\{u, v\} \in E_1 \Leftrightarrow \{\sigma(u), \sigma(v)\} \in E_2$.

- σ is called an **isomorphism**同构映射
- **nonisomorphic:** not isomorphic





u_1	u_2	u_3	u_4	u_5	u_6
v_6	v_3	v_4	v_5	v_1	v_2

Isomorphism σ

1	2	3	4	5	6
a	c	e	b	d	f

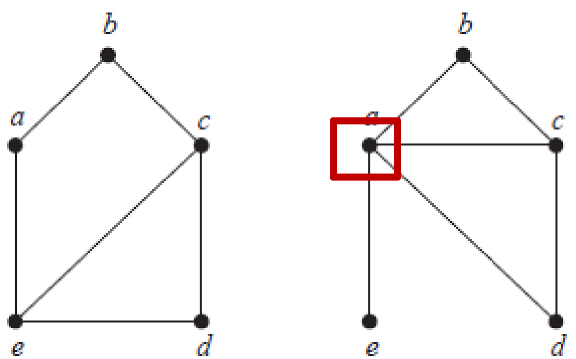
Isomorphism σ

Graph Invariants

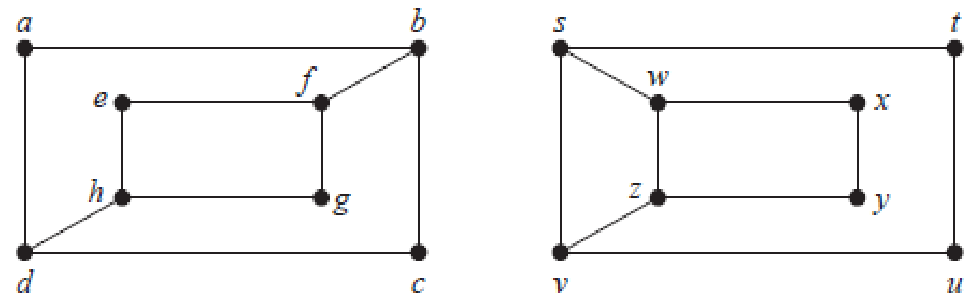
DEFINITION: **Graph invariants** are properties preserved by graph isomorphism. For example,

- The number of vertices
- The number of edges
- The number of vertices of each degree

REAMRKS: The graph invariants can be used to determine if two graphs are isomorphic or not.



There is no vertex of degree 4 in the 1st graph



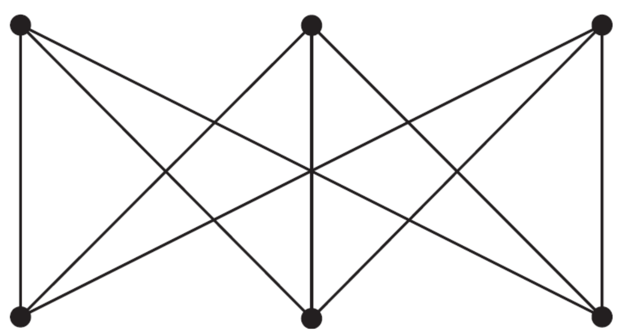
Induced subgraphs are not isomorphic. Original graphs are not isomorphic.

The subgraphs induced by the vertices of degree 3 must be isomorphic to each other.

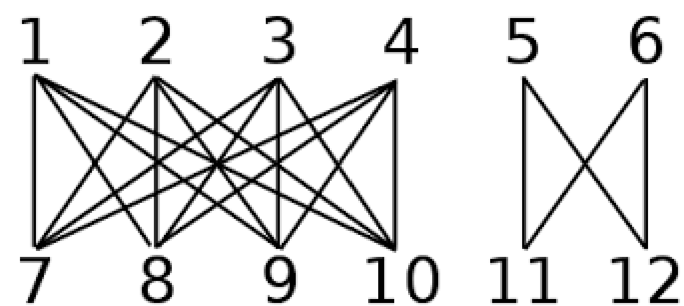
Bipartite Graph

DEFINITION: $G = (V, E)$ is a **bipartitie graph**二分图、二部图 if V has a partition $\{V_1, V_2\}$ such that $E \subseteq \{\{u_1, u_2\}: u_1 \in V_1, u_2 \in V_2\}$.

- (V_1, V_2) is a **bipartition**二划分 of the vertex set V.



A bipartite graph of order 6



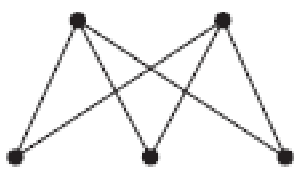
A bipartite graph of order 12

- $V_1 = \{1,2,3,4,5,6\}$
- $V_2 = \{7,8,9,10,11,12\}$

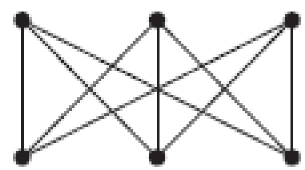
Complete Bipartite Graph

DEFINITION: A **complete bipartite graph**完全二部图 is a graph $K_{m,n} = (V, E)$ with $V = \{x_1, \dots, x_m\} \cup \{y_1, \dots, y_n\}$ and $E = \left\{\{x_i, y_j\}: i \in [m], j \in [n]\right\}$

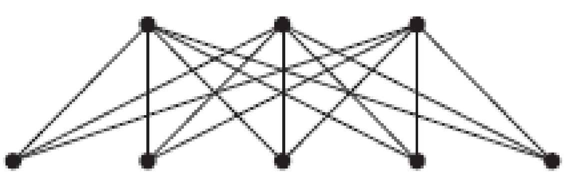
- Every vertex in V_1 is adjacent to every vertex in V_2



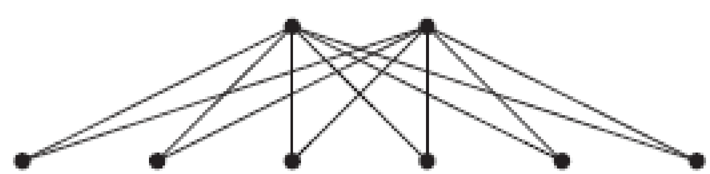
$K_{2,3}$



$K_{3,3}$



$K_{3,5}$



$K_{2,6}$

Bipartite Graph

Theorem

A simple graph is bipartite if and only if it is possible to assign one of two different colors to each vertex such that no two adjacent vertices have the same color.

Proof:

- If $G = (V, E)$ is bipartite, $V = V_1 \cup V_2$. Assign color c_1 to vertices of V_1 and color c_2 to vertices of V_2.
- Reversely, suppose we can assign colors c_1 and c_2 to the vertices such that no two adjacent have the same. Let V_i be the set of vertices of color c_i, for $i = 1, 2$. Then $V = V_1 \cup V_2$. By assumption there are no edges connecting two vertices of V_1 or two vertices of V_2, so each edge connects one vertex of V_1 with one vertex of V_2. $\square$

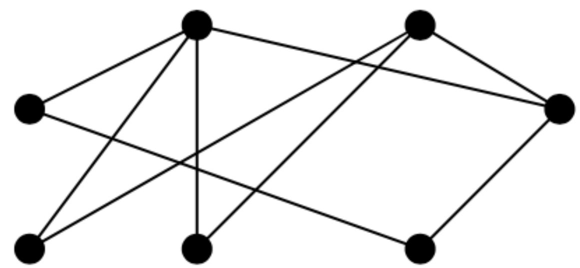
Bipartite Graph*

THEOREM: A simple graph $G = (V, E)$ is a bipartite graph iff there is a map $f: V \to \{1,2\}$ such that "$\{x, y\} \in E \Rightarrow f(x) \neq f(y)$"

- Only if: $G = (V_1 \cup V_2,\ E)$, where $V_1 \cap V_2 = \emptyset$.
  - Define $f: V \to \{1,2\}$ such that $f(x) = \begin{cases} 1 & \text{if } x \in V_1 \\ 2 & \text{if } x \in V_2 \end{cases}$
  - $\{x, y\} \in E \Rightarrow x \in V_1, y \in V_2$ or $x \in V_2, y \in V_1$
    - $f(x) \neq f(y)$
- If: $f: V \to \{1,2\}$ is a map such that "$\{x, y\} \in E \Rightarrow f(x) \neq f(y)$"
  - Let $V_1 = f^{-1}(1), V_2 = f^{-1}(2)$
    - $V = V_1 \cup V_{2,} V_1 \cap V_2 = \emptyset$
      - $\{V_1, V_2\}$ is a bipartition of V
  - $\{x, y\} \in E \Rightarrow f(x) \neq f(y) \Rightarrow x \in V_1, y \in V_2$ or $x \in V_2, y \in V_1$
    - G is a bipartite graph.

Bipartite Graph

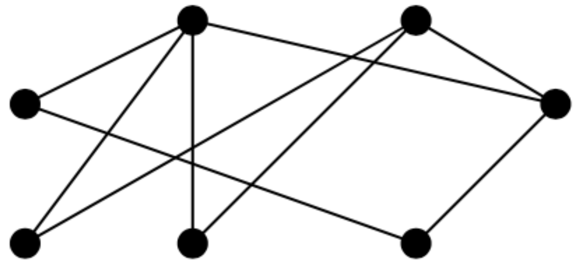
Example: Is the graph G bipartite?



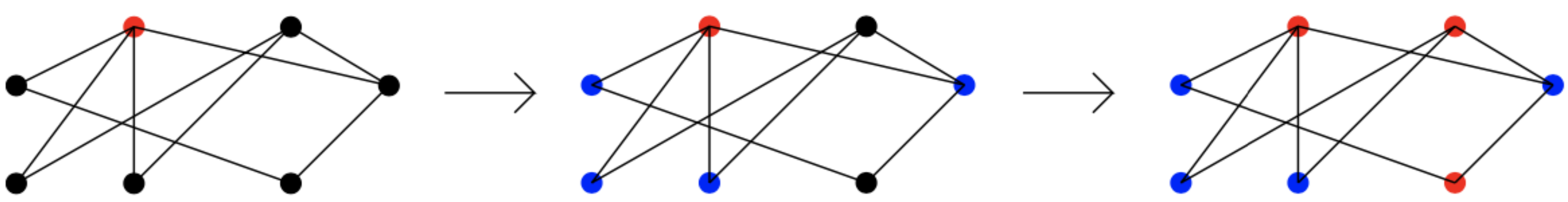
G

Bipartite Graph

Example: Is the graph G bipartite?



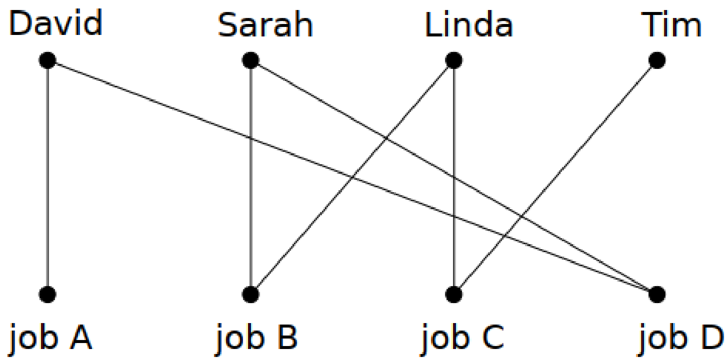
G



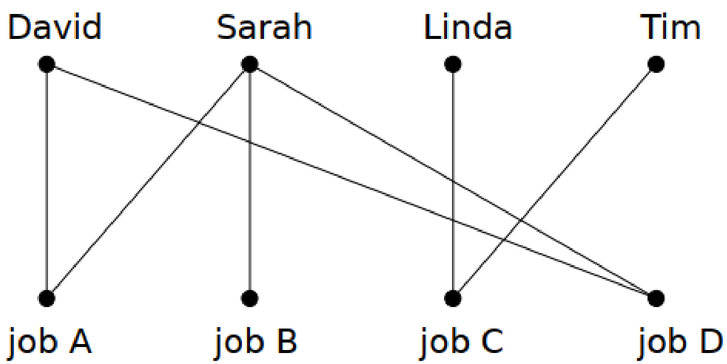
Motivation: Job Assignment

Suppose there are m employees and n different jobs to be done, with $m \geq n$.

1



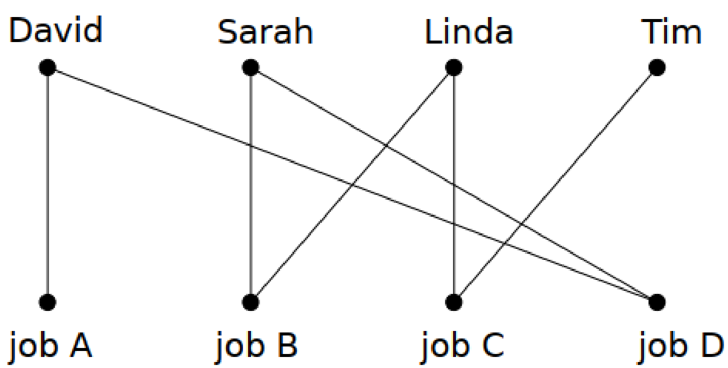
2



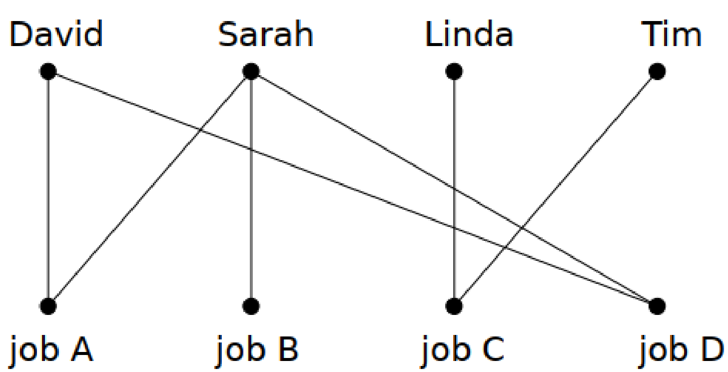
Motivation: Job Assignment

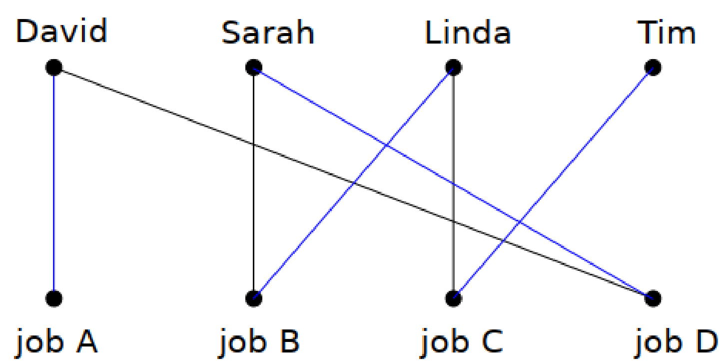
Suppose there are m employees and n different jobs to be done, with $m \geq n$.

1



2



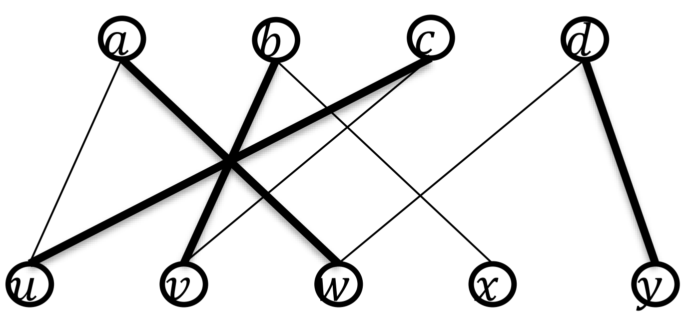


Possible solution for situation 1

Matching

DEFINITION: Let $G = (V, E)$ be a simple graph. $M \subseteq E$ is a **matching**匹配 if $e \cap e' = \emptyset$ for every $e, e' \in M$. A vertex $v \in V$ is **matched** in M if $\exists e \in M$ such that $v \in e$, otherwise, v is **not matched.**

- **maximum matching**最大匹配**:** a matching with largest number of edges.
- In a bipartite graph $G = (A \cup B, E)$, $M \subseteq E$ is a **complete matching**完全匹配 from A to B if every $u \in A$ is matched.



- $V = \{a, b, c, d, u, v, w, x, y\}$
- $V_1 = \{a, b, c, d\}$;
- $V_2 = \{u, v, w, x, y\}$
- $E = \{au, aw, bv, bx, cu, cv, dw, dy\}$

- $M = \{au, bv\}$ is a matching
  - a, b, u, v are matched in M
  - c, d, x, y are not matched in M
  - M is not a maximum matching
- $M' = \{aw, bv, cu, dy\}$ is a maximum matching
- M' is a complete matching from V_1 to V_2

Matching

DEFINITION: Let $G = (V, E)$ be a simple graph. $M \subseteq E$ is a **matching**匹配 if $e \cap e' = \emptyset$ for every $e, e' \in M$. A vertex $v \in V$ is **matched** in M if $\exists e \in M$ such that $v \in e$, otherwise, v is **not matched.**

- **maximum matching**最大匹配**:** a matching with largest number of edges.
- In a bipartite graph $G = (A \cup B, E)$, $M \subseteq E$ is a **complete matching**完全匹配 from A to B if every $u \in A$ is matched.

Example: Marriages. Suppose there are m men and n women on an island. Each person has a list of people of the opposite gender acceptable as a spouse $\Rightarrow$ bipartite graph.

- matching $\Leftrightarrow$ marriages
- maximum matching $\Leftrightarrow$ largest possible number of marriages
- complete matching from women to men $\Leftrightarrow$ marriages such that every women is married but possibly not all men.

Hall's Theorem

EXAMPLE: Marriage on an Island

- There are m boys $X = \{x_1, \dots, x_m\}$ and n girls $Y = \{y_1, \dots, y_n\}$
- $G = \left(X \cup Y, E = \left\{\{x_i, y_j\}: x_i \text{ and } y_j \text{ are willing to get married}\right\}\right)$
- What is the largest number of couples that can be formed?

THEOREM (Hall 1935): A bipartitie graph $G = (X \cup Y, E)$ has a complete matching from X to Y iff $|N(A)| \geq |A|$ for any $A \subseteq X$.

- $\Rightarrow$: Let $\{\{x_1, y_1\}, \dots, \{x_m, y_m\}\}$ be a complete matching from X to Y
  - For any $A = \{x_{i_1}, \dots, x_{i_s}\} \subseteq X$, $N(A) \supseteq \{y_{i_1}, \dots, y_{i_s}\}$
    - $|N(A)| \geq s = |A|$
- $\Leftarrow$: suppose that $|N(A)| \geq |A|$ for any $A \subseteq X$. Find a complete matching M.
  - By induction on $|X|$
  - $|X| = 1$: Let $X = \{x\}$.
    - $|N(X)| \geq 1$
      - $\exists y \in Y$ such that $e = \{x, y\} \in E$.
        - $M = \{e\}$ is a complete matching from X to Y

Hall's Theorem

- **Induction hypothesis**: "$\forall A \subseteq X, |N(A)| \geq |A| \Rightarrow \exists$ complete matching" is true when $|X| \leq k$
- Prove that "$\forall A \subseteq X, |N(A)| \geq |A| \Rightarrow \exists$ complete matching" when $|X| = k + 1$
  - Let $X = \{x_1, \dots, x_k, x_{k+1}\}$.
  - **Case 1:** $\forall\ A \subseteq X$ with $1 \leq |A| \leq k$, $|N_G(A)| \geq |A| + 1$
    - $N_G(A)$: A's neighborhood in G
    - Say $y_{k+1} \in N_G(\{x_{k+1}\})$.
    - Let $V' = (X \setminus \{x_{k+1}\}) \cup (Y \setminus \{y_{k+1}\})$; $E' = \{e \in E : e \subseteq V' \times V'\}$
    - Let $G' = (V', E') = G - \{x_{k+1}\} - \{y_{k+1}\}$.
      - $\forall\ A \subseteq \{x_1, \dots, x_k\}, |N_{G'}(A)| \geq |N_G(A)| - |\{y_{k+1}\}| \geq |A| + 1 - 1 = |A|$
        - $\exists$ a complete matching M' from $X - \{x_{k+1}\}$ to $Y - \{y_{k+1}\}$ in G' (IH)
    - $M = M' \cup \{\{x_{k+1}, y_{k+1}\}\}$ is a complete matching from X to Y in G

Hall's Theorem

- **Case 2:** $\exists A \subseteq X, 1 \leq |A| \leq k$ such that $|N_G(A)| = |A|$
  - Say $A = \{x_1, \dots, x_j\}$ and $N_G(A) = \{y_1, \dots, y_j\}$, where $1 \leq j \leq k$
  - Let $V' = A \cup N_G(A), E' = \{e \in E: e \subseteq V' \times V'\}$ and $G' = (V', E')$
    - $\forall A' \subseteq A,\ |N_{G'}(A')| = |N_G(A')| \geq |A'|$
    - There is a complete matching M' from A to $N_G(A)$ in G' (IH)
  - Let $V'' = (X \setminus A) \cup \big(Y \setminus N_G(A)\big), E'' = \{e \in E: e \subseteq V'' \times V''\}$,
  - Let $G'' = (V'', E'') = G - A - N_G(A)$
    - Then $\forall A'' \subseteq X \backslash A, |N_{G''}(A'')| \geq |A''|$.
      - Otherwise, $|N_G(A'' \cup A)| = |N_{G''}(A'')| + |N_G(A)| < |A''| + |A|$
        - $\exists$ a complete matching M'' from $X \setminus A$ to $Y \setminus N_G(A)$ (IH)
  - $M = M' \cup M''$ is a complete matching from X to Y

Discrete Mathematics: Lecture 25

Matching, path, connected, disconnected, connected component,

cut vertex, vertex cut, nonseparable, vertex connectivity, k-connected,

cut edge, edge cut, edge connectivity

Xuming He

Associate Professor

School of Information Science and Technology

ShanghaiTech University

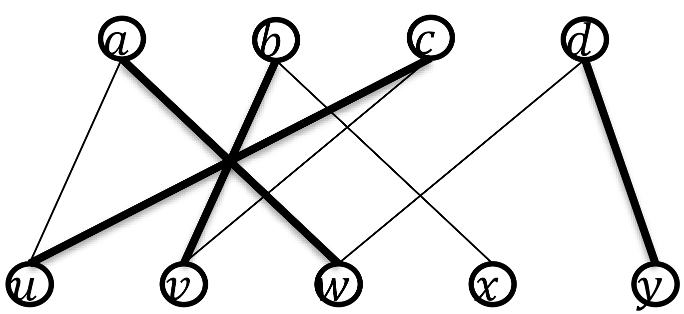
Spring Semester, 2022

Notes by Prof. Liangfeng Zhang

Matching

DEFINITION: Let $G = (V, E)$ be a simple graph. $M \subseteq E$ is a **matching**匹配 if $e \cap e' = \emptyset$ for every $e, e' \in M$. A vertex $v \in V$ is **matched** in M if $\exists e \in M$ such that $v \in e$, otherwise, v is **not matched.**

- **maximum matching**最大匹配**:** a matching with largest number of edges.
- In a bipartite graph $G = (A \cup B, E)$, $M \subseteq E$ is a **complete matching**完全匹配 from A to B if every $u \in A$ is matched.



- $M = \{au, bv\}$ is a matching
  - a, b, u, v are matched in M
  - c, d, x, y are not matched in M
  - M is not a maximum matching
- $M' = \{aw, bv, cu, dy\}$ is a maximum matching
- M' is a complete matching from V_1 to V_2

- $V = \{a, b, c, d, u, v, w, x, y\}$
- $V_1 = \{a, b, c, d\}$;
- $V_2 = \{u, v, w, x, y\}$
- $E = \{au, aw, bv, bx, cu, cv, dw, dy\}$

Hall's Theorem

EXAMPLE: Marriage on an Island

- There are m boys $X = \{x_1, \dots, x_m\}$ and n girls $Y = \{y_1, \dots, y_n\}$
- $G = \left(X \cup Y, E = \left\{\{x_i, y_j\}: x_i \text{ and } y_j \text{ are willing to get married}\right\}\right)$
- What is the largest number of couples that can be formed?

THEOREM (Hall 1935): A bipartitie graph $G = (X \cup Y, E)$ has a complete matching from X to Y iff $|N(A)| \geq |A|$ for any $A \subseteq X$.

- $\Rightarrow$: Let $\{\{x_1, y_1\}, \dots, \{x_m, y_m\}\}$ be a complete matching from X to Y
  - For any $A = \{x_{i_1}, \dots, x_{i_s}\} \subseteq X$, $N(A) \supseteq \{y_{i_1}, \dots, y_{i_s}\}$
    - $|N(A)| \geq s = |A|$
- $\Leftarrow$: suppose that $|N(A)| \geq |A|$ for any $A \subseteq X$. Find a complete matching M.
  - By induction on $|X|$
  - $|X| = 1$: Let $X = \{x\}$.
    - $|N(X)| \geq 1$
      - $\exists y \in Y$ such that $e = \{x, y\} \in E$.
        - $M = \{e\}$ is a complete matching from X to Y

Hall's Theorem

- **Induction hypothesis**: "$\forall A \subseteq X, |N(A)| \geq |A| \Rightarrow \exists$ complete matching" is true when $|X| \leq k$
- Prove that "$\forall A \subseteq X, |N(A)| \geq |A| \Rightarrow \exists$ complete matching" when $|X| = k + 1$
  - Let $X = \{x_1, \dots, x_k, x_{k+1}\}$.
  - **Case 1:** $\forall\ A \subseteq X$ with $1 \leq |A| \leq k$, $|N_G(A)| \geq |A| + 1$
    - $N_G(A)$: A's neighborhood in G
    - Say $y_{k+1} \in N_G(\{x_{k+1}\})$.
    - Let $V' = (X \setminus \{x_{k+1}\}) \cup (Y \setminus \{y_{k+1}\})$; $E' = \{e \in E: e \subseteq V' \times V'\}$
    - Let $G' = (V', E') = G - \{x_{k+1}\} - \{y_{k+1}\}$.
      - $\forall\ A \subseteq \{x_1, \dots, x_k\}, |N_{G'}(A)| \geq |N_G(A)| - |\{y_{k+1}\}| \geq |A| + 1 - 1 = |A|$
        - $\exists$ a complete matching M' from $X - \{x_{k+1}\}$ to $Y - \{y_{k+1}\}$ in G' (IH)
    - $M = M' \cup \left\{\{x_{k+1}, y_{k+1}\}\right\}$ is a complete matching from X to Y in G

Hall's Theorem

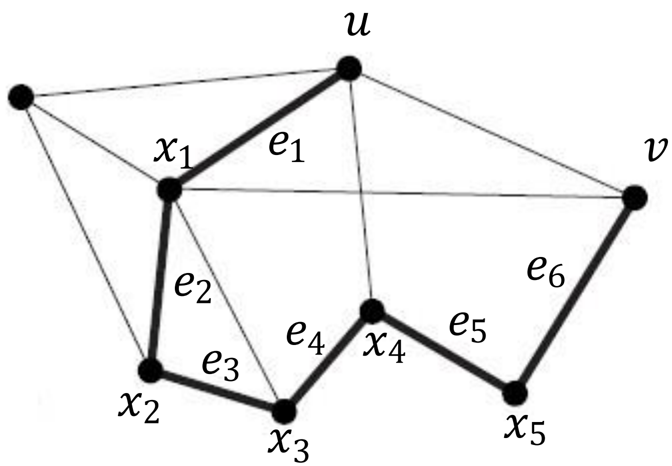
- **Case 2:** $\exists A \subseteq X, 1 \leq |A| \leq k$ such that $|N_G(A)| = |A|$
  - Say $A = \{x_1, \dots, x_j\}$ and $N_G(A) = \{y_1, \dots, y_j\}$, where $1 \leq j \leq k$
  - Let $V' = A \cup N_G(A), E' = \{e \in E: e \subseteq V' \times V'\}$ and $G' = (V', E')$
    - $\forall A' \subseteq A,\ |N_{G'}(A')| = |N_G(A')| \geq |A'|$
    - There is a complete matching M' from A to $N_G(A)$ in G' (IH)
  - Let $V'' = (X \setminus A) \cup \left(Y \setminus N_G(A)\right), E'' = \{e \in E: e \subseteq V'' \times V''\}$,
  - Let $G'' = (V'', E'') = G - A - N_G(A)$
    - Then $\forall\ A'' \subseteq X \setminus A, |N_{G''}(A'')| \geq |A''|$.
      - Otherwise, $|N_G(A'' \cup A)| = |N_{G''}(A'')| + |N_G(A)| < |A''| + |A|$
        - $\exists$ a complete matching M'' from $X \setminus A$ to $Y \setminus N_G(A)$ (IH)
  - $M = M' \cup M''$ is a complete matching from X to Y

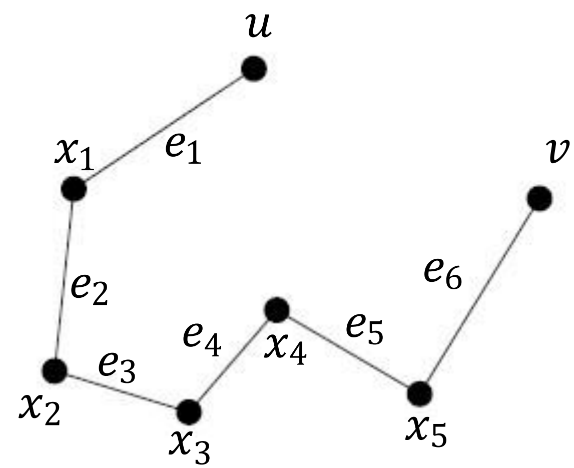
Path (Undirected)

DEFINITION: Let $G = (V, E)$ be an undirected graph and let $k \in \mathbb{N}$. A **path**路径 **of length** $\boldsymbol{k}$ from u to v in G is a sequence of k edges $e_1, \ldots, e_k$ of G for which there exist vertices $x_0 = u, x_1, \ldots, x_{k-1}, x_k = v$ such that $e_i = \{x_{i-1}, x_i\}$ for every $i \in [k]$.

- The path is **circuit**回路 if $u = v$ and $k > 0$
- The path **passes through**经过 $x_1, \ldots, x_{k-1}$
- The path **traverses**遍历 $e_1, e_2, \ldots, e_k$
- The path is **simple**简单 if it doesn't contain an edge more than once.
- If G is simple, the path can be denoted as $x_0, x_1, \ldots, x_k$

Example





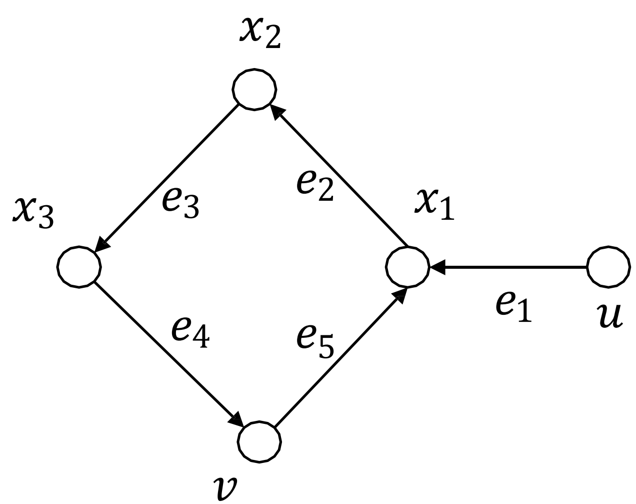
- The right-hand side graph is a path from u to v
- The path is $e_1, e_2, e_3, e_4, e_5, e_6$
- The path is simple
- The path can be denoted by $u, x_1, x_2, x_3, x_4, x_5, v$
- The path passes through x_1, x_2, x_3, x_4, x_5
- The path traverses $e_1, e_2, e_3, e_4, e_5, e_6$
- $e_1, e_2, e_3, e_4, e_5, e_6, e_7 = \{v, u\}$ is a (simple) circuit

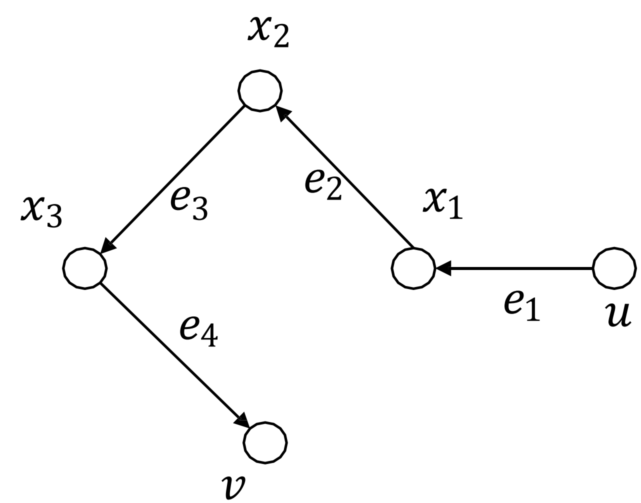
Path (Directed)

DEFINITION: Let $G = (V, E)$ be a directed graph and let $k \in \mathbb{N}$. A **path of length $\boldsymbol{k}$** from u to v in G is a sequence of k edges $e_1, \dots, e_k$ of G for which there exist vertices $x_0 = u, x_1, \dots, x_{k-1}, x_k = v$ such that $e_i = (x_{i-1}, x_i)$ for every $i \in [k]$.

- The path is a **circuit** if $u = v$ and $k > 0$
- The path **passes through** $x_1, \dots, x_{k-1}$
- The path **traverses** $e_1, e_2, \dots, e_k$
- The path is **simple** if it doesn't contain an edge more than once.
- If G has no multiple edges, the path can be denoted as $x_0, \dots, x_k$

Example



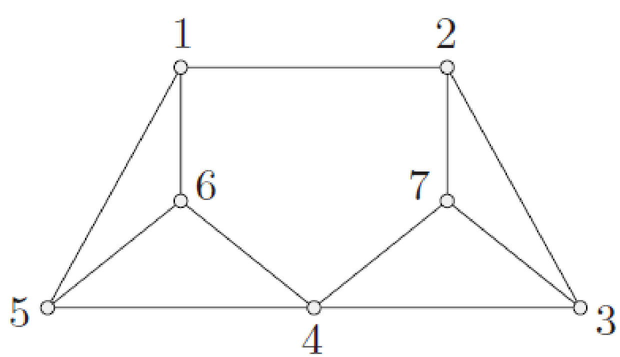


- e_1, e_2, e_3, e_4 is a path
- The path is simple
- The path can be denoted by u, x_1, x_2, x_3, v
- The path passes through x_1, x_2, x_3
- The path traverses e_1, e_2, e_3, e_4
- e_2, e_3, e_4, e_5 is a (simple) circuit

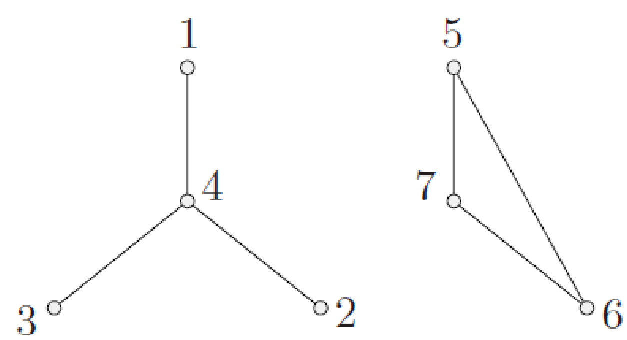
Connectivity

DEFINITION: An undirected graph G is said to be **connected**连通的 if there is a path between any pair of distinct vertices.

- Graph of order 1 is connected; the complete graph K_n is connected
- **disconnected**非连通的: not connected
- **disconnect** G**:** remove vertices or edges to produce a disconnected subgraph



A Connected Graph



A Disconnected Graph

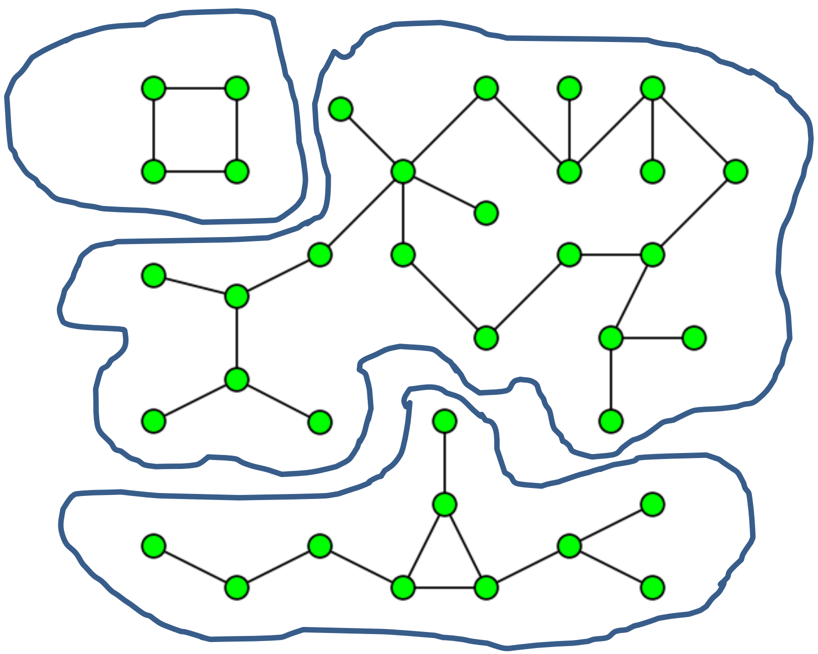
Connectivity

THEOREM: Let $G = (V, E)$ be a connected undirected graph. Then there is a simple path between any pair of distinct vertices.

- Let $u, v \in V$ and $u \neq v$. Find a simple path from u to v.
- G is connected $\Rightarrow$ there are paths from u to v.
  - Let $x_0 = u, x_1, \dots, x_{k-1}, x_k = v$ be one that has least length k.
    - This path must be simple.
      - otherwise, the path contains some edge more than once
        - $\exists i, j \in \{0,1, \dots, k\}$, say $i < j$, such that $x_i = x_j$
          - $x_0, x_1, \dots, x_{i-1}, x_j, \dots, x_k$ is a shorter path from u to v
      - The contradiction shows that the path must be simple

Connected Component

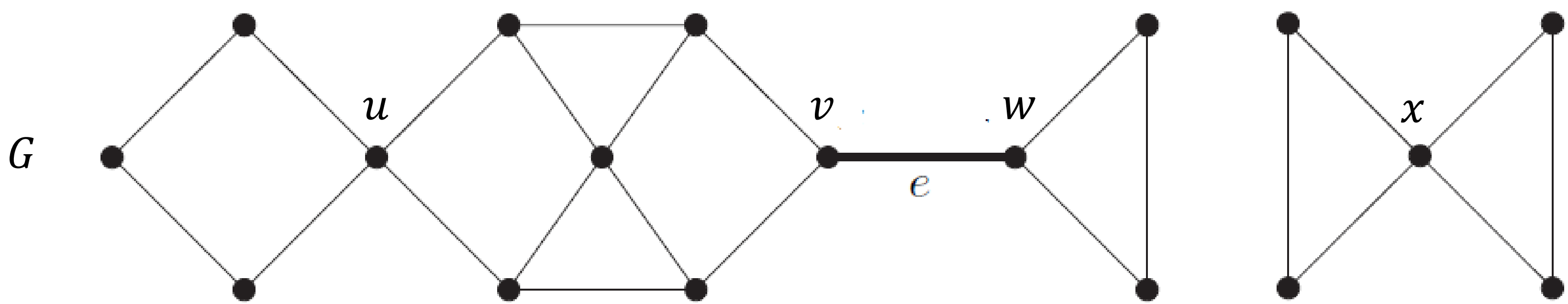
DEFINITION: A **connected component**连通分支 of a graph $G = (V, E)$ is a connected subgraph of G that is not a proper subgraph of a connected subgraph of G. //i.e., maximal极大 connected subgraph



Connected Component

DEFINITION: A **connected component**连通分支 of a graph $G = (V, E)$ is a connected subgraph of G that is not a proper subgraph of a connected subgraph of G. //i.e., maximal极大 connected subgraph

- $v \in V$ is a **cut vertex**割点 if $G - v$ has more connected components than G
- $e \in E$ is a **cut edge**割边、**bridge**桥 if $G - e$ has more connected components than G



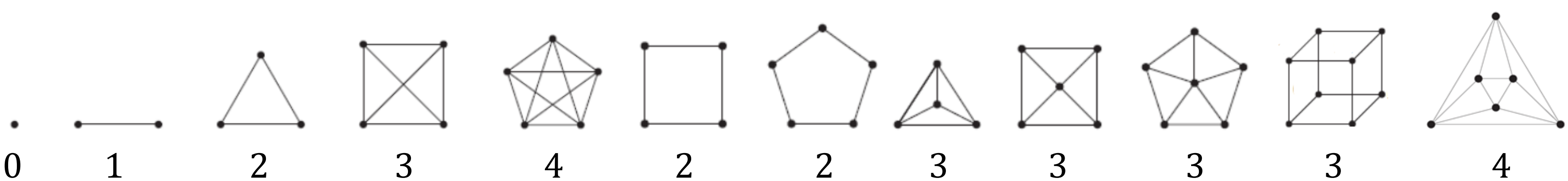
- There are 2 connected components in the graph G
- cut vertices: u, v, w, x
- cut edge: e

Vertex Connectivity

DEFINITION: A connected undirected graph $G = (V, E)$ is said to be **nonseparable**不可分的 if G has no cut vertex.

DEFINITION: Let $G = (V, E)$ be a connected simple graph.

- **vertex cut**点割集: A subset $V' \subseteq V$ such that $G - V'$ is disconnected
- **vertex connectivity**点连通度 $\kappa(G)$: the minimum number of vertices whose removal disconnect G or results in K_1; equivalently,
  - if G is disconnected, $\kappa(G) = 0$; //additional definition
  - if $G = K_n, \kappa(G) = n - 1$ //K_n has no vertex cut
  - else, $\kappa(G)$ is the minimum size of a vertex cut of G



These graphs are all nonseparable

Vertex Connectivity

THEOREM: Let $G = (V, E)$ be a simple graph of order n. Then

- $0 \leq \kappa(G) \leq n - 1$
  - Removing $n - 1$ vertices gives K_1
    - $\kappa(G) \leq n - 1$
- $\kappa(G) = 0$ iff G is disconnected or $G = K_1$
  - trivial
- $\kappa(G) = n - 1$ iff $G = K_n (n \geq 2)$
  - If: obvious
  - Only if:
    - $n = 2$: $\kappa(G) = 1 \Rightarrow G = K_2$
    - $n \geq 3$: Prove by contradiction. Suppose that $G \neq K_n$.
      - There exist distinct $u, v \in V$ such that $u \neq v$ and $\{u, v\} \notin E$
        - Let $X = V - \{u, v\}$. Then $G - X$ is disconnected.
          - $\kappa(G) \leq |X| = n - 2 < n - 1.$
            - This contradicts the condition $\kappa(G) = n - 1.$

Vertex Connectivity

DEFINITION: A simple graph $G = (V, E)$ is called $\boldsymbol{k}$**-connected**$_{k连通的}$ ($\boldsymbol{k}$**-vertex-connected**) $_{k点连通的}$ if $\kappa(G) \geq k$.

THEOREM: Let $G = (V, E)$ be a simple graph of order n. Then

- G is 1-connected iff G is connected and $G \neq K_1$.
  - **Only if**: G disconnected or $G = K_1 \Rightarrow \kappa(G) = 0$
  - **If** : $G \neq K_1 \Rightarrow n \geq 2$; G is connected $\Rightarrow$ removing 0 vertex cannot disconnect G or give $K_1 \Rightarrow \kappa(G) \geq 1$
- G is 2-connected iff G is nonseparable and $n \geq 3$.
  - **Only if**: $n \leq 2 \Rightarrow \kappa(G) \leq 1$; G not nonseparable $\Rightarrow$ G has cut vertex $\Rightarrow \kappa(G) \leq 1$.
  - **If**: $n \geq 3 \Rightarrow$ removing ≤ 1 vertex cannot result in K_1; G nonseparable$\Rightarrow$ removing ≤ 1 vertex cannot disconnect G; Hence. $\kappa(G) \geq 2$.
- G is k-connected iff G is j-connected for all $j \in \{0,1, \dots, k\}$
  - **Only if**: $\kappa(G) \geq k \Rightarrow \kappa(G) \geq j$ for all $j \in \{0,1, \dots, k\} \Rightarrow G$ is j connected
  - **If**: G is obviously k-connected

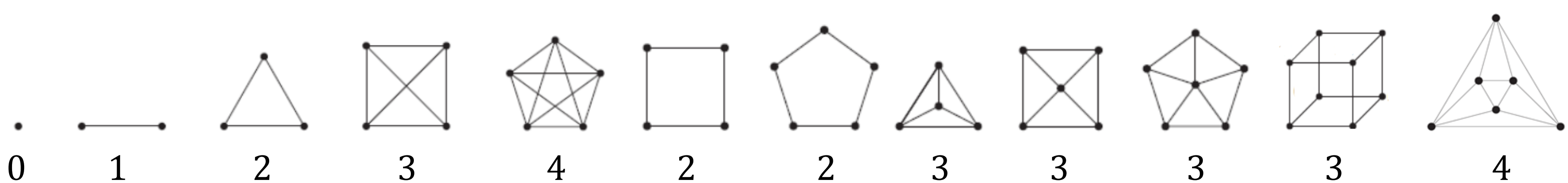
Edge Connectivity

DEFINITION: Let $G = (V, E)$ be a connected simple graph. $E' \subseteq E$ is an **edge cut**边割集 of G if $G - E'$ is disconnected.

DEFINITION: Let $G = (V, E)$ be a simple graph.

The **edge connectivity**边连通度 $(\lambda(G))$ of G is defined as below:

- G disconnected: $\lambda(G) = 0$
- G connected:
  - $|V| = 1$: $\lambda(G) = 0$
  - $|V| > 1$: $\lambda(G)$ is the minimum size of edge cuts of G.



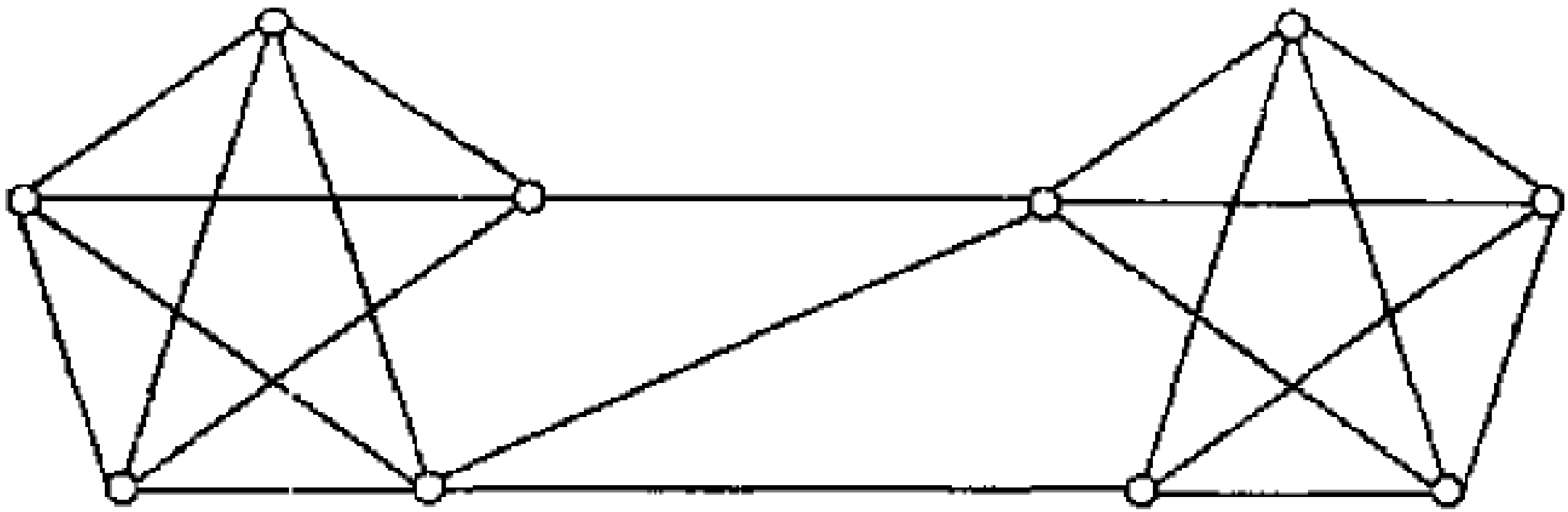
Edge Connectivity

THEOREM: Let $G = (V, E)$ be a simple graph of order n. Then

- $0 \leq \lambda(G) \leq n - 1$
  - $n = 1$: $G = K_1$ and $\lambda(G) = 0$
  - $n > 1$: $\deg(u) \leq n - 1$ for every $u \in V$
    - By removing $\{\{u, x\}: \{u, x\} \in E\}$, we can disconnect G.
      - Hence, $\lambda(G) \leq n - 1$.
- $\lambda(G) = 0$ iff G is disconnected or $G = K_1$
  - Only if: $n > 1$ and G connected $\Rightarrow \lambda(G) \geq 1$;
  - If: definition
- $\lambda(G) = n - 1$ iff $G = K_n$ $(n \geq 2)$
  - Only if: if $G \neq K_n$, then $\deg(u) < n - 1$ for some $u \in V$.
    - Remove $\big\{\{u, x\}: \{u, x\} \in E\big\}$. Then G is disconnected. $\lambda(G) < n - 1$
  - If: $\lambda(K_n) \geq \kappa(K_n) = n - 1$. (see the next theorem)

Connectivity

THEOREM: Let $G = (V, E)$ be a simple graph. Then $\kappa(G) \leq \lambda(G) \leq \delta(G)$, where $\delta(G) = \min_{v \in V} \deg(v)$ is the least degree of $G's$ vertices.



- $\kappa(G) = 2$
- $\lambda(G) = 3$
- $\delta(G) = 4$

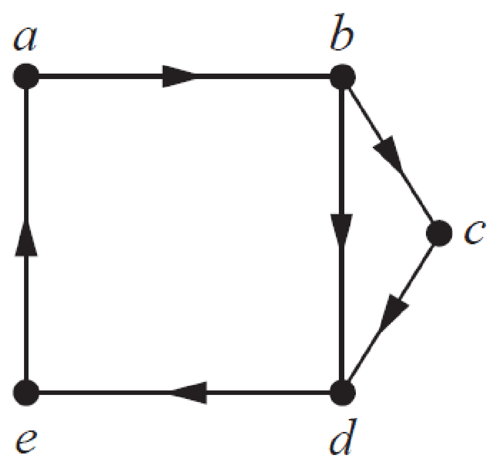
https://cp-algorithms.com/graph/edge_vertex_connectivity.html

http://www.math.caltech.edu/~2014-15/2term/ma006b/05%20connectivity%201.pdf

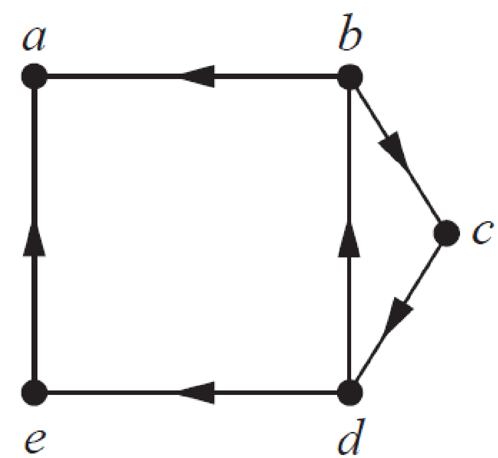
Connected Directed Graphs

DEFINITION: Let $G = (V, E)$ be a directed graph. G is said to be **strongly connected** if there is a path from u to v and a path from v to u for all $u, v \in V$ $(u \neq v)$.

- **weakly connected***:* the graph is connected if we remove the directions of all direct edges.



Strongly connected

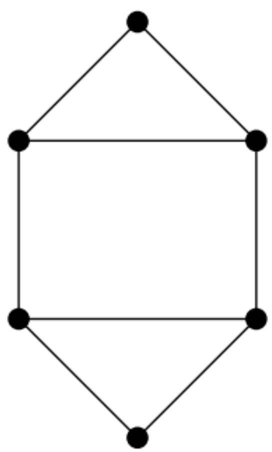


Weakly connected

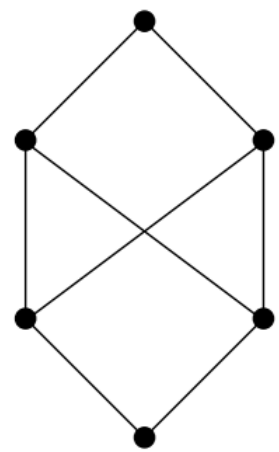
Paths and Isomorphism

Theorem

The existence of a simple circuit of length k, $k \geq 3$ is an isomorphism invariant for simple graphs.



G_1



G_2

6 vertices, 8 edges

Degree sequence: 3, 3, 3, 3, 2, 2

Paths and Isomorphism*

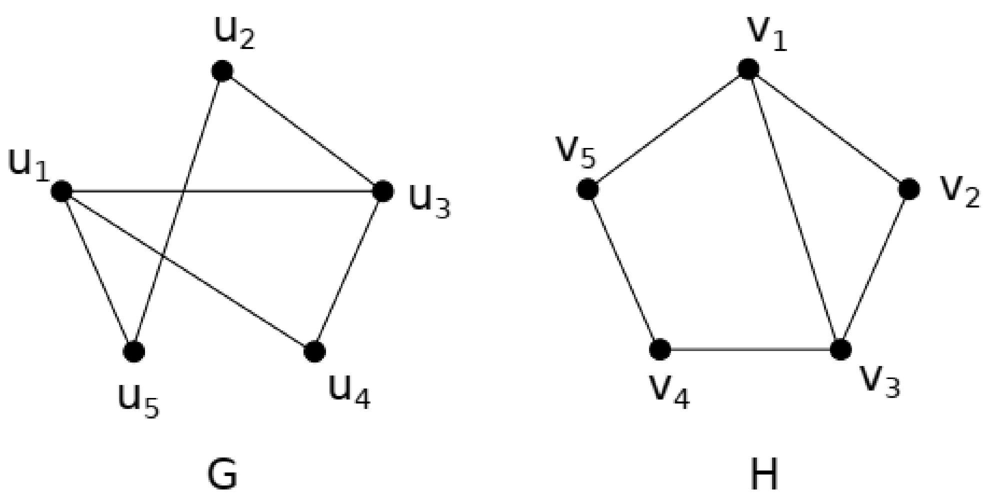
Theorem

The existence of a simple circuit of length k, $k \geq 3$ is an isomorphism invariant for simple graphs.

Proof: Let $G_1 = (V_1, E_1)$ and $G_2 = (V_2, E_2)$ be isomorphic graphs: there is a bijective function $f : V_1 \to V_2$ respecting adjacency conditions.
Assume G_1 has a simple circuit of length k: $u_0, u_1, \ldots, u_k = u_0$, with $u_i \in V_1$ for $0 \leq i \leq k$. Let's denote $v_i = f(u_i)$, for $0 \leq i \leq k$.

$$(u_i, u_{i+1}) \in E_1 \Rightarrow (f(u_i), f(u_{i+1})) = (v_i, v_{i+1}) \in E_2, \text{ for } 0 \leq i \leq k-1.$$

So $v_0, \ldots, v_k$ is a path of length k in G_2.
It is a circuit because $v_k = f(u_k) = f(u_0) = v_0$.
It is simple: if not, at least one edge is traversed more than once, so it would mean that there exist $0 \leq i \neq j \leq k-1$ such that $(v_i, v_{i+1}) = (v_j, v_{j+1})$. But this implies $(u_i, u_{i+1}) = (u_j, u_{j+1})$ by bijectivity of f. This is impossible because $u_0, u_1, \ldots, u_k$ is simple.



5 vertices, 6 edges
Degree sequence: 3, 3, 2, 2, 2
1 simple circuit of length 3,
1 simple circuit of length 4,
1 simple circuit of length 5.

Isomorphic graphs ?

If there is an iso $f : V_G \to V_H$, the simple circuit of length 5 u_1, u_4, u_3, u_2, u_5 must be sent to the simple circuit of length 5 in H, respecting the degrees of vertices.
Check that $f(u_1) = v_1, f(u_4) = v_2, f(u_3) = v_3, f(u_2) = v_4, f(u_5) = v_5$ is an isomorphism by writing adjacency matrices.

Counting Paths Between Vertices

Theorem

Let G be a graph with adjacency matrix A with respect to the ordering of vertices $v_1, \ldots, v_n$. The number of different paths of length $r \geq 1$ from v_i to v_j equals the (i, j) entry of the matrix A^r.

Proof: By induction

• $r = 1$: the number of paths of length 1 from v_i to v_j is equal to the (i, j) entry of A by definition of A, as it corresponds to the number of edges from v_i to v_j.

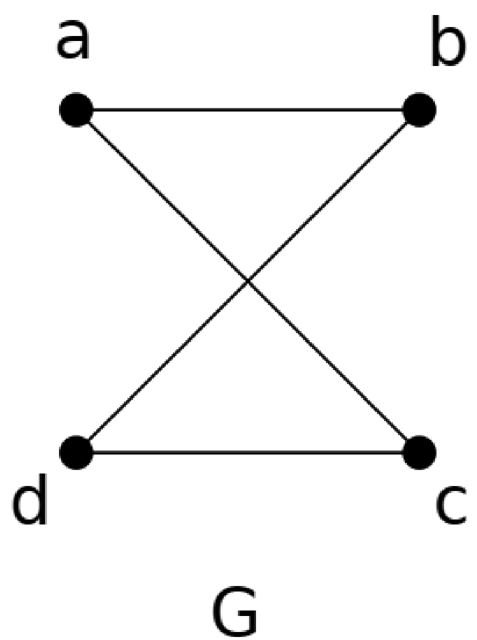
• Assume the (i, j) entry of the matrix A^r is the number of different paths of length r from v_i to v_j.
We can write $A^{r+1} = A^r A$
Let's denote $A^r = (b_{ij})_{1 \leq i,j \leq n}$, and $A = (a_{ij})_{1 \leq i,j \leq n}$. The (i, j) entry of A^{r+1} is given by:

$$\sum_{k=1}^{n} b_{ik} a_{kj} = b_{i1} a_{1j} + b_{i2} a_{2j} + \cdots + b_{in} a_{nj} \tag{1}$$

By hypothesis: b_{ik} equals the number of paths of length r from v_i to v_k.
"Path of length $r+1$ from v_i to v_j = path of length r from v_i to any vertex v_k + an edge from v_k to v_j."
This is equal to the sum (1).

Example

How many paths of length four are there from a to d in the simple graph G



with ordering of vertices (a, b, c, d, e):

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

$$A_G^2 = \begin{pmatrix} 2 & 0 & 0 & 2 \\ 0 & 2 & 2 & 0 \\ 0 & 2 & 2 & 0 \\ 2 & 0 & 0 & 2 \end{pmatrix} \quad A_G^3 = \begin{pmatrix} 0 & 4 & 4 & 0 \\ 4 & 0 & 0 & 4 \\ 4 & 0 & 0 & 4 \\ 0 & 4 & 4 & 0 \end{pmatrix} \quad A_G^4 = \begin{pmatrix} 8 & 0 & 0 & 8 \\ 0 & 8 & 8 & 0 \\ 0 & 8 & 8 & 0 \\ 8 & 0 & 0 & 8 \end{pmatrix}$$

Discrete Mathematics: Lecture 26

Paths and Isomorphism, Counting Paths, Euler Paths and Circuits

Xuming He
Associate Professor

School of Information Science and Technology
ShanghaiTech University

Spring Semester, 2022

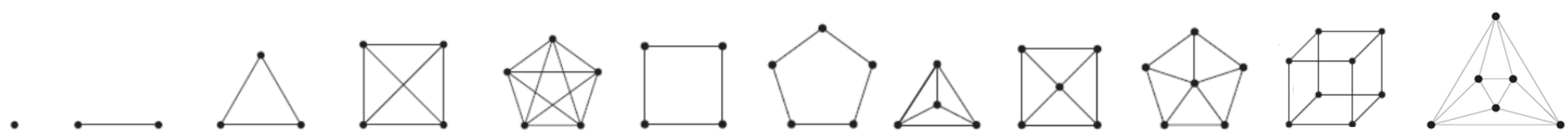
Notes by Prof. Liangfeng Zhang

Vertex Connectivity

DEFINITION: A connected undirected graph $G = (V, E)$ is said to be **nonseparable**不可分的 if G has no cut vertex.

DEFINITION: Let $G = (V, E)$ be a connected simple graph.

- **vertex cut**点割集: A subset $V' \subseteq V$ such that $G - V'$ is disconnected
- **vertex connectivity**点连通度 $\kappa(G)$: the minimum number of vertices whose removal disconnect G or results in K_1; equivalently,
  - if G is disconnected, $\kappa(G) = 0$; //additional definition
  - if $G = K_n, \kappa(G) = n - 1$ //K_n has no vertex cut
  - else, $\kappa(G)$ is the minimum size of a vertex cut of G



0 1 2 3 4 2 2 3 3 3 3 4

These graphs are all nonseparable

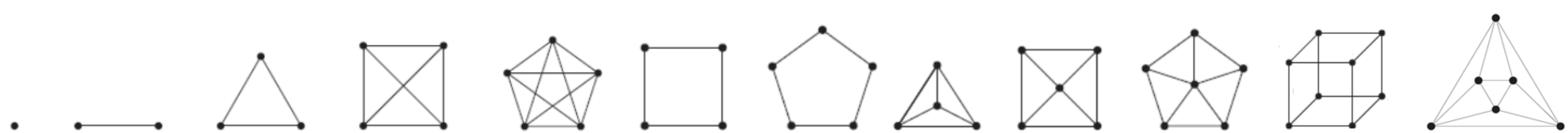
Edge Connectivity

DEFINITION: Let $G = (V, E)$ be a connected simple graph. $E' \subseteq E$ is an **edge cut**边割集 of G if $G - E'$ is disconnected.

DEFINITION: Let $G = (V, E)$ be a simple graph.

The **edge connectivity**边连通度 *(*$\lambda(G)$*)* of G is defined as below:

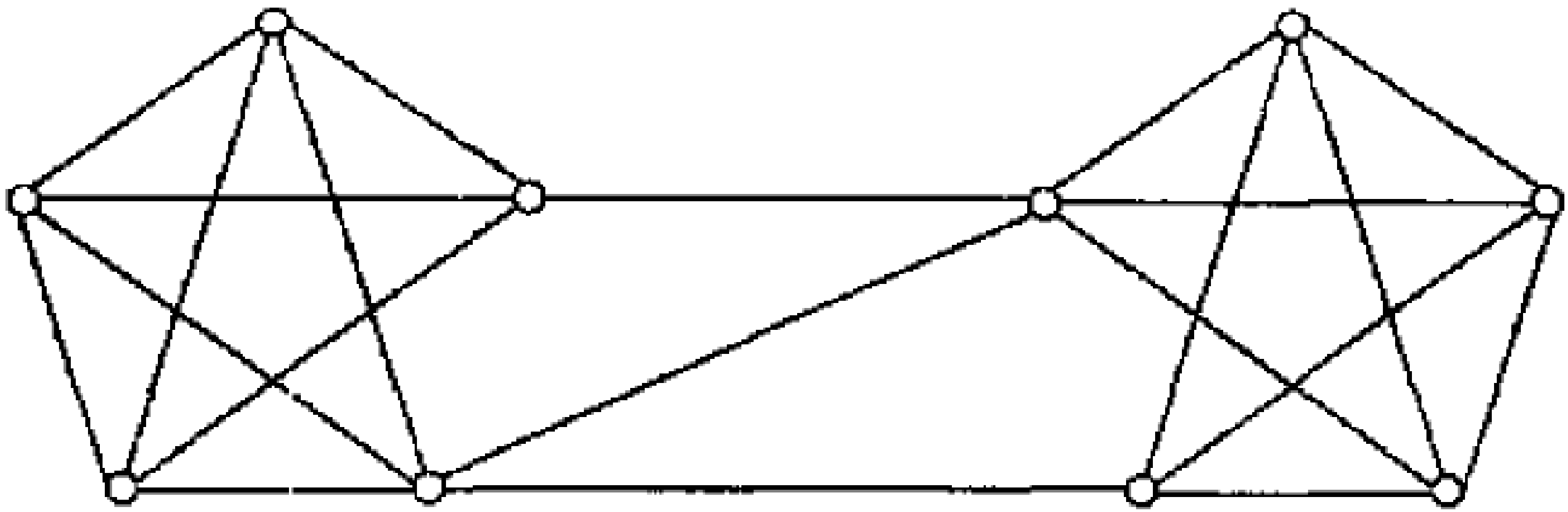
- G disconnected*:* $\lambda(G) = 0$
- G connected*:*
    - $|V| = 1$: $\lambda(G) = 0$
    - $|V| > 1$: $\lambda(G)$ is the minimum size of edge cuts of G.



0 1 2 3 4 2 2 3 3 3 3 4

Connectivity

THEOREM: Let $G = (V, E)$ be a simple graph. Then $\kappa(G) \leq \lambda(G) \leq \delta(G)$, where $\delta(G) = \min_{v \in V} \deg(v)$ is the least degree of $G's$ vertices.



- $\kappa(G) = 2$
- $\lambda(G) = 3$
- $\delta(G) = 4$

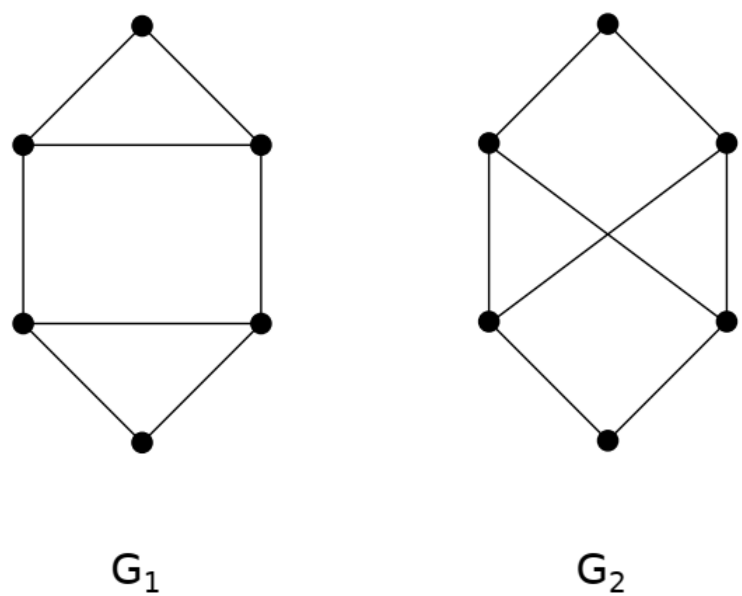
https://cp-algorithms.com/graph/edge_vertex_connectivity.html

http://www.math.caltech.edu/~2014-15/2term/ma006b/05%20connectivity%201.pdf

Paths and Isomorphism

Theorem

The existence of a simple circuit of length k, $k \geq 3$ is an isomorphism invariant for simple graphs.



6 vertices, 8 edges

Degree sequence: 3, 3, 3, 3, 2, 2

Paths and Isomorphism*

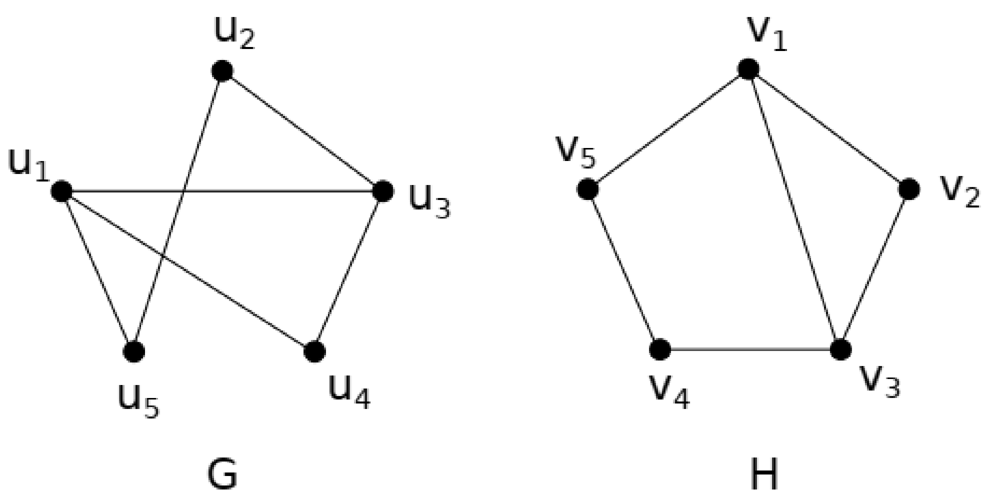
Theorem

The existence of a simple circuit of length k, $k \geq 3$ is an isomorphism invariant for simple graphs.

Proof: Let $G_1 = (V_1, E_1)$ and $G_2 = (V_2, E_2)$ be isomorphic graphs: there is a bijective function $f : V_1 \to V_2$ respecting adjacency conditions.
Assume G_1 has a simple circuit of length k: $u_0, u_1, \ldots, u_k = u_0$, with $u_i \in V_1$ for $0 \leq i \leq k$. Let's denote $v_i = f(u_i)$, for $0 \leq i \leq k$.

$$(u_i, u_{i+1}) \in E_1 \Rightarrow (f(u_i), f(u_{i+1})) = (v_i, v_{i+1}) \in E_2, \text{ for } 0 \leq i \leq k-1.$$

So $v_0, \ldots, v_k$ is a path of length k in G_2.
It is a circuit because $v_k = f(u_k) = f(u_0) = v_0$.
It is simple: if not, at least one edge is traversed more than once, so it would mean that there exist $0 \leq i \neq j \leq k-1$ such that $(v_i, v_{i+1}) = (v_j, v_{j+1})$. But this implies $(u_i, u_{i+1}) = (u_j, u_{j+1})$ by bijectivity of f. This is impossible because $u_0, u_1, \ldots, u_k$ is simple.



5 vertices, 6 edges
Degree sequence: 3, 3, 2, 2, 2
1 simple circuit of length 3,
1 simple circuit of length 4,
1 simple circuit of length 5.

Isomorphic graphs ?

If there is an iso $f : V_G \to V_H$, the simple circuit of length 5 u_1, u_4, u_3, u_2, u_5 must be sent to the simple circuit of length 5 in H, respecting the degrees of vertices.
Check that $f(u_1) = v_1, f(u_4) = v_2, f(u_3) = v_3, f(u_2) = v_4, f(u_5) = v_5$ is an isomorphism by writing adjacency matrices.

Counting Paths Between Vertices

Theorem

Let G be a graph with adjacency matrix A with respect to the ordering of vertices $v_1, \ldots, v_n$. The number of different paths of length $r \geq 1$ from v_i to v_j equals the (i, j) entry of the matrix A^r.

Proof: By induction

• $r = 1$: the number of paths of length 1 from v_i to v_j is equal to the (i, j) entry of A by definition of A, as it corresponds to the number of edges from v_i to v_j.

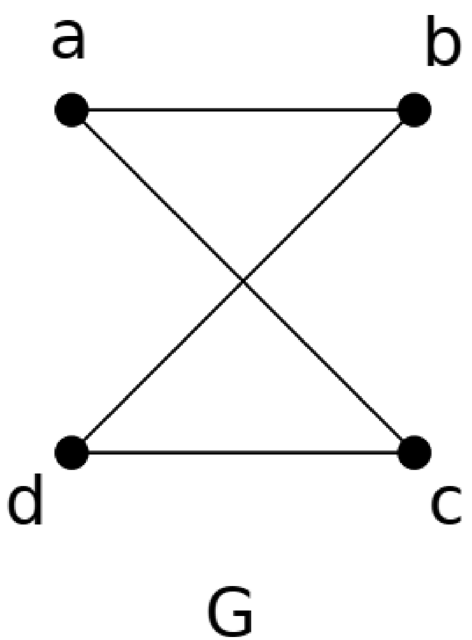
• Assume the (i, j) entry of the matrix A^r is the number of different paths of length r from v_i to v_j.
We can write $A^{r+1} = A^r A$
Let's denote $A^r = (b_{ij})_{1 \le i,j \le n}$, and $A = (a_{ij})_{1 \le i,j \le n}$. The (i, j) entry of A^{r+1} is given by:

$$\sum_{k=1}^{n} b_{ik} a_{kj} = b_{i1} a_{1j} + b_{i2} a_{2j} + \cdots + b_{in} a_{nj} \tag{1}$$

By hypothesis: b_{ik} equals the number of paths of length r from v_i to v_k.
"Path of length $r+1$ from v_i to v_j = path of length r from v_i to any vertex v_k + an edge from v_k to v_j."
This is equal to the sum (1).

Example

How many paths of length four are there from a to d in the simple graph G



with ordering of vertices (a, b, c, d, e):

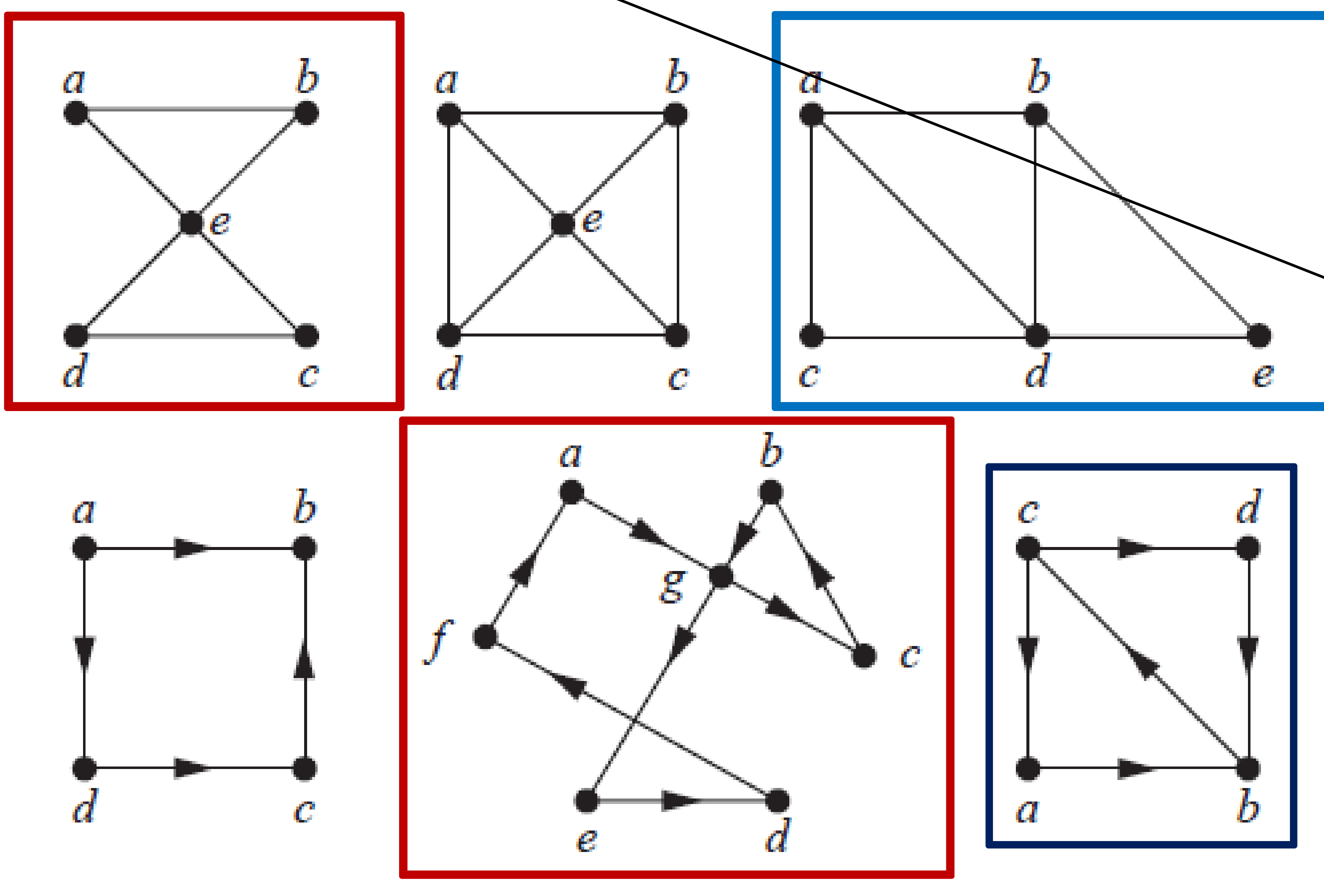
$$A_G = \begin{pmatrix} 0 & 1 & 1 & 0 \\ 1 & 0 & 0 & 1 \\ 1 & 0 & 0 & 1 \\ 0 & 1 & 1 & 0 \end{pmatrix}$$

$$A_G^2 = \begin{pmatrix} 2 & 0 & 0 & 2 \\ 0 & 2 & 2 & 0 \\ 0 & 2 & 2 & 0 \\ 2 & 0 & 0 & 2 \end{pmatrix} \quad A_G^3 = \begin{pmatrix} 0 & 4 & 4 & 0 \\ 4 & 0 & 0 & 4 \\ 4 & 0 & 0 & 4 \\ 0 & 4 & 4 & 0 \end{pmatrix} \quad A_G^4 = \begin{pmatrix} 8 & 0 & 0 & 8 \\ 0 & 8 & 8 & 0 \\ 0 & 8 & 8 & 0 \\ 8 & 0 & 0 & 8 \end{pmatrix}$$

Euler Paths and Circuits

DEFINITION: Let $G = (V, E)$ be a graph.

- **Euler Path**欧拉路径: a simple path that traverses every edge of G.
- **Euler Circuit**欧拉回路: a simple circuit that traverses every edge of G.



Remark: When G has multiple edges, these edges will be given different names and considered as different. This is implicit in the textbook.

Euler Circuits

THEOREM: Let $G = (V, E)$ be a connected multigraph of order ≥ 2.

Then G has an Euler circuit iff $2|\deg(x)$ for every $x \in V$.

- $\Rightarrow$: Let P: $\{x_0, x_1\}, \dots, \{x_{i-1}, x_i\}, \dots, \{x_{n-1}, x_n\}$ be an Euler circuit, $x_0 = x_n$
  - Every occurrence of x_i in P contributes 2 to $\deg(x_i)$
    - Every vertex x_i has an even degree
- $\Leftarrow$: Let P: $\{x_0, x_1\}, \dots, \{x_{n-1}, x_n\}$ be a longest simple path in G.
  - Let $H = G[P]$, the subgraph of G induced by all edges in P
    - If $x_n \neq x_0$, then $\deg_H(x_n)$ is odd and so P cannot be longest.
      - $x_n = x_0$, P is a simple circuit, and $2|\deg_H(x_i)$ for all i.
    - If $\exists i \in \{0,1, \dots, n-1\}$ such that $\deg_H(x_i) < \deg_G(x_i)$,
      - then $\exists y \in V$ such that $\{x_i, y\} \notin P$
        - $y, x_i, x_{i+1}, \dots, x_n, x_1, \dots, x_{i-1}, x_i$ is longer than P
    - Hence, $\deg_H(x_i) = \deg_G(x_i)$ for all $i \in \{0,1, \dots, n-1\}$.
      - $V = \{x_0, x_1, \dots, x_{n-1}\}$ and $H = G$.
        - P is an Euler circuit.

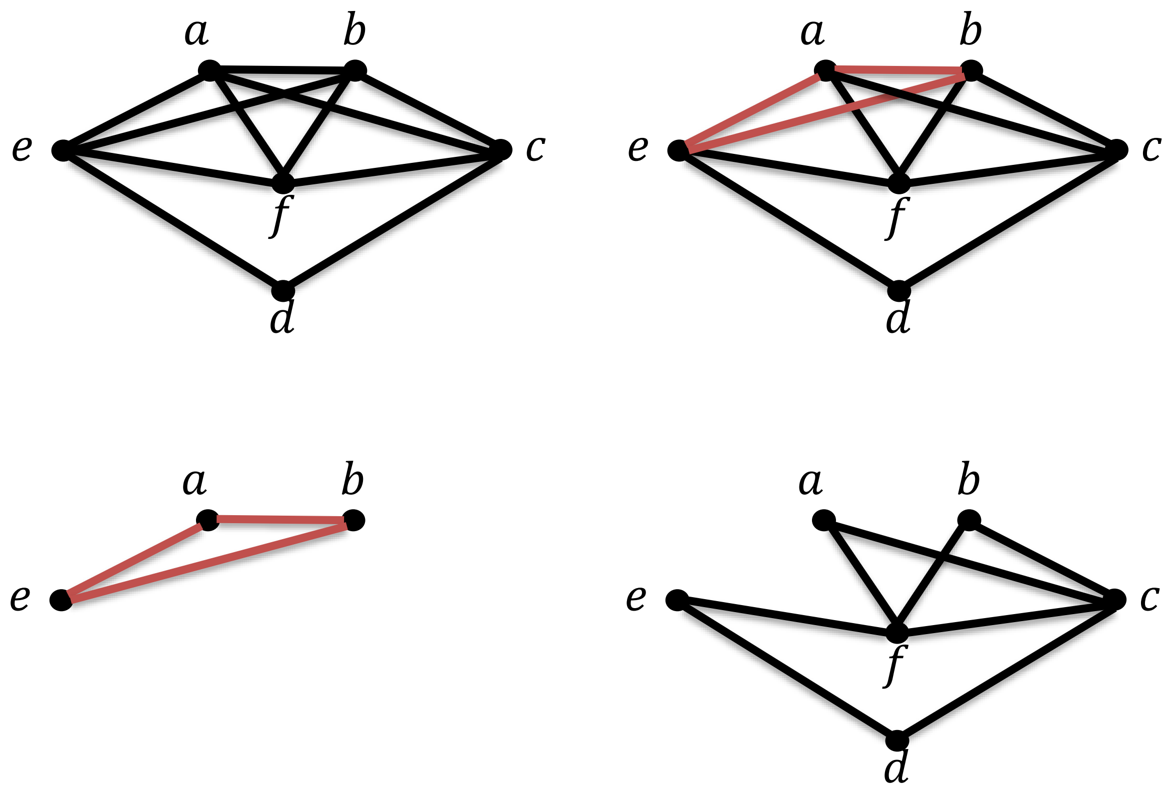
Remark: H contains all vertices of G. Otherwise, P can be extended.

Construction

ALGORITHM (Hierholzer):

- **Input**: $G = (V, E)$, a connected multigraph, $2 | \deg(x)$, $\forall x \in V$
- **Output**: an Euler circuit
  - **circuit**$:=$ a circuit in G
  - $H := G -$ **circuit** $-$ isolated vertices
  - while H has edges do
    - **subcircuit**$:=$ a circuit in H that intersects **circuit**
    - $H := H -$ **subcircuit** $-$ isolated vertices
    - **circuit**$:=$ **circuit** $\cup$ **subcircuit**
  - return **circuit**

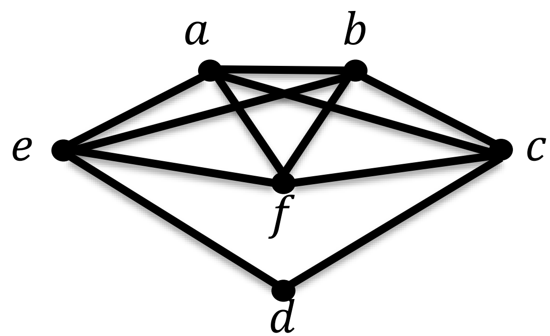
Example

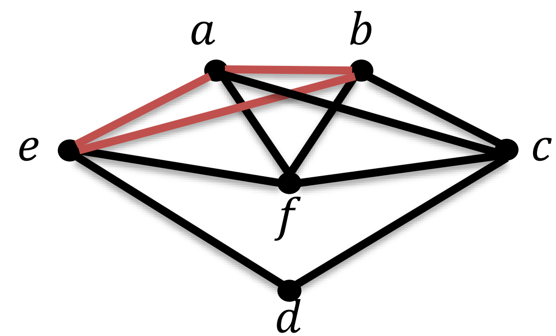


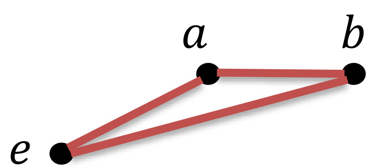
circuit $= a, b, e, a$

H

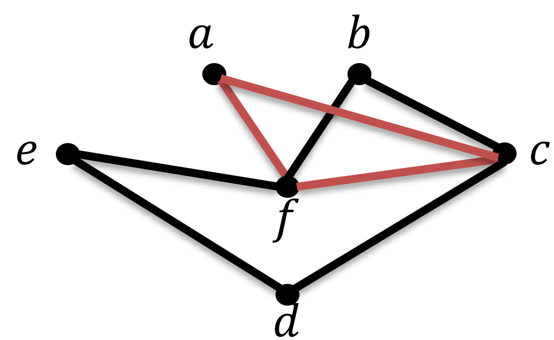
Example







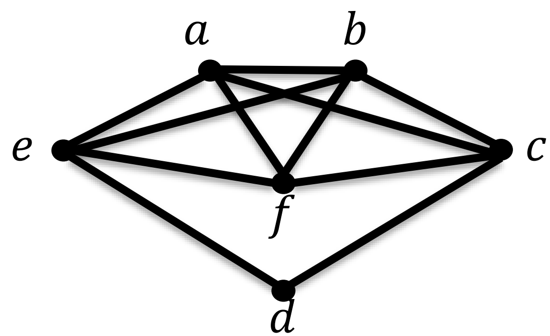
circuit $= a, b, e, a$

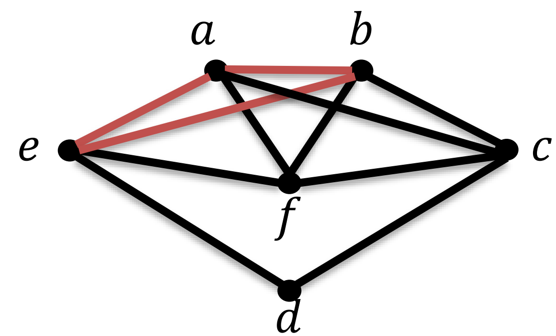


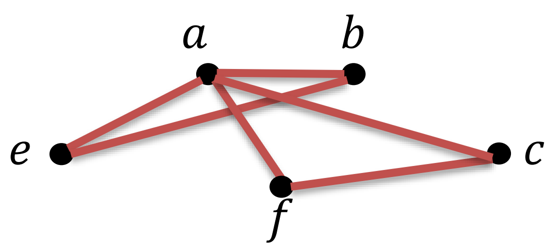
H

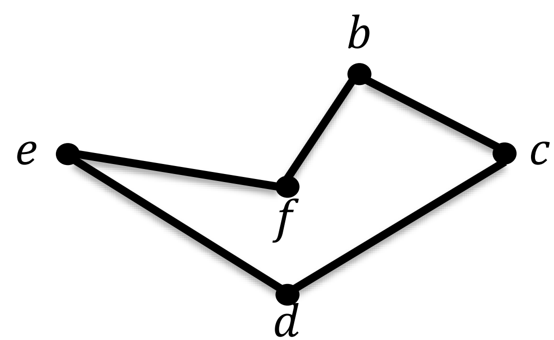
subcircuit $= a, c, f, a$

Example







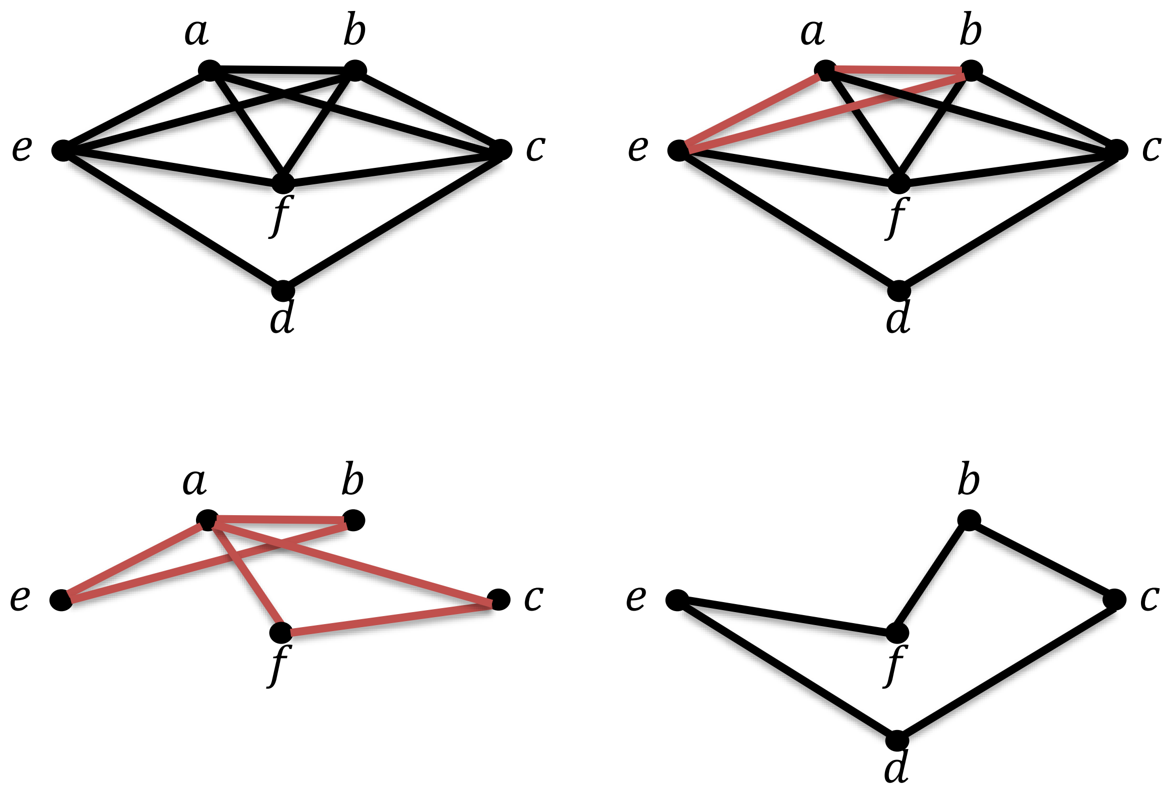


circuit $= a, b, e, a$

H

subcircuit $= a, c, f, a$

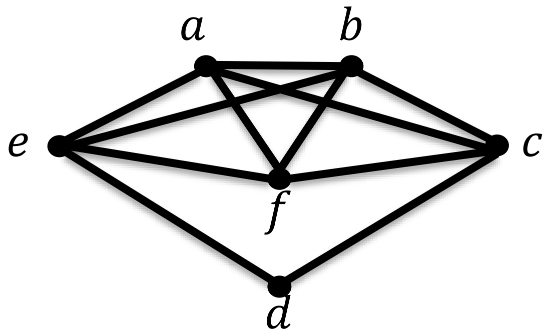
Example

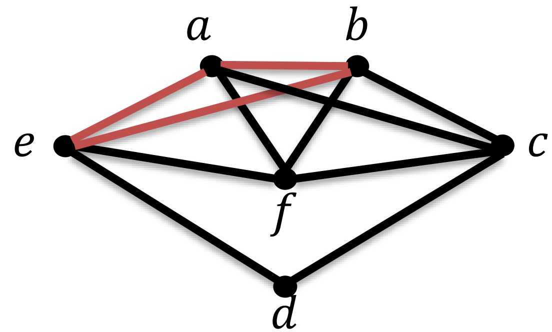


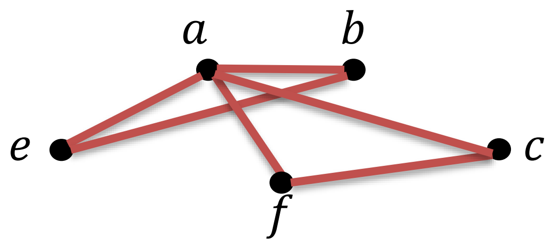
circuit $= a, b, e, a, c, f, a$

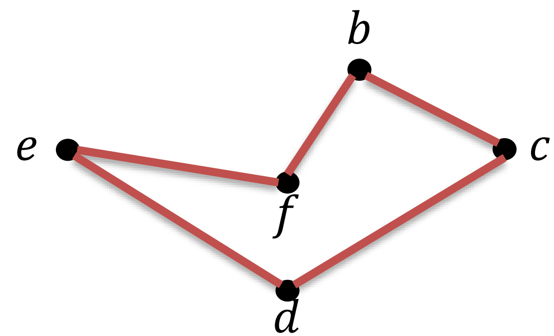
H

Example







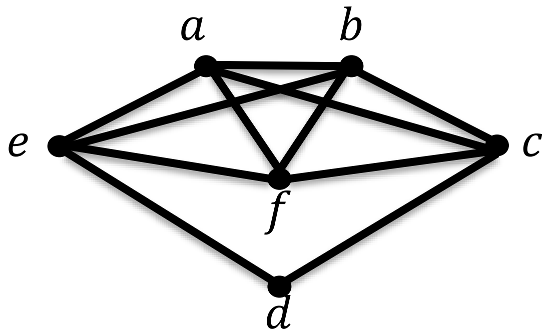


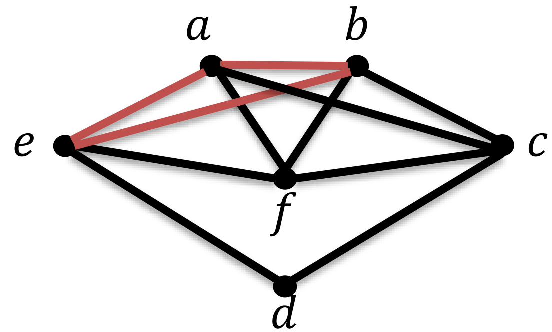
circuit $= a, b, e, a, c, f, a$

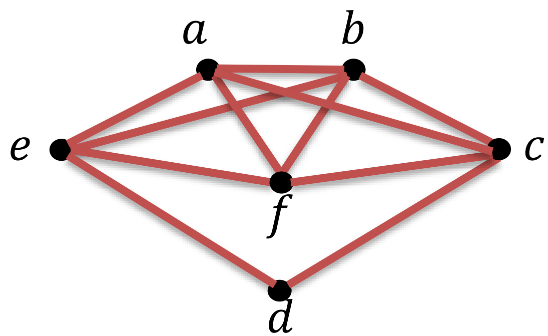
H

subcircuit $= c, d, e, f, b, c$

Example





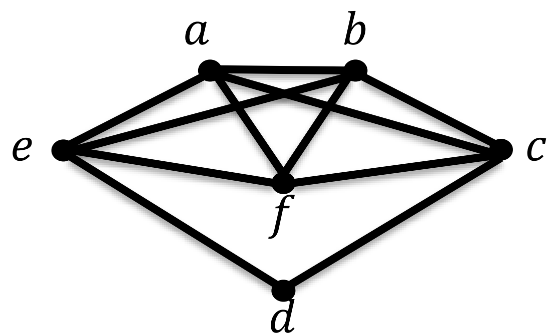


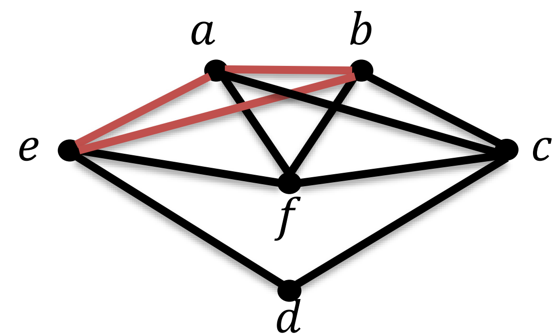
circuit $= a, b, e, a, c, f, a$

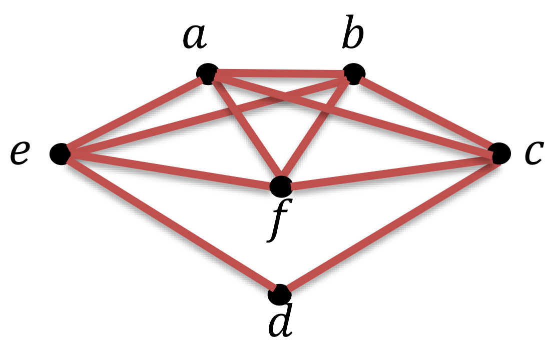
H

subcircuit $= c, d, e, f, b, c$

Example







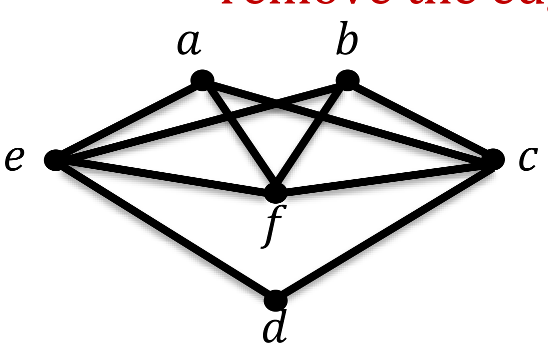
circuit $= a, b, e, a, c, d, e, f, b, c, f, a$

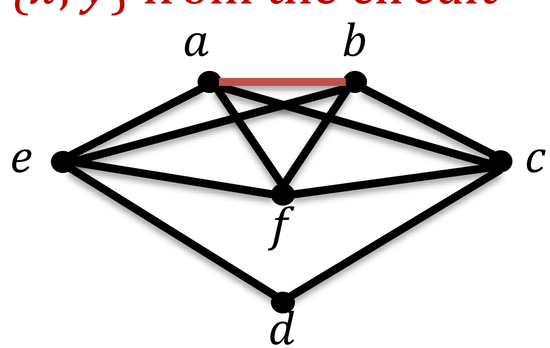
Euler Paths

THEOREM: Let $G = (V, E)$ be a connected multigraph of order ≥ 2.
Then G has an Euler path (not Euler circuit) iff G has exactly 2 vertices of odd degree.

ALGORITHM:

- **Input**: $G = (V, E)$, a connected multigraph, $x, y \in V$ have odd degrees
- **Output**: an Euler path
    - $H := G + \{x, y\}$
    - $\text{find an Euler circuit using Hierholzer's algorithm}$
    - $\text{remove the edge } \{x, y\} \text{ from the circuit}$





$a, c, d, e, f, b, a, e, b, c, f, a$

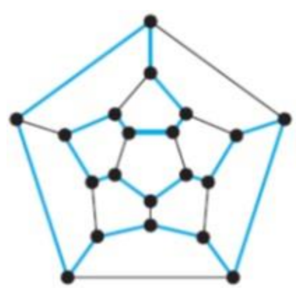
$a, c, d, e, f, b, a, e, b, c, f, a$

$a, e, b, c, f, a, c, d, e, f, b$

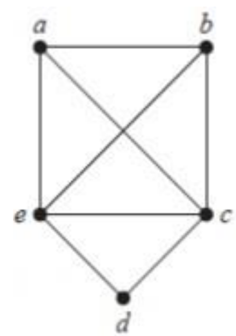
Hamilton Paths and Circuits

DEFINITION: Let $G = (V, E)$ be a graph.

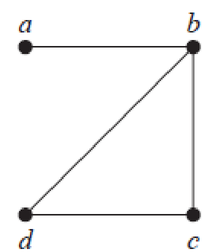
- **Hamilton Path**: A simple path that passes through every vertex exactly once.
- **Hamilton Circuit:** A simple circuit that passes through every vertex exactly once.



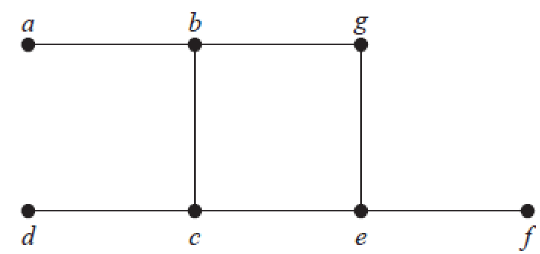
√ Hamilton path
√ Hamilton circuit



√ Hamilton path
√ Hamilton circuit



√ Hamilton path
×Hamilton circuit



× Hamilton path
× Hamilton circuit

Hamilton Circuits

Determine if there is a Hamilton circuit in a given graph G?

- This problem is NP-Complete. //that means very difficult

Necessary conditions on Hamilton circuit.

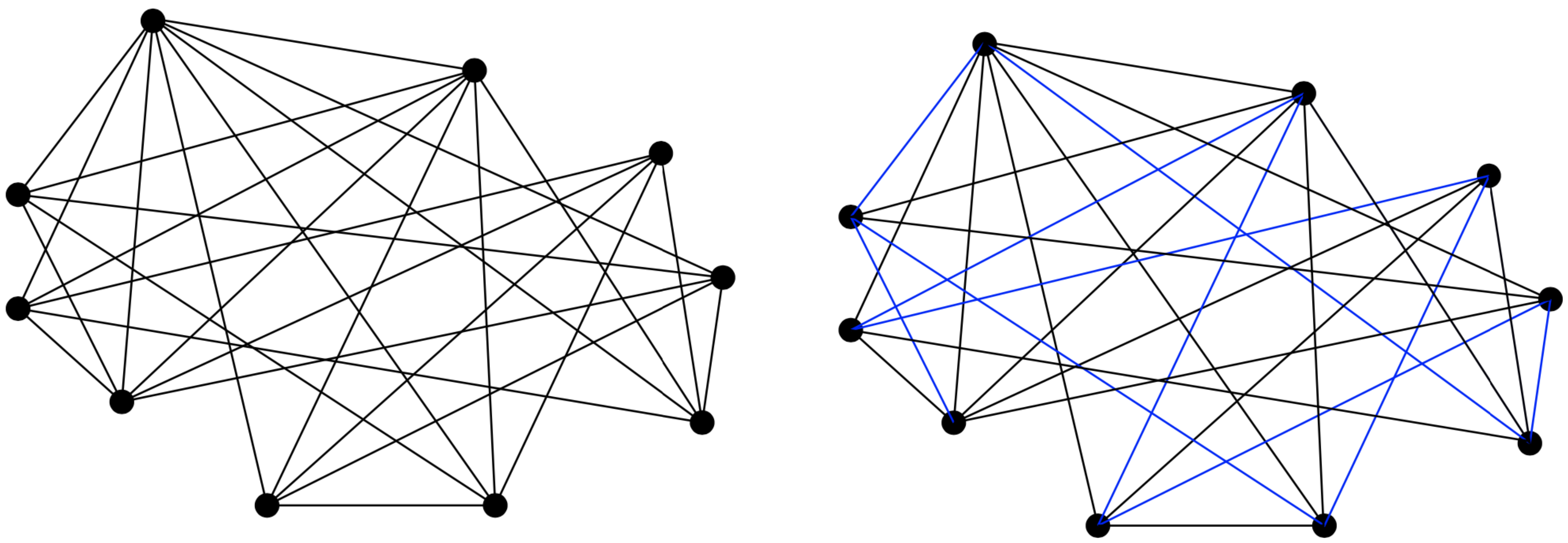
- If G has a vertex of degree 1, then G cannot have a Hamilton circuit.
- If G has a vertex of degree 2, then a Hamilton circuit of G traverses both edges.

Sufficient conditions on Hamilton circuit.

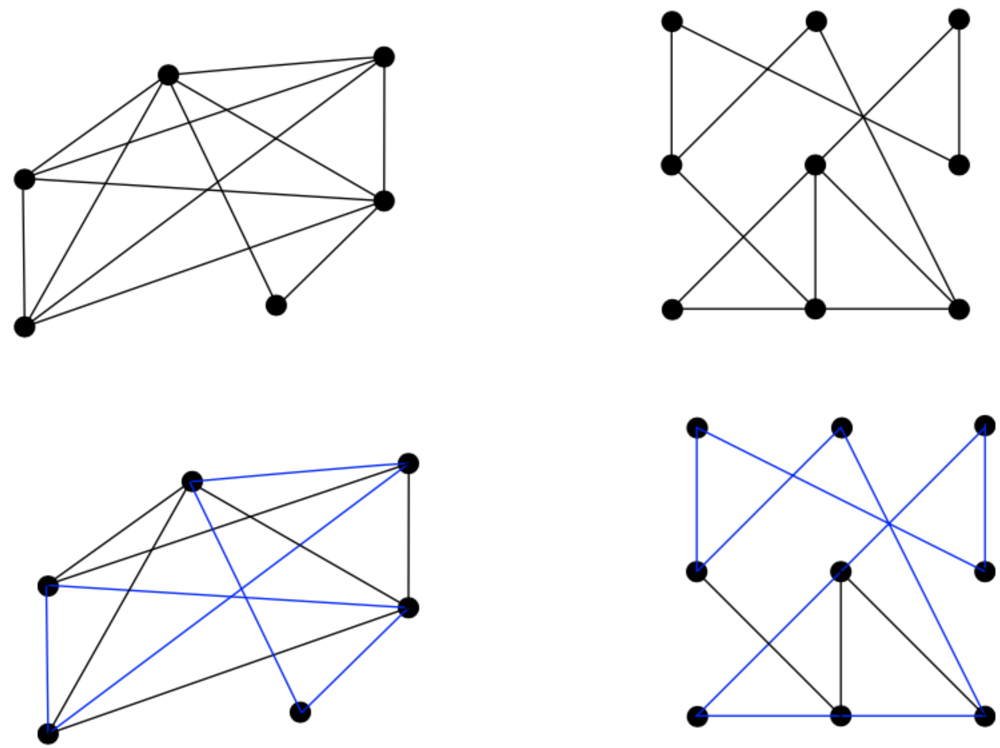
- **Ore's Theorem:** Let $G = (V, E)$ be a simple graph of order $n \geq 3$. If $\deg(u) + \deg(v) \geq n$ for all $\{u, v\} \notin E$, then G has a Hamilton circuit.
- **Dirac's Theorem:** Let $G = (V, E)$ be a simple graph of order $n \geq 3$. If $\deg(u) \geq n/2$ for every $u \in V$, then G has a Hamilton circuit.
    - This is a corollary of Ore's Theorem
        - $\forall u \in V,\ \deg(u) \geq n/2 \Rightarrow \forall u, v \in V, \deg(u) + \deg(v) \geq n$

Hamilton Circuits

- Examples (sufficient condition)



Hamilton Circuits

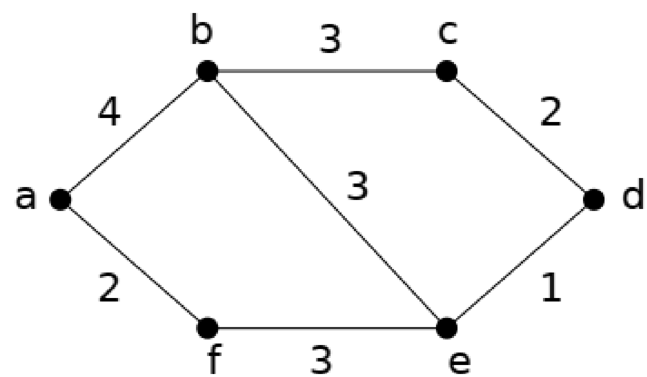


Remark: Dirac's and Ore's Theorems do not give a necessary condition for the existence of a Hamilton circuit!

Shortest Path Problem

Definition

A **weighted graph** is a graph $G = (V, E)$ such that each edge is assigned with a strictly positive number.
The **length** of a path in weighted graph is the sum of the weights of the edges of this path.

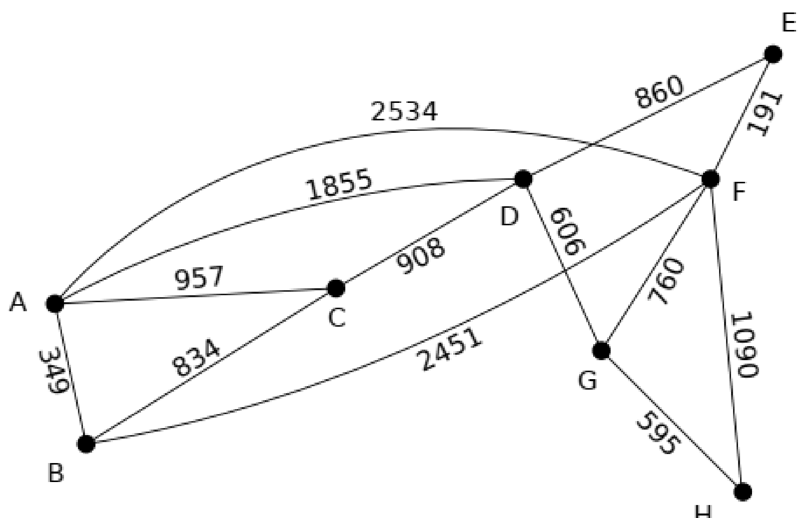


a, b, c is a path of length 7 and b, e, d, c is a path of length 6

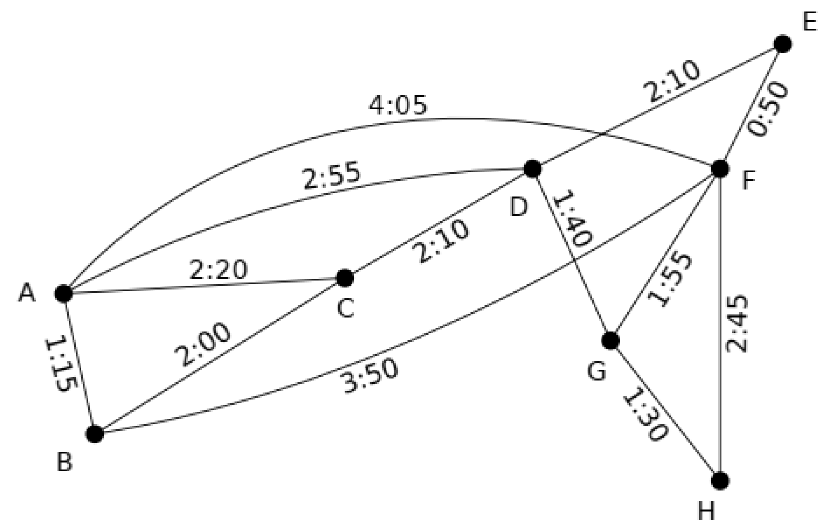
Remark: Observe that in a non-weighted graph the length of a path is the number of edges in the path!

Shortest Path Problem

Examples



distance

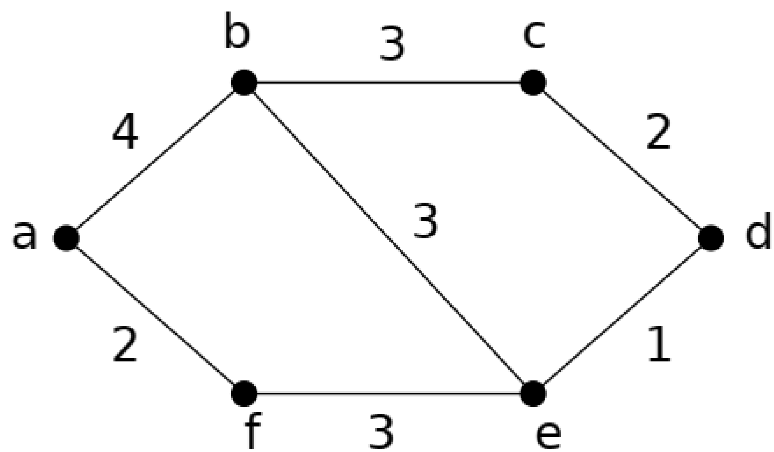


time

What is the shortest path in air distance between cities A and E?
What combination of flights has the smallest total flight time?

Shortest Path Problem

Question: Find the shortest path from a to d.



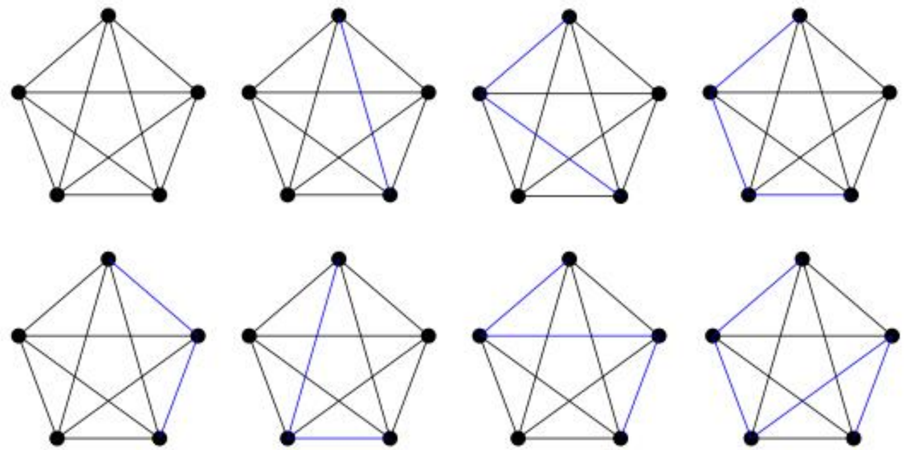
Method: Find the closest vertex to a, then the second closest, the third closest... until we reach d.

$\Rightarrow$ Dijkstra's algorithm

Shortest Path Problem

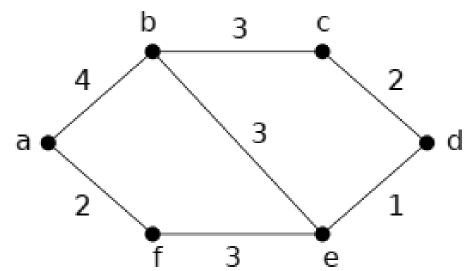
Remarks:

- Of course in the example above, we could have looked at all the paths between a and d and compute their length, but too complicated if the graph has a lot of edges.



- Advantage of Dijkstra's algorithm: we can compute the length of a shortest path from one vertex to all other vertices of the graph.

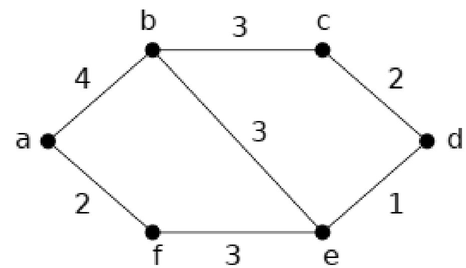
Djikstra's Algorithm



1. Find the closest vertex to $a \rightsquigarrow$ analyse all the edges starting from a:
   a, b of length 4
   a, f of length 2
   $\Rightarrow$ f is the closest vertex to a. The shortest path from a to f has length 2.

2. Find the second closest vertex to $a \rightsquigarrow$ shortest paths from a to a vertex in $\{a, f\}$ followed by an edge from a vertex in $\{a, f\}$ to a vertex not in this set:
   a, b of length 4
   a, f, e of length 5
   $\Rightarrow$ b is the second closest vertex to a. The shortest path from a to b has length 4.

Djikstra's Algorithm



3. Find the third closest vertex to $a \rightsquigarrow$ shortest path from a to a vertex in $\{a, f, b\}$ followed by an edge from a vertex in $\{a, f, b\}$ to a vertex not in this set:
   a, b, c of length 7
   a, b, e of length 7
   a, f, e of length 5
   $\Rightarrow$ e is the third closest vertex to a. The shortest path from a to e has length 5.

4. Find the fourth closest vertex to $a \rightsquigarrow$ shortest path from a to a vertex in $\{a, f, b, e\}$ followed by an edge from a vertex in $\{a, f, b, e\}$ to a vertex not in this set:
   a, b, c of length 7
   a, f, e, d of length 6
   $\Rightarrow$ d is the fourth closest vertex to a. The shortest path from a to d has length 6.

Djikstra's Algorithm

Goal: find the length of a shortest path from a to z with a series of iterations.

• A distinguished set of vertices is constructed by adding one vertex at each iteration.

• A labeling procedure is carried out at each iteration: a vertex w is labeled with the length of a shortest path from a to w that contains only vertices in the distinguished set.

• The vertex added to the distinguished set is one with minimal label among those vertices not already in the set.

Notations: $S_k :=$ distinguished set after k iterations, $L_k(v) :=$ length of a shortest path from a to v containing only vertices in S_k ("label" of v).

Initialization: $L_0(a) = 0$,
$L_0(v) = \infty$ for every vertex $v \neq a$,
$S_0 = \emptyset$.

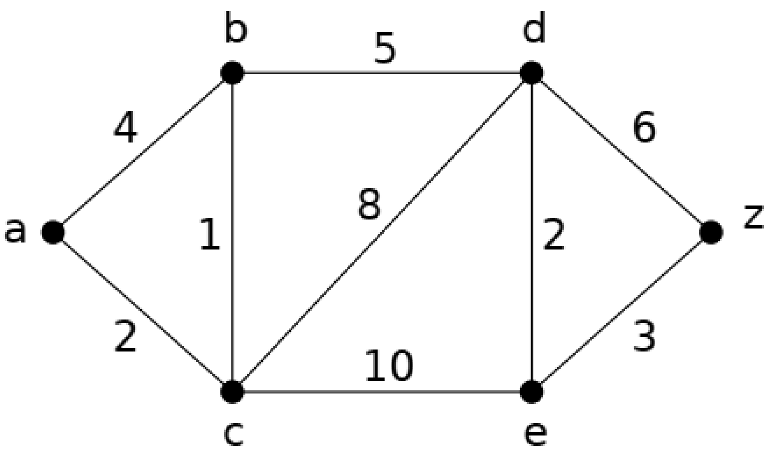
***k*th iteration:**

• S_k is formed from S_{k-1} by adding a vertex u not in S_{k-1} with smallest label,

• Update the labels of all vertices not in S_k so that $L_k(v)$ is the length of a shortest path from a to v containing only vertices in S_k, i.e.

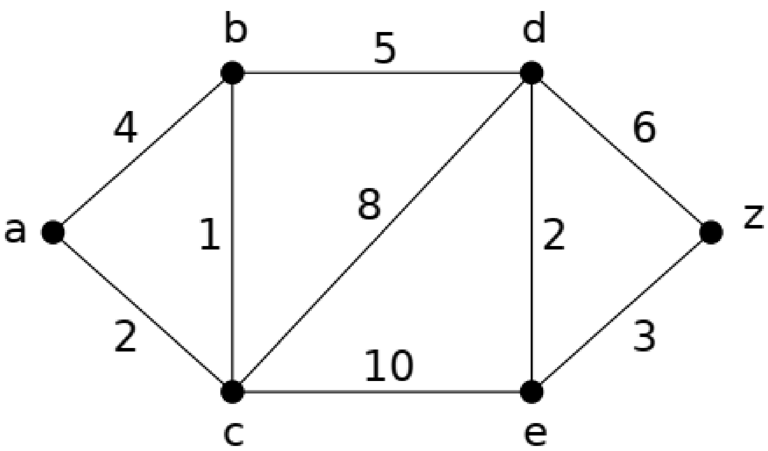
$L_k(v) = min\{L_{k-1}(v), L_{k-1}(u) + w(u, v)\}$ (with $w(u, v)$ length of the edge (u, v))

Djikstra’s Algorithm



- **k=0 (initialization):** $S_0 = \emptyset$, $L_0(a) = 0$, $L_0(b) = L_0(c) = L_0(d) = L_0(e) = L_0(z) = \infty$

Djikstra's Algorithm



- **k=0 (initialization):** $S_0 = \emptyset$, $L_0(a) = 0,\ L_0(b) = L_0(c) = L_0(d) = L_0(e) = L_0(z) = \infty$

- **k=1:** $u := a \rightsquigarrow S_1 = \{a\}$,
  $L_0(a) + w(a,b) = 4 < L_0(b) \rightsquigarrow L_1(b) = 4$
  $L_0(a) + w(a,c) = 2 < L_0(c) \rightsquigarrow L_1(c) = 2$

- **k=2:** $u := c \rightsquigarrow S_1 = \{a,c\}$,
  $L_1(c) + w(c,b) = 3 < L_1(b) \rightsquigarrow L_2(b) = 3$
  $L_1(c) + w(c,d) = 10 < L_1(d) \rightsquigarrow L_2(d) = 10$
  $L_1(c) + w(c,e) = 12 < L_1(e) \rightsquigarrow L_2(e) = 12$

- **k=3:** $u := b \rightsquigarrow S_1 = \{a,c,b\}$,
  $L_2(b) + w(b,d) = 8 < L_2(d) \rightsquigarrow L_3(d) = 8$

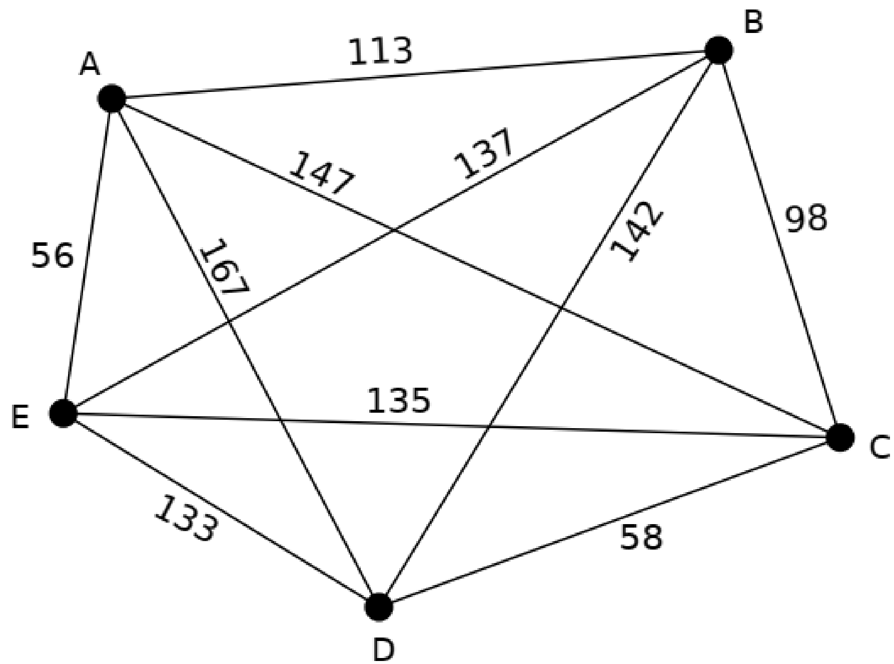
- **k=4:** $u := d \rightsquigarrow S_1 = \{a,c,b,d\}$,
  $L_3(d) + w(d,e) = 10 < L_3(e) \rightsquigarrow L_4(e) = 10$
  $L_3(d) + w(d,z) = 14 < L_3(z) \rightsquigarrow L_4(z) = 14$

- **k=5:** $u := e \rightsquigarrow S_1 = \{a,c,b,d,e\}$,
  $L_4(e) + w(e,z) = 13 < L_4(z) \rightsquigarrow L_5(z) = 13$

- **k=6:** $u := z \rightsquigarrow S_1 = \{a,c,b,d,z\}$,

- **return:** $L(z) = 13$

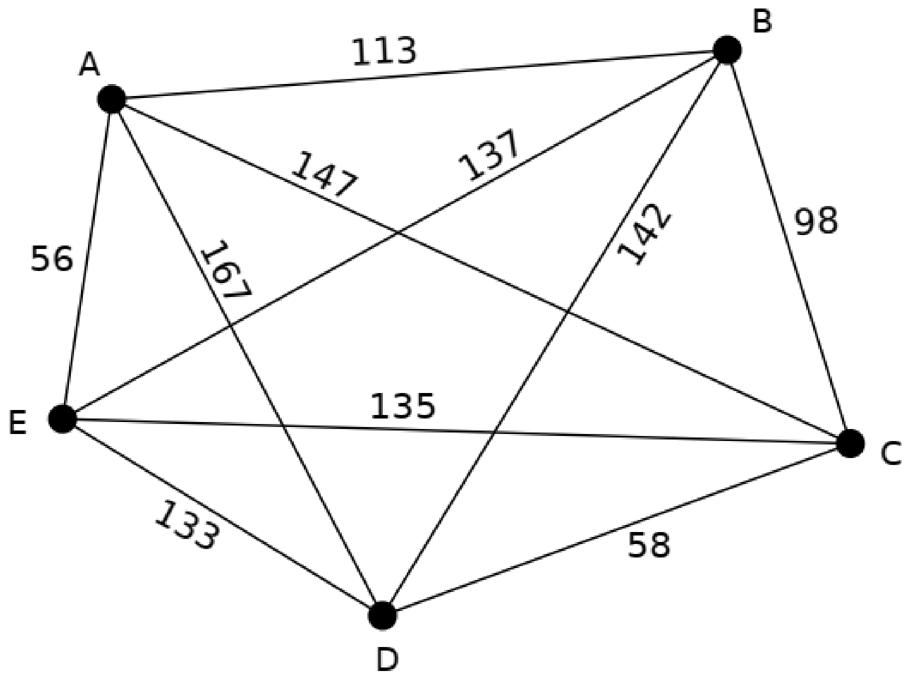
Traveling Salesperson Problem



Traveling salesperson problem: a traveling salesperson wants to visit each of the cities once and return to his starting point. In which order should he visit these cities to travel the minimum total distance?

$\Rightarrow$ **Hamiltonian circuit with minimum total weight in the complete graph.**

Traveling Salesperson Problem



Route	Tot. dist.
A, B, C, D, E, A	610
A, B, C, E, D, A	516
A, B, E, D, C, A	588
A, B, E, C, D, A	458
A, B, D, E, C, A	540
A, B, D, C, E, A	504
A, D, B, C, E, A	598
A, D, B, E, C, A	576
A, D, E, B, C, A	682
A, D, C, B, E, A	646
A, C, D, B, E, A	670
A, C, B, D, E, A	728

Traveling salesperson problem: a traveling salesperson wants to visit each of the cities once and return to his starting point. In which order should he visit these cities to travel the minimum total distance?

$\Rightarrow$ **Hamiltonian circuit with minimum total weight in the complete graph.**

Discrete Mathematics: Lecture 27

Shortest Paths and Djikstra’s Algorithm, Traveling Salesperson Problem, Planar Graph, Euler’s Formula, Kuratowski's Theorem

Xuming He
Associate Professor

School of Information Science and Technology
ShanghaiTech University

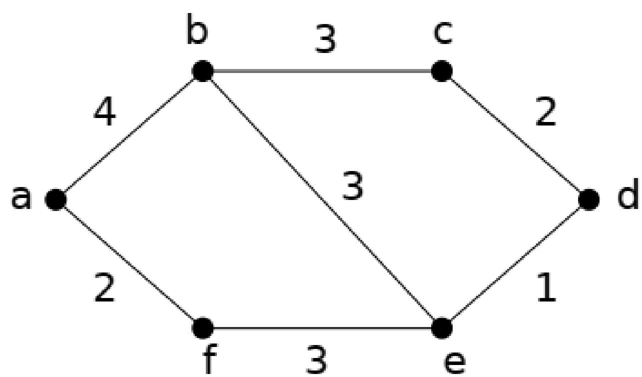
Spring Semester, 2022

Notes by Prof. Liangfeng Zhang

Shortest Path Problem

Definition

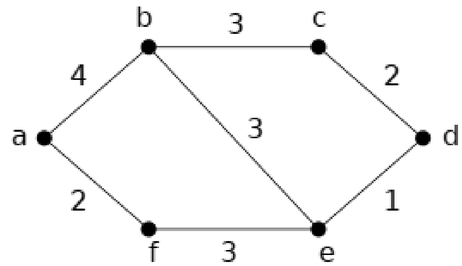
A **weighted graph** is a graph $G = (V, E)$ such that each edge is assigned with a strictly positive number.
The **length** of a path in weighted graph is the sum of the weights of the edges of this path.



a, b, c is a path of length 7 and b, e, d, c is a path of length 6

Remark: Observe that in a non-weighted graph the length of a path is the number of edges in the path!

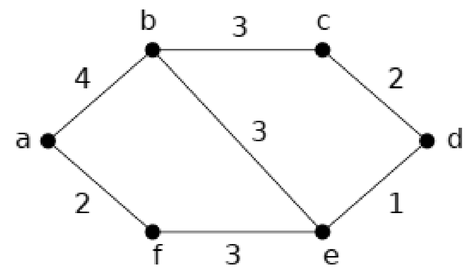
Djikstra's Algorithm



1. Find the closest vertex to $a \rightsquigarrow$ analyse all the edges starting from a:
   a, b of length 4
   a, f of length 2
   $\Rightarrow$ f is the closest vertex to a. The shortest path from a to f has length 2.

2. Find the second closest vertex to $a \rightsquigarrow$ shortest paths from a to a vertex in $\{a, f\}$ followed by an edge from a vertex in $\{a, f\}$ to a vertex not in this set:
   a, b of length 4
   a, f, e of length 5
   $\Rightarrow$ b is the second closest vertex to a. The shortest path from a to b has length 4.

Djikstra's Algorithm



3. Find the third closest vertex to $a \rightsquigarrow$ shortest path from a to a vertex in $\{a, f, b\}$ followed by an edge from a vertex in $\{a, f, b\}$ to a vertex not in this set:
   a, b, c of length 7
   a, b, e of length 7
   a, f, e of length 5
   $\Rightarrow$ e is the third closest vertex to a. The shortest path from a to e has length 5.

4. Find the fourth closest vertex to $a \rightsquigarrow$ shortest path from a to a vertex in $\{a, f, b, e\}$ followed by an edge from a vertex in $\{a, f, b, e\}$ to a vertex not in this set:
   a, b, c of length 7
   a, f, e, d of length 6
   $\Rightarrow$ d is the fourth closest vertex to a. The shortest path from a to d has length 6.

Djikstra's Algorithm

Goal: find the length of a shortest path from a to z with a series of iterations.

- A distinguished set of vertices is constructed by adding one vertex at each iteration.
- A labeling procedure is carried out at each iteration: a vertex w is labeled with the length of a shortest path from a to w that contains only vertices in the distinguished set.
- The vertex added to the distinguished set is one with minimal label among those vertices not already in the set.

Notations: $S_k :=$ distinguished set after k iterations, $L_k(v) :=$ length of a shortest path from a to v containing only vertices in S_k ("label" of v).

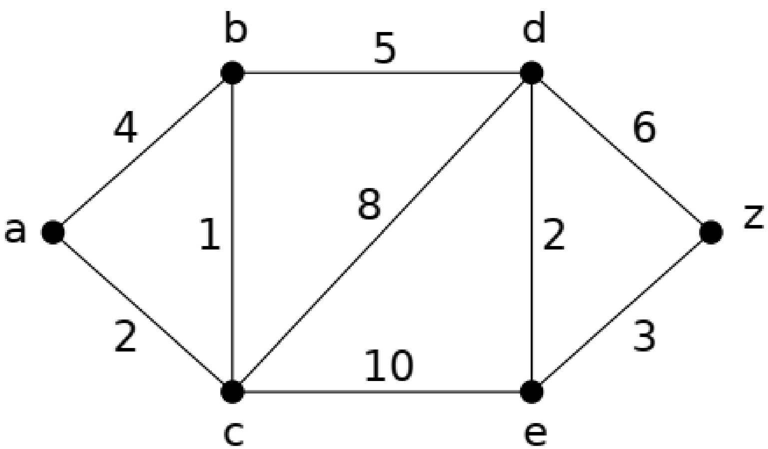
Initialization: $L_0(a) = 0$,
$L_0(v) = \infty$ for every vertex $v \neq a$,
$S_0 = \emptyset$.

kth iteration:

- S_k is formed from S_{k-1} by adding a vertex u not in S_{k-1} with smallest label,
- Update the labels of all vertices not in S_k so that $L_k(v)$ is the length of a shortest path from a to v containing only vertices in S_k, i.e.

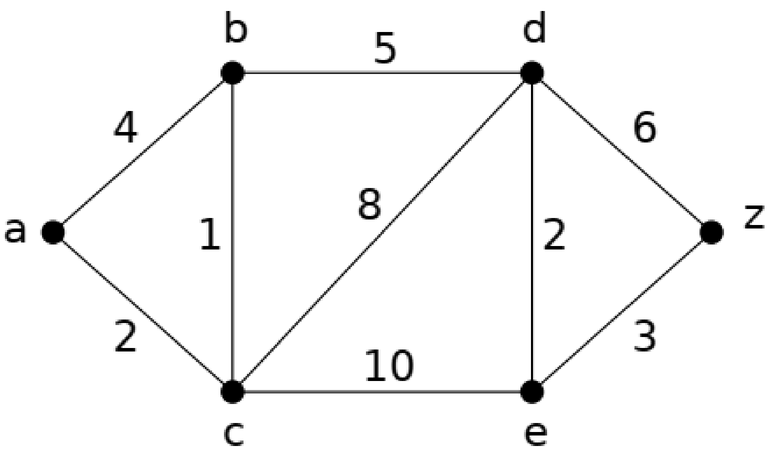
$L_k(v) = min\{L_{k-1}(v), L_{k-1}(u) + w(u,v)\}$ (with $w(u,v)$ length of the edge (u,v))

Djikstra's Algorithm



- **k=0 (initialization):** $S_0 = \emptyset$, $L_0(a) = 0,\ L_0(b) = L_0(c) = L_0(d) = L_0(e) = L_0(z) = \infty$

Djikstra's Algorithm



- **k=0 (initialization):** $S_0 = \emptyset$, $L_0(a) = 0,\ L_0(b) = L_0(c) = L_0(d) = L_0(e) = L_0(z) = \infty$

- **k=1:** $u := a \rightsquigarrow S_1 = \{a\}$,
  $L_0(a) + w(a,b) = 4 < L_0(b) \rightsquigarrow L_1(b) = 4$
  $L_0(a) + w(a,c) = 2 < L_0(c) \rightsquigarrow L_1(c) = 2$

- **k=2:** $u := c \rightsquigarrow S_1 = \{a, c\}$,
  $L_1(c) + w(c,b) = 3 < L_1(b) \rightsquigarrow L_2(b) = 3$
  $L_1(c) + w(c,d) = 10 < L_1(d) \rightsquigarrow L_2(d) = 10$
  $L_1(c) + w(c,e) = 12 < L_1(e) \rightsquigarrow L_2(e) = 12$

- **k=3:** $u := b \rightsquigarrow S_1 = \{a, c, b\}$,
  $L_2(b) + w(b,d) = 8 < L_2(d) \rightsquigarrow L_3(d) = 8$

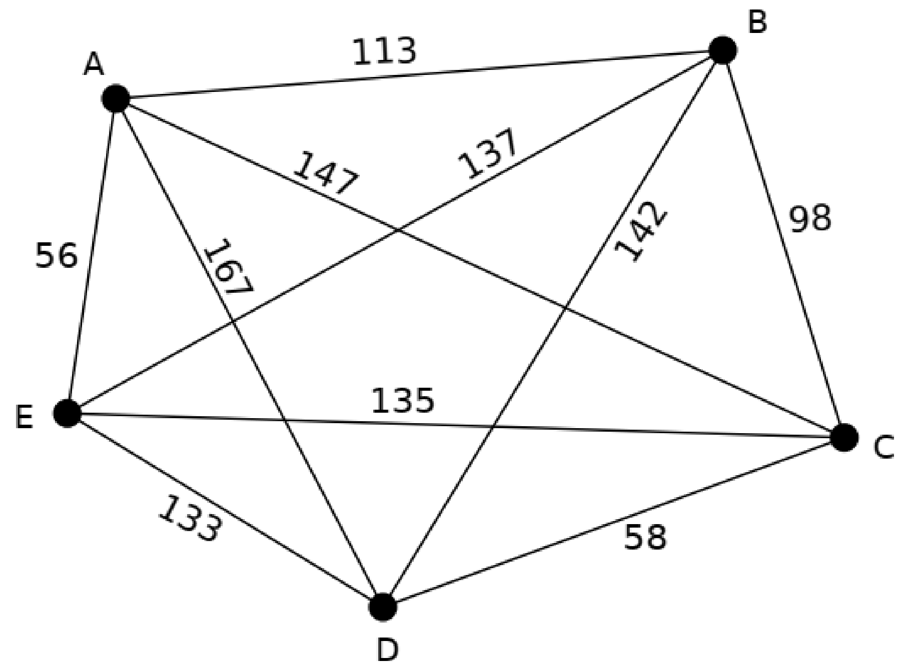
- **k=4:** $u := d \rightsquigarrow S_1 = \{a, c, b, d\}$,
  $L_3(d) + w(d,e) = 10 < L_3(e) \rightsquigarrow L_4(e) = 10$
  $L_3(d) + w(d,z) = 14 < L_3(z) \rightsquigarrow L_4(z) = 14$

- **k=5:** $u := e \rightsquigarrow S_1 = \{a, c, b, d, e\}$,
  $L_4(e) + w(e,z) = 13 < L_4(z) \rightsquigarrow L_5(z) = 13$

- **k=6:** $u := z \rightsquigarrow S_1 = \{a, c, b, d, z\}$,

- **return:** $L(z) = 13$

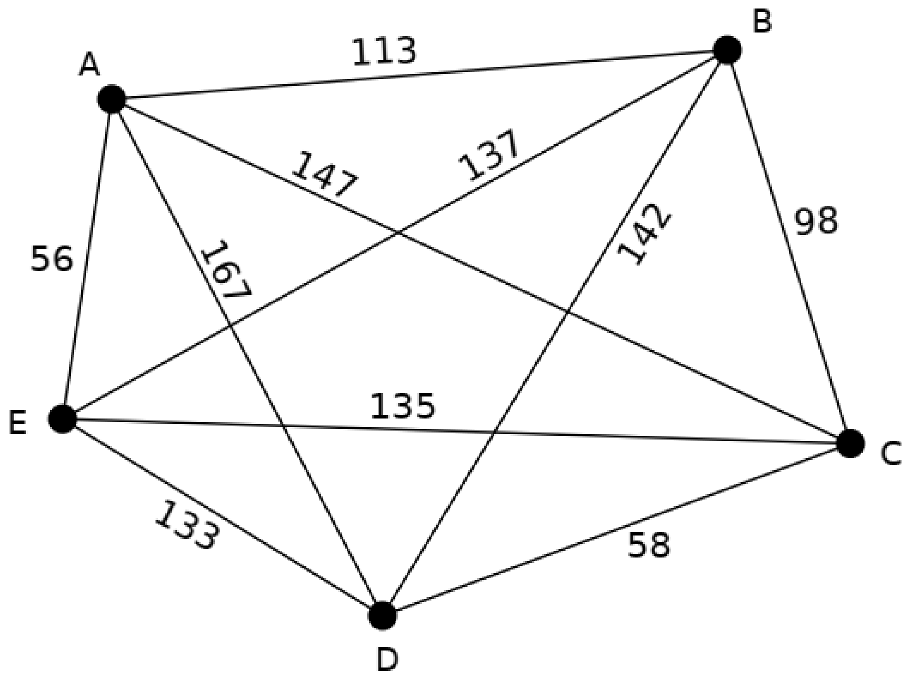
Traveling Salesperson Problem



Traveling salesperson problem: a traveling salesperson wants to visit each of the cities once and return to his starting point. In which order should he visit these cities to travel the minimum total distance?

$\Rightarrow$ **Hamiltonian circuit with minimum total weight in the complete graph.**

Traveling Salesperson Problem



Route	Tot. dist.
A, B, C, D, E, A	610
A, B, C, E, D, A	516
A, B, E, D, C, A	588
A, B, E, C, D, A	458
A, B, D, E, C, A	540
A, B, D, C, E, A	504
A, D, B, C, E, A	598
A, D, B, E, C, A	576
A, D, E, B, C, A	682
A, D, C, B, E, A	646
A, C, D, B, E, A	670
A, C, B, D, E, A	728

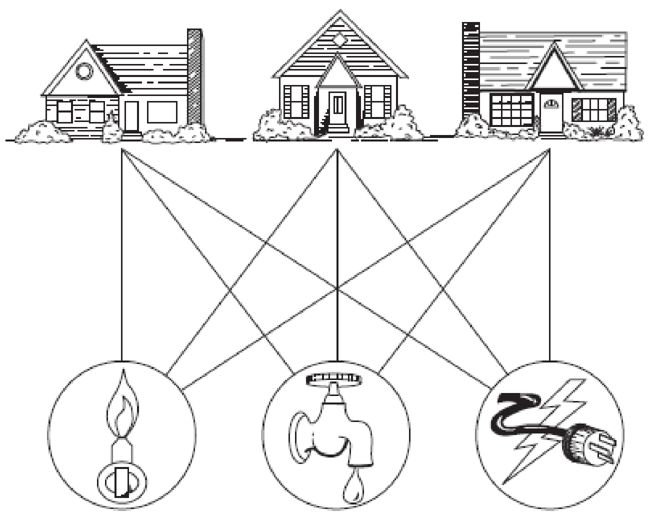
Traveling salesperson problem: a traveling salesperson wants to visit each of the cities once and return to his starting point. In which order should he visit these cities to travel the minimum total distance?

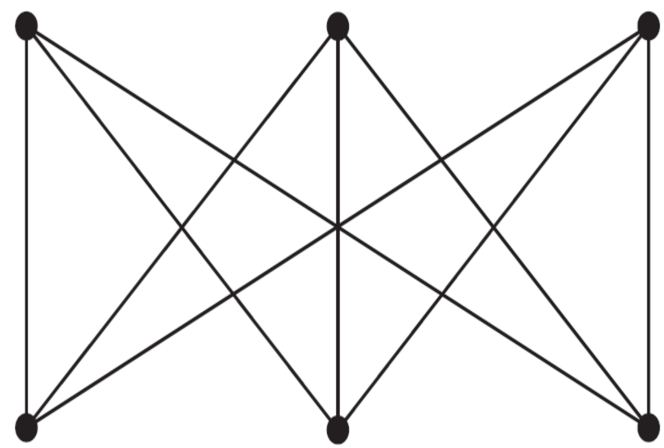
$\Rightarrow$ **Hamiltonian circuit with minimum total weight in the complete graph.**

Planar Graph

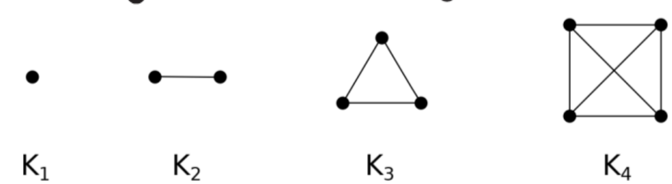
DEFINITION: Let $G = (V, E)$ be an undirected graph. G is called a **planar graph**平面图 if it can be drawn in the plane without any edges crossing.

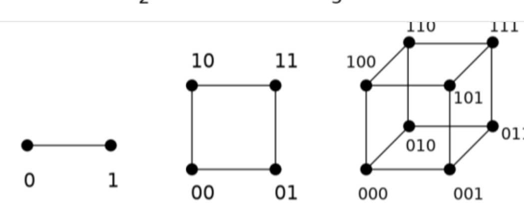
- Crossing of edges: an intersection other than endpoints (vertices)
- **planar representation**平面表示: a drawing w/o edge crossing; **nonplanar**非平面的





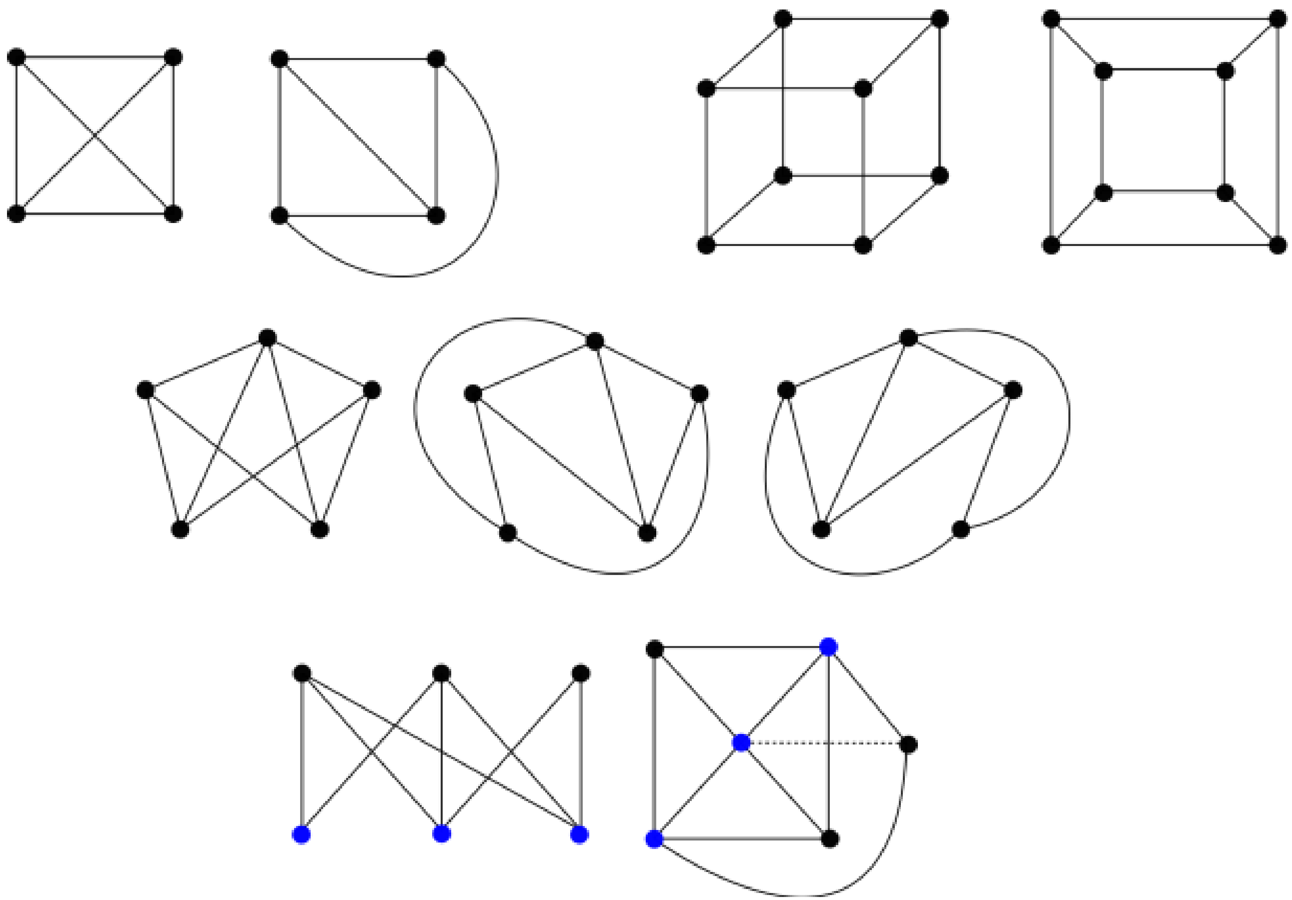
- K_1, K_2, K_3, K_4 are planar graphs
- $K_{1,n}, K_{2,n}$ are planar graphs
- $C_n\ (n \geq 3), W_n (n \geq 3)$ are planar graphs
- Q_1, Q_2, Q_3 are planar graphs





Planar Graph

Examples

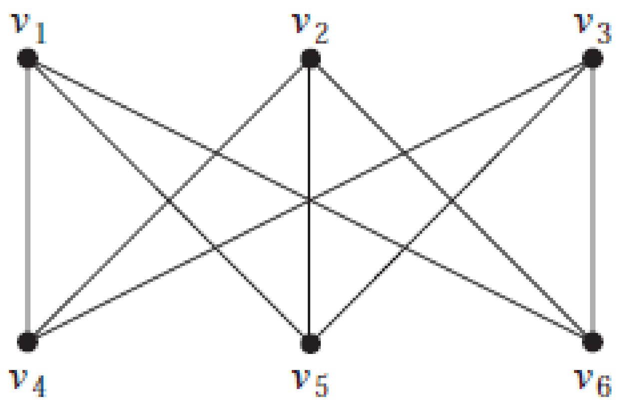


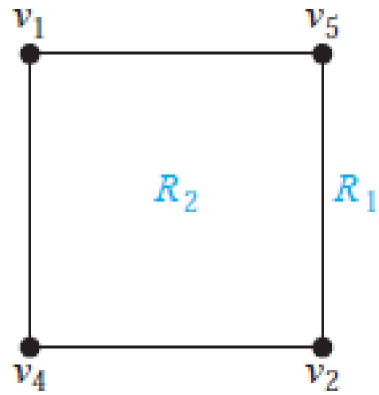
A graph may be planar even if it is usually drawn with crossings, because it may be possible to draw it in a different way without crossings.

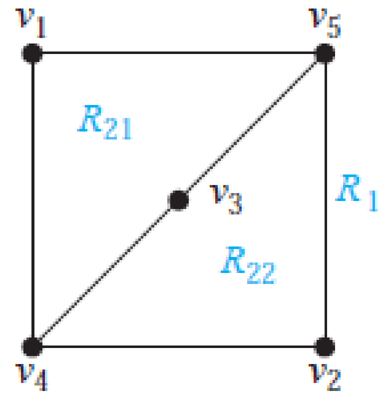
Nonplanar Graph

Jordan Curve Theorem: Every **simple closed planar curve** Γ separates the plane into a bounded interior region and an unbounded exterior region. Any planar curve connecting the two regions must intersect Γ.

EXAMPLE: The bipartite graph $K_{3,3}$ is not planar.



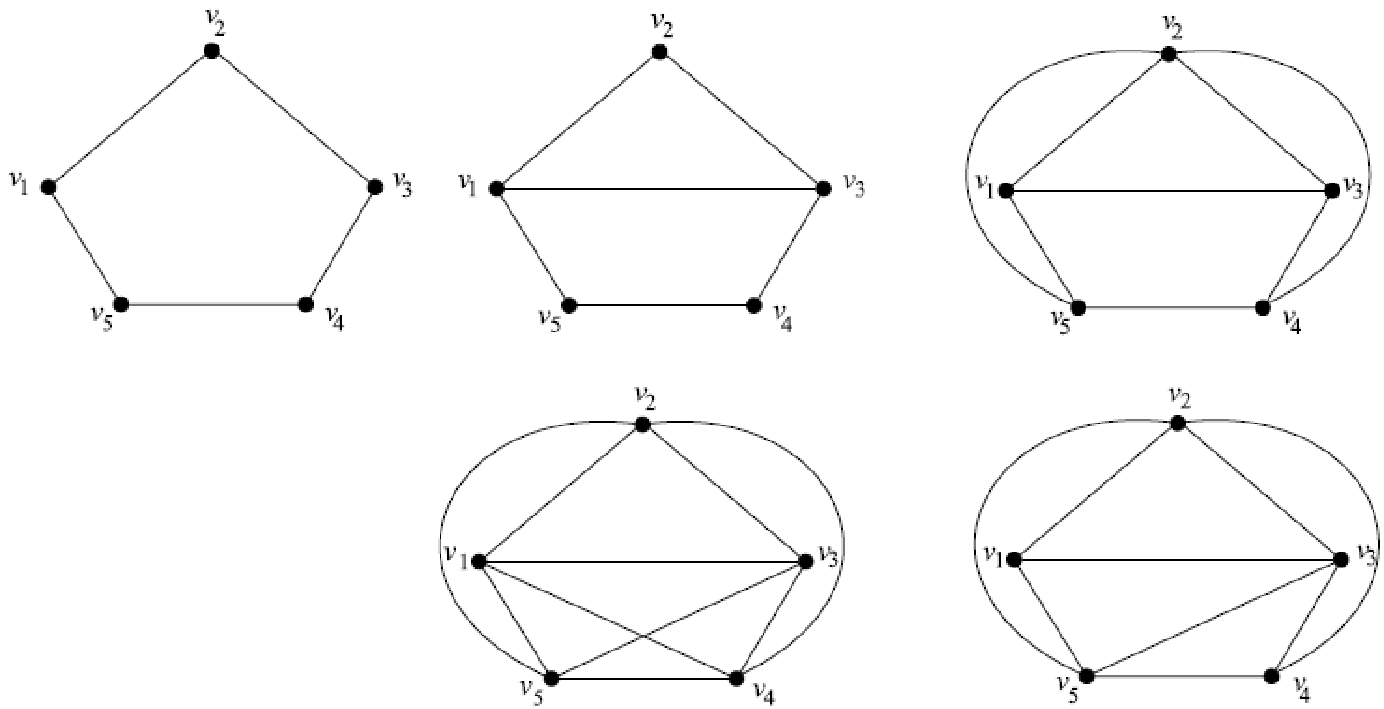




- choose a simple circuit v_1, v_5, v_2, v_4, v_1 in $K_{3,3}$
- If $K_{3,3}$ is a planar, then the circuit forms a simple closed planar curve
- Add v_3, v_6 and the edges incident with them.
  - Intersection occurs (due to the Jordan curve Theorem).

Nonplanar Graph

EXAMPLE: The complete graph K_5 is not planar.

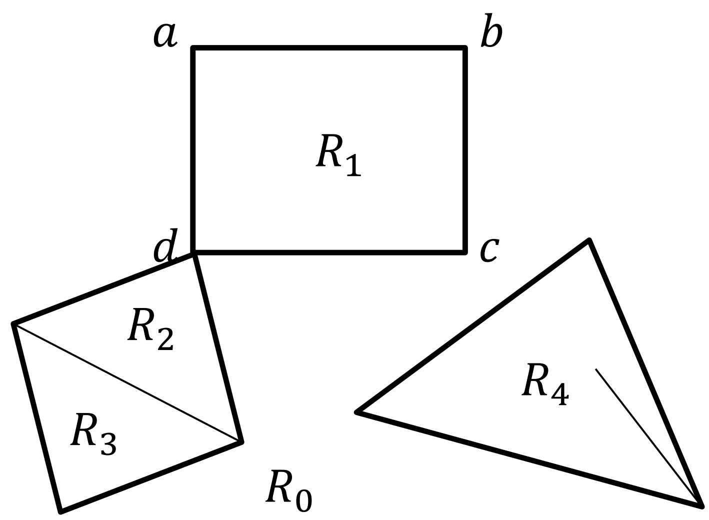


- $v_1, v_2, v_3, v_4, v_5, v_1$ is a simple closed curve in the planar representation of K_5
- Every remaining edge is in the interior region or in the exterior region
  - at least one is in the interior region
- No matter how you draw the remaining edges, crossing occurs.

Regions

DEFINITION: Let $G = (V, E)$ be a planar graph. Then the plane is divided into several **regions**面 by the edges of G.

- The infinite region is **exterior region**外部面. The others are **interior regions**内部面.
- The **boundary**边界 of a region is a subset of E.
- The **degree**度数 of a region is the number of edges on its boundary.
  - If an edge is shared by R_i, R_j, then it contributes 1 to $\deg(R_i)$, $\deg(R_j)$
  - If an edge is on the boundary of a single region R_i, then it contributes 2 to $\deg(R_i)$



- The plane is divided into 5 regions R_0, R_1, R_2, R_3, R_4
  - R_0 is the exterior region
  - R_1, R_2, R_3, R_4 are interior regions
- The boundary of R_1; $\deg(R_1) = 4$
- There are 4 edges on the boundary of R_4
  - $\deg(R_4) = 1 + 1 + 1 + 2 = 5$ because one of the edges contribute 2 to $\deg(R_4)$
- $\deg(R_0) = 11, \deg(R_1) = 4, \deg(R_2) = 3, \deg(R_3) = 3, \deg(R_4) = 5$

Euler's Formula

THEOREM: Let $G = (V, E)$ be a connected planar simple graph with e edges and v vertices. Let r be the number of regions in a planar representation of G. Then $r = e - v + 2$.

THEOREM: Let G be a planar simple graph with p connected components. Then $|V(G)| - |E(G)| + |R(G)| = p + 1$.

- Let $G_1, G_2, \ldots, G_p$ be the connected components of G.
  - By Euler's formula, $|R(G_i)| = |E(G)_i| - |V(G_i)| + 2$ for all $i \in [p]$
- $|V(G)| = |V(G_1)| + |V(G_2)| + \cdots + |V(G_p)|$
- $|E(G)| = |E(G_1)| + |E(G_2)| + \cdots + |E(G_p)|$
- $|R(G)| = |R(G_1)| + |R(G_2)| + \cdots + \left|R\left(G_p\right)\right| - p + 1$
- $|V(G)| - |E(G)| + |R(G)| = \sum_{i=1}^{p}(|V(G_i)| - |E(G_i)| + |R(G_i)|) - p + 1$
  $= 2p - p + 1 = p + 1$

Euler's Formula: Proof*

Proof of Euler's formula by induction on the number e of edges

- A simple connected planar graph with 0 edges has only one vertex and one face (unbounded). The relation $f = e - v + 2$ is satisfied.

- Suppose the relation is satisfied for all simple connected planar graphs with k edges.
Consider a simple connected planar graph G with $k + 1$ edges, $k \geq 0$. This graph can be seen as a simple connected planar graph G' with k edges (satisfying the relation by induction hypothesis) to which we add one edge. There are two ways to add an edge to G' to get G:
  - either the two endpoints of the edge are already in G': in this case, adding the edge adds also one face,
  - either only one of the endpoint is already in G': in this case, adding the edge adds also one vertex but no other face.

In both cases, the relation $f = e - v + 2$ is satisfied by G.

Application

THEOREM: Let G be a connected planar simple graph. If every region has degree $\geq l$ in a planar representation of G, then

then $|E(G)| \leq \frac{l}{l-2}(|V(G)| - 2)$.

- Let $R_1, \ldots R_t$ be the regions given by a planar representation of G $\quad //t = |R(G)|$
  - $\deg(R_i) \geq l$ for every $i = 1,2, \ldots, t$
- Let $r = \deg(R_1) + \deg(R_2) + \cdots + \deg(R_t)$. Then $r = 2|E(G)|$.
  - Every edge contributes 2 to r
    - If $e \in E$ is on the boundary of a single region R_i, then e contributes 2 to $\deg(R_i)$;
    - If $e \in E$ is shared by R_i and R_j, then e contributes 1 to $\deg(R_i)$ and 1 to $\deg(R_j)$;
- $2|E(G)| = r = \deg(R_1) + \deg(R_2) + \cdots + \deg(R_t) \geq lt = l|R(G)|$
- $|R(G)| = |E(G)| - |V(G)| + 2$
- Hence, $|E(G)| \leq \frac{l}{l-2}(|V(G)| - 2)$

Application

COROLLARY: Let G be a connected planar simple graph. If $|V(G)| \geq 3$, then $|E(G)| \leq 3|V(G)| - 6$.

- Every region has degree ≥ 3 in a planar representation of G
- Let $l = 3$ in the previous theorem
  - $|E(G)| \leq \frac{3}{3-2}(|V(G)| - 2) = 3|V(G)| - 6.$

EXAMPLE: The complete graph K_5 is not planar.

- $|E(K_5)| = \binom{5}{2} = 10, |V(K_5)| = 5, K_5$ is connected simple and of order ≥ 3
- $|E(K_5)| > 3|V(K_5)| - 6$
  - Hence, K_5 cannot be planar

COROLLARY: Let G be a connected planar simple graph. Then G has a vertex of degree ≤ 5.

- $|V(G)| < 3$: the statement is true.
- $|V(G)| \geq 3$: $\forall u \in V(G), \deg(u) \geq 6 \Rightarrow 2|E(G)| = \sum_u \deg(u) \geq 6|V(G)|$
  - G cannot be planar

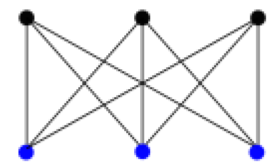
Application

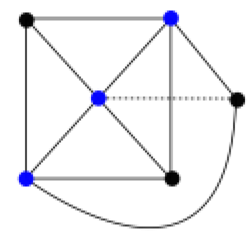
COROLLARY: Let G be a connected planar simple graph. If$|V(G)| \geq 3$ and there is no circuits of length 3 in G, then $|E(G)| \leq 2|V(G)| - 4$.

- Let $R_1, \dots R_t$ be the regions given by a planar representation of G $\quad //t = |R(G)|$
  - $\deg(R_i) \geq 4$ for every $i = 1,2, \dots, t$
- Hence, $|E(G)| \leq \frac{4}{4-2}(|V(G)| - 2) = 2|V(G)| - 4$

EXAMPLE: The complete bipartite graph $K_{3,3}$ is not planar.

- $\left|E\left(K_{3,3}\right)\right| = 3 \times 3 = 9, \left|V\left(K_{3,3}\right)\right| = 3 + 3 = 6 \geq 3$
- $K_{3,3}$ is connected, simple and of order ≥ 3.
- There is no circuits of length 3 in $K_{3,3}$
- $\left|E\left(K_{3,3}\right)\right| = 9 > 8 = 2\left|V\left(K_{3,3}\right)\right| - 4$
- Hence, $K_{3,3}$ cannot be planar



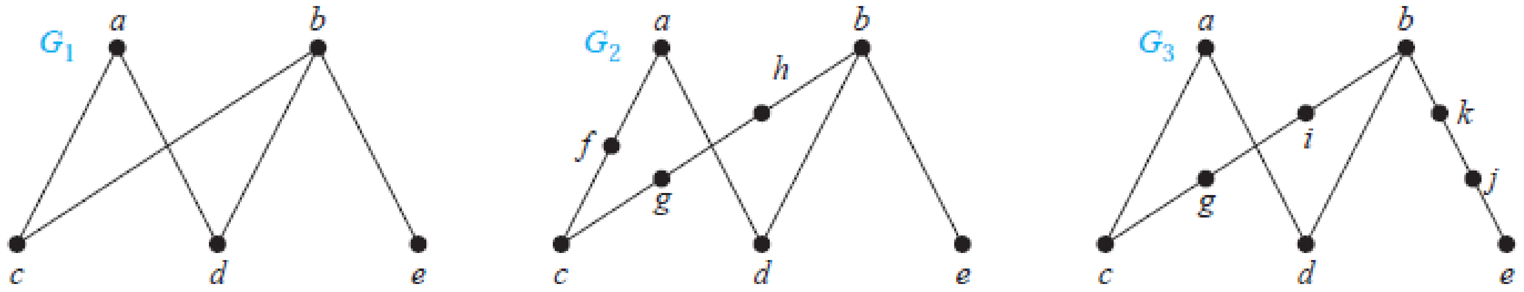


REMARKS: K_5 and $K_{3,3}$ are fundamental nonplanar graphs.

Homeomorphic

DEFINITION: Let $G = (V, E)$ be a graph and $\{u, v\} \in E$.

- **elementary subdivision**初等细分: $G' = (V \cup \{w\}, E - \{u, v\} + \{u, w\} + \{v, w\})$
- Two graphs are **homeomorphic**同胚的 if they can be obtained from the same graph via elementary subdivisions

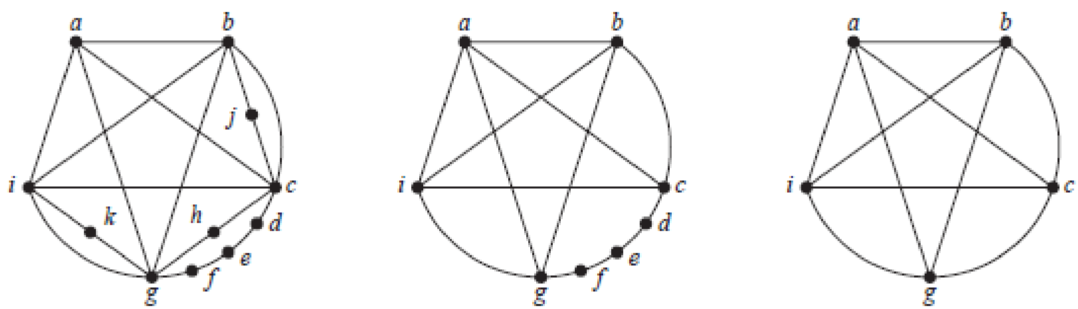


G_2 and G_3 are homeomorphic

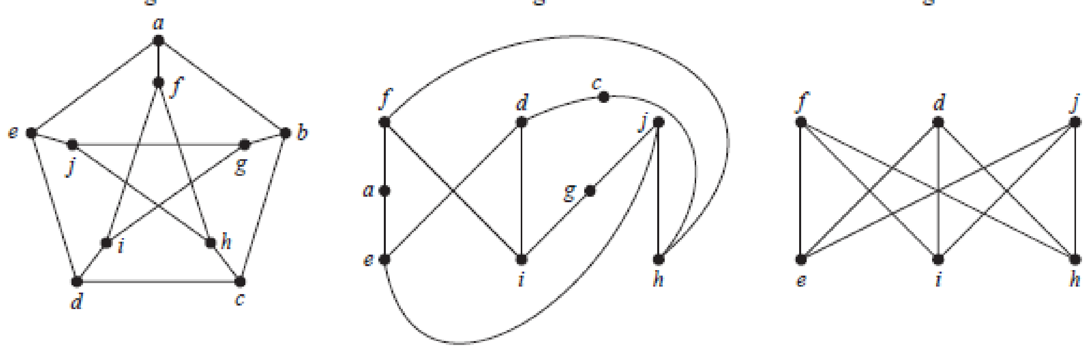
Kuratowski's Theorem

THEOREM: A graph G is nonplanar if and only if it has a subgraph homeomorphic to $K_{3,3}$ or K_5.

EXAMPLE: The following graph is nonplanar.



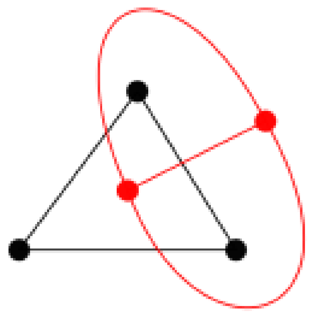
There is a subgraph homoemorphic to K_5

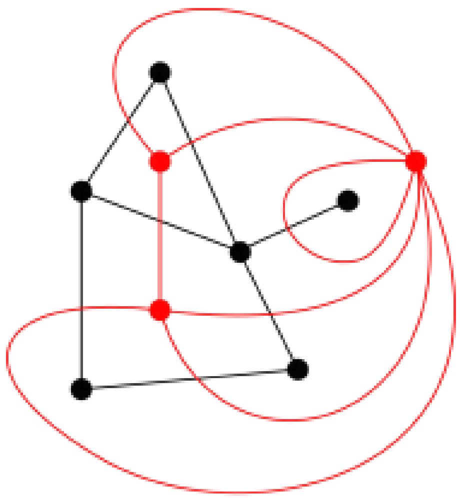


There is a subgraph homoemorphic to $K_{3,3}$

Dual Graph

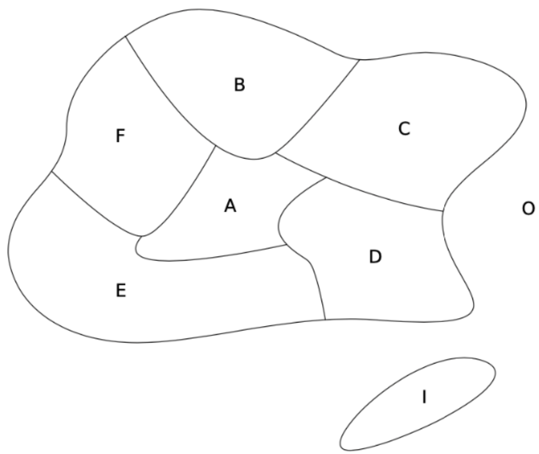
Let G be a planar graph and assume we take a planar representation of G that we denote also G. The **dual of** G is the graph G^* that has a vertex for each face of G and an edge connecting two vertices if the corresponding faces in G have a common edge in their boundary.

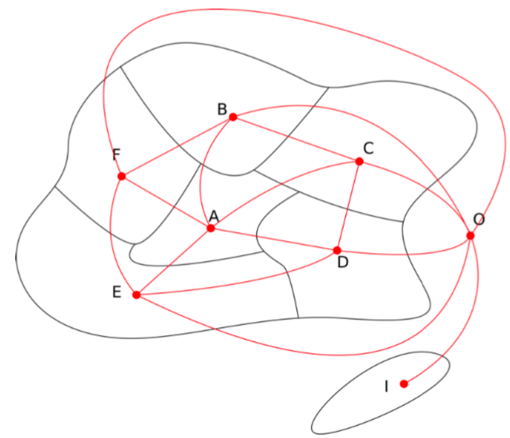


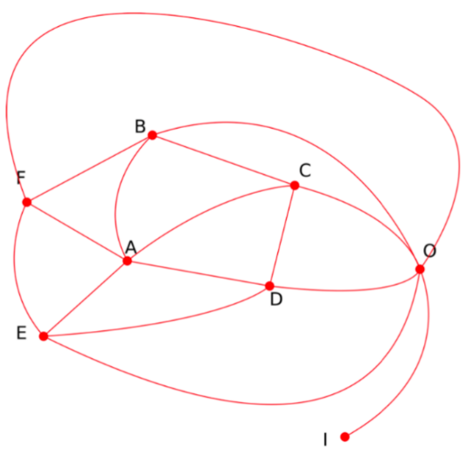


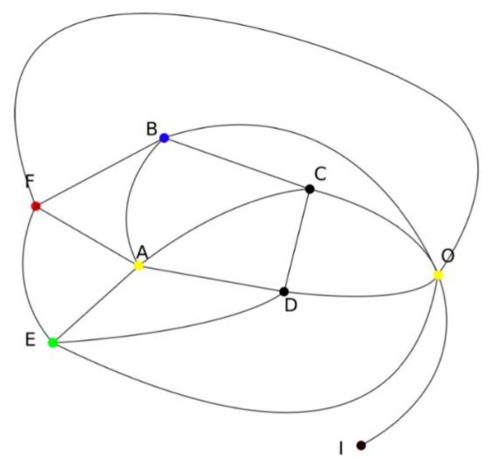
Remark: The dual of a planar simple graph is not necessarily simple.

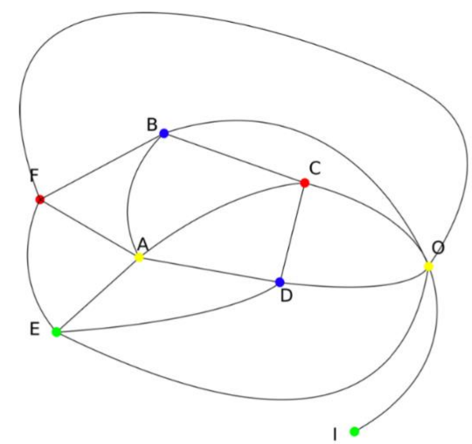
Coloring a Map









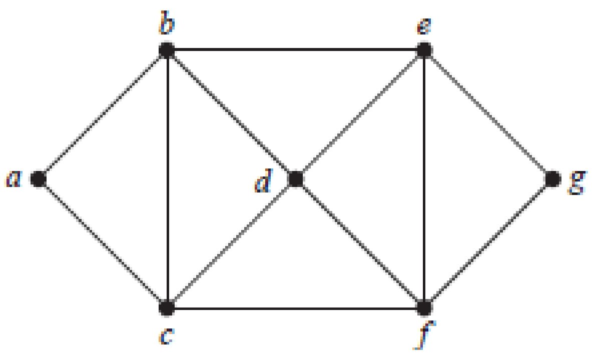


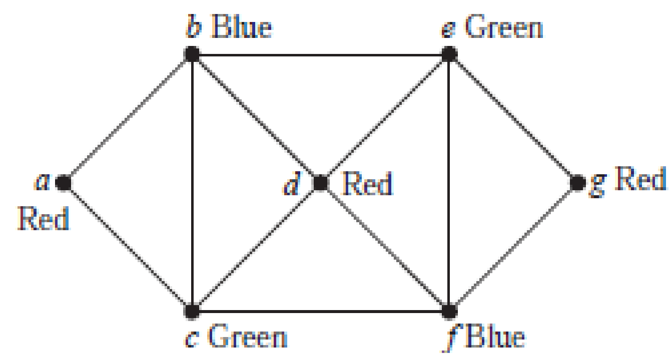
Coloring regions of the map $\Leftrightarrow$ Coloring vertices of the dual graph

Graph Coloring

DEFINITION: Let $G = (V, E)$ be a simple graph. A ***k*-coloring**$_{k\text{-着色}}$ of G is a map $f: V \to [k]$ such that $f(u) \neq f(v)$ whenever $\{u, v\} \in E$.

- **chromatic number ($\chi(G)$)**$_{\text{色数}}$**:** the least k s.t. G has a k-coloring.





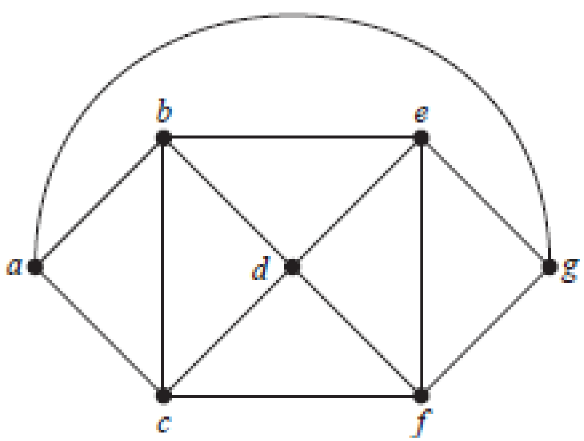
$\chi(G) = 3$

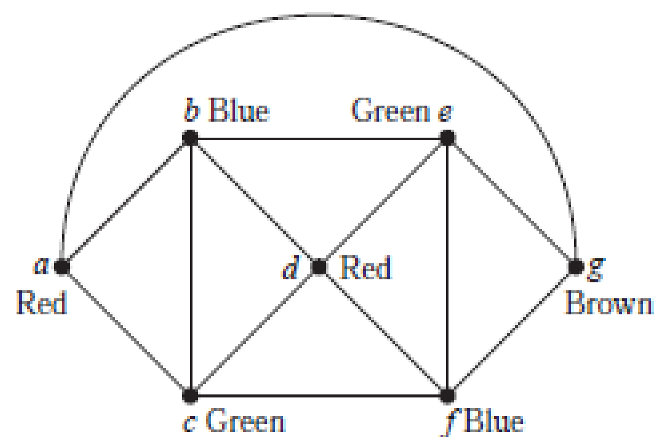
The chromatic number is at least 3 because *a; b; c* is a circuit of length 3

Graph Coloring

DEFINITION: Let $G = (V, E)$ be a simple graph. A ***k*-coloring**$_{k\text{-着色}}$ of G is a map $f: V \to [k]$ such that $f(u) \neq f(v)$ whenever $\{u, v\} \in E$.

- **chromatic number ($\chi(G)$)**$_{色数}$**:** the least k s.t. G has a k-coloring.



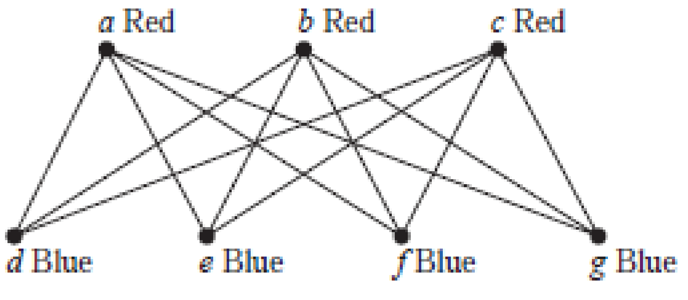


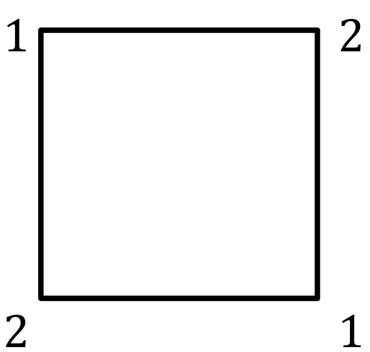
$\chi(G) = 4$

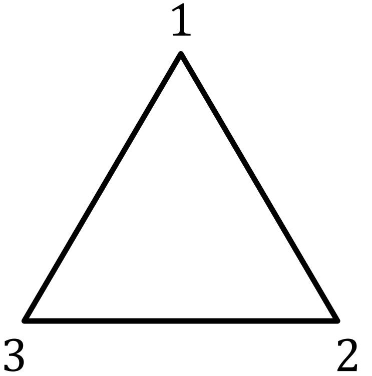
Graph Coloring

THEOREM: Let $G = (V, E)$ be a simple graph.

- $1 \leq \chi(G) \leq |V|$
- $\chi(G) = 1$ iff $E = \emptyset$
- $\chi(G) = 2$ iff G is bipartite and $|E| \geq 1$.
- $\chi(K_n) = n$ for every integer $n \geq 1$.
    - $\chi(G) \geq n$ if G has a subgraph isomorphic to K_n
- $\chi(C_n) = 2$ if $2|n$; $\chi(C_n) = 3$ if $2|(n-1)$; $(n \geq 3)$
- $\chi(G) \leq \Delta(G) + 1$, where $\Delta(G) = \max\{\deg(v) : v \in V\}$.

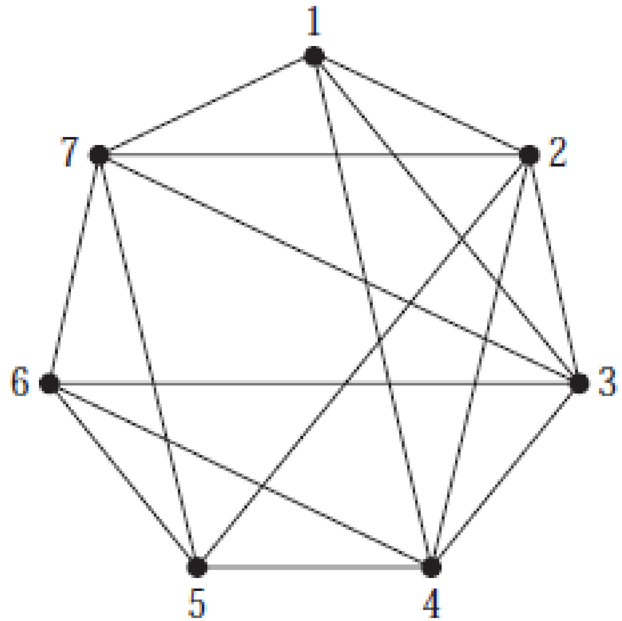


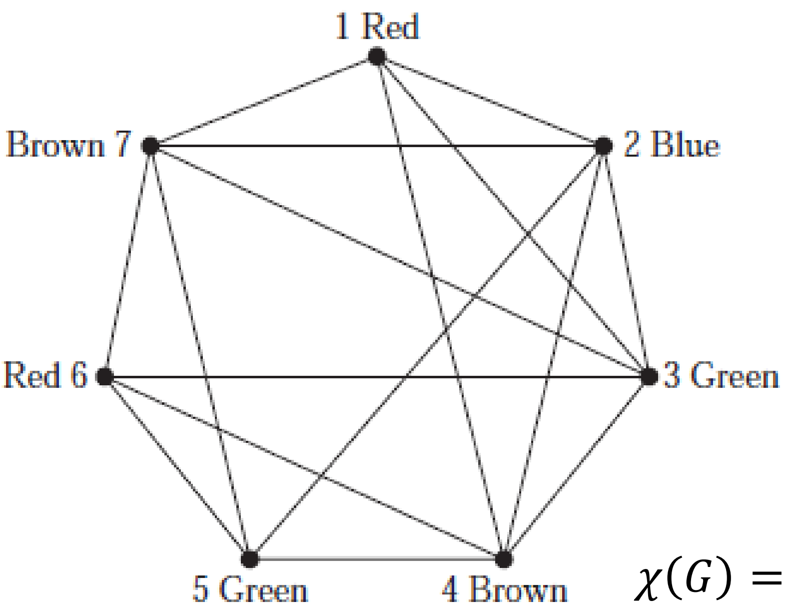




Application

PROBLEM: How can the final exams at a university be scheduled so that no student has two exams at the same time?





$\chi(G) = 4$

- There are 7 different courses, they are vertices of a graph.
- Two courses are adjacent if there is a student registered both courses.
- Choose time slots for the courses such that no two adjacent courses take place at the same time.
  - $\chi(G)$ time slots is needed.

$1 \leq \chi(G) \leq 7$

$1 \leq \chi(G) \leq \Delta(G) + 1 = 6$

$\chi(G) \geq 4$: G has a subgraph isomorphic to K_4

4-coloring Theorem

Theorem (Four coloring Theorem)

The chromatic number of a simple planar graph is no greater than 4.

Remarks: The proof of the 4-coloring Theorem depends on a computer. The two previous theorems are true for planar graphs only. A non planar graph can have an arbitrarily large chromatic number.

Discrete Mathematics: Lecture 28

Homeomorphic, Kuratowski's Theorem, Graph Coloring, Tree

Xuming He
Associate Professor

School of Information Science and Technology
ShanghaiTech University

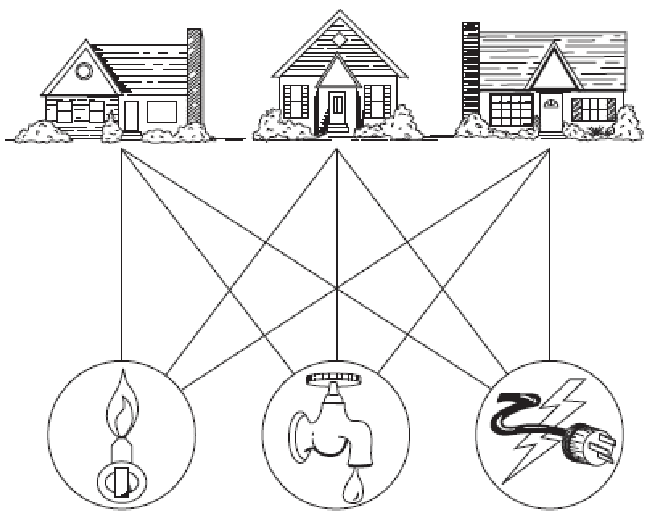
Spring Semester, 2022

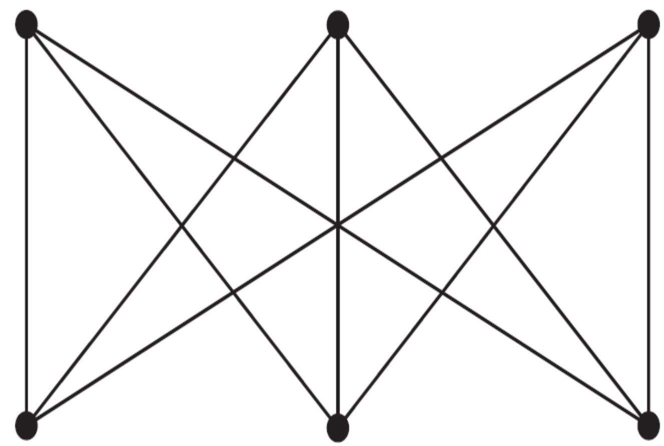
Notes by Prof. Liangfeng Zhang

Planar Graph

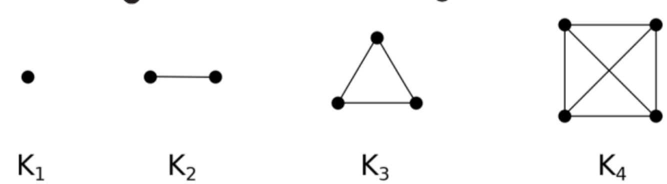
DEFINITION: Let $G = (V, E)$ be an undirected graph. G is called a **planar graph**平面图 if it can be drawn in the plane without any edges crossing.

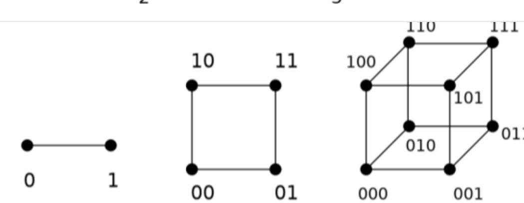
- Crossing of edges: an intersection other than endpoints (vertices)
- **planar representation**平面表示: a drawing w/o edge crossing; **nonplanar**非平面的





- K_1, K_2, K_3, K_4 are planar graphs
- $K_{1,n}, K_{2,n}$ are planar graphs
- $C_n\ (n \geq 3), W_n (n \geq 3)$ are planar graphs
- Q_1, Q_2, Q_3 are planar graphs

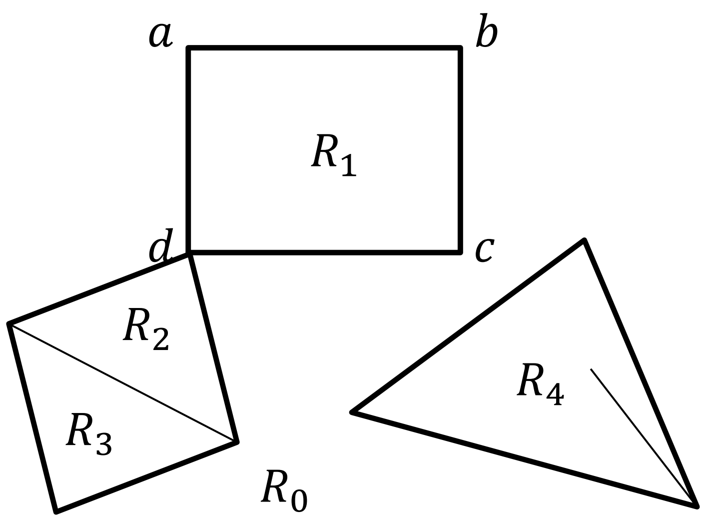




Regions

DEFINITION: Let $G = (V, E)$ be a planar graph. Then the plane is divided into several **regions**面 by the edges of G.

- The infinite region is **exterior region**外部面. The others are **interior regions**内部面.
- The **boundary**边界 of a region is a subset of E.
- The **degree**度数 of a region is the number of edges on its boundary.
  - If an edge is shared by R_i, R_j, then it contributes 1 to $\deg(R_i)$, $\deg(R_j)$
  - If an edge is on the boundary of a single region R_i, then it contributes 2 to $\deg(R_i)$



- The plane is divided into 5 regions R_0, R_1, R_2, R_3, R_4
  - R_0 is the exterior region
  - R_1, R_2, R_3, R_4 are interior regions
- The boundary of R_1; $\deg(R_1) = 4$
- There are 4 edges on the boundary of R_4
  - $\deg(R_4) = 1 + 1 + 1 + 2 = 5$ because one of the edges contribute 2 to $\deg(R_4)$
- $\deg(R_0) = 11, \deg(R_1) = 4, \deg(R_2) = 3, \deg(R_3) = 3, \deg(R_4) = 5$

Euler's Formula

THEOREM: Let $G = (V, E)$ be a connected planar simple graph with e edges and v vertices. Let r be the number of regions in a planar representation of G. Then $r = e - v + 2$.

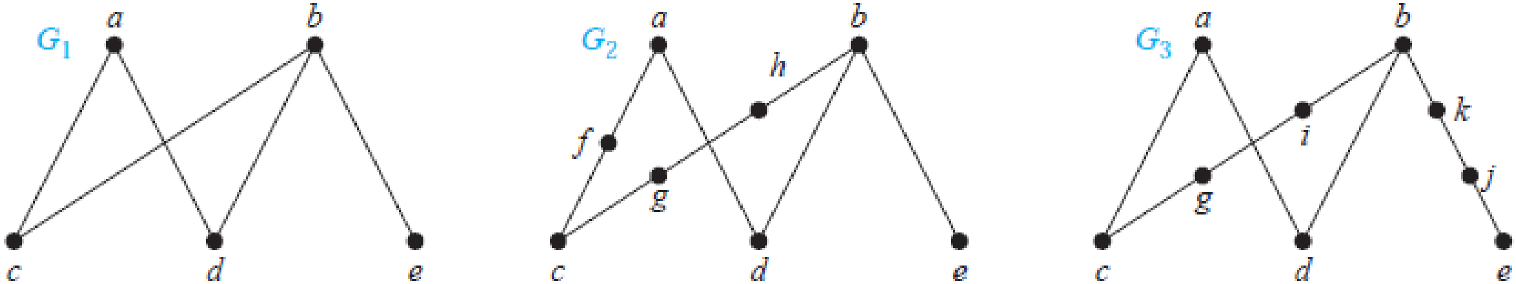
THEOREM: Let G be a planar simple graph with p connected components. Then $|V(G)| - |E(G)| + |R(G)| = p + 1$.

- Let $G_1, G_2, \dots, G_p$ be the connected components of G.
  - By Euler's formula, $|R(G_i)| = |E(G)_i| - |V(G_i)| + 2$ for all $i \in [p]$
- $|V(G)| = |V(G_1)| + |V(G_2)| + \cdots + |V(G_p)|$
- $|E(G)| = |E(G_1)| + |E(G_2)| + \cdots + |E(G_p)|$
- $|R(G)| = |R(G_1)| + |R(G_2)| + \cdots + \left|R\left(G_p\right)\right| - p + 1$
- $|V(G)| - |E(G)| + |R(G)| = \sum_{i=1}^{p}(|V(G_i)| - |E(G_i)| + |R(G_i)|) - p + 1$
  $= 2p - p + 1 = p + 1$

Homeomorphic

DEFINITION: Let $G = (V, E)$ be a graph and $\{u, v\} \in E$.

- **elementary subdivision**初等细分: $G' = (V \cup \{w\}, E - \{u, v\} + \{u, w\} + \{v, w\})$
- Two graphs are **homeomorphic**同胚的 if they can be obtained from the same graph via elementary subdivisions

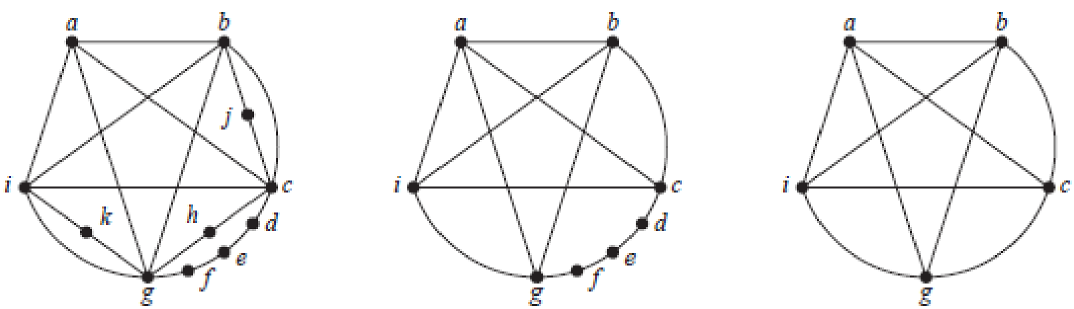


G_2 and G_3 are homeomorphic

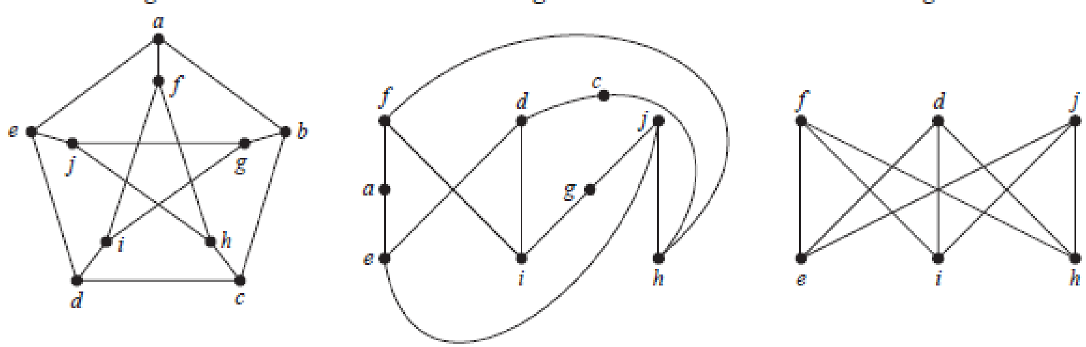
Kuratowski's Theorem

THEOREM: A graph G is nonplanar if and only if it has a subgraph homeomorphic to $K_{3,3}$ or K_5.

EXAMPLE: The following graph is nonplanar.



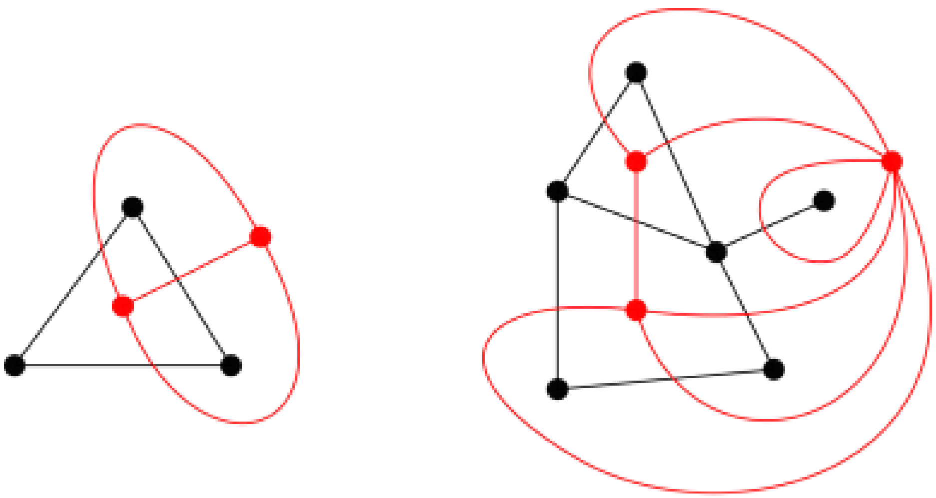
There is a subgraph homoemorphic to K_5



There is a subgraph homoemorphic to $K_{3,3}$

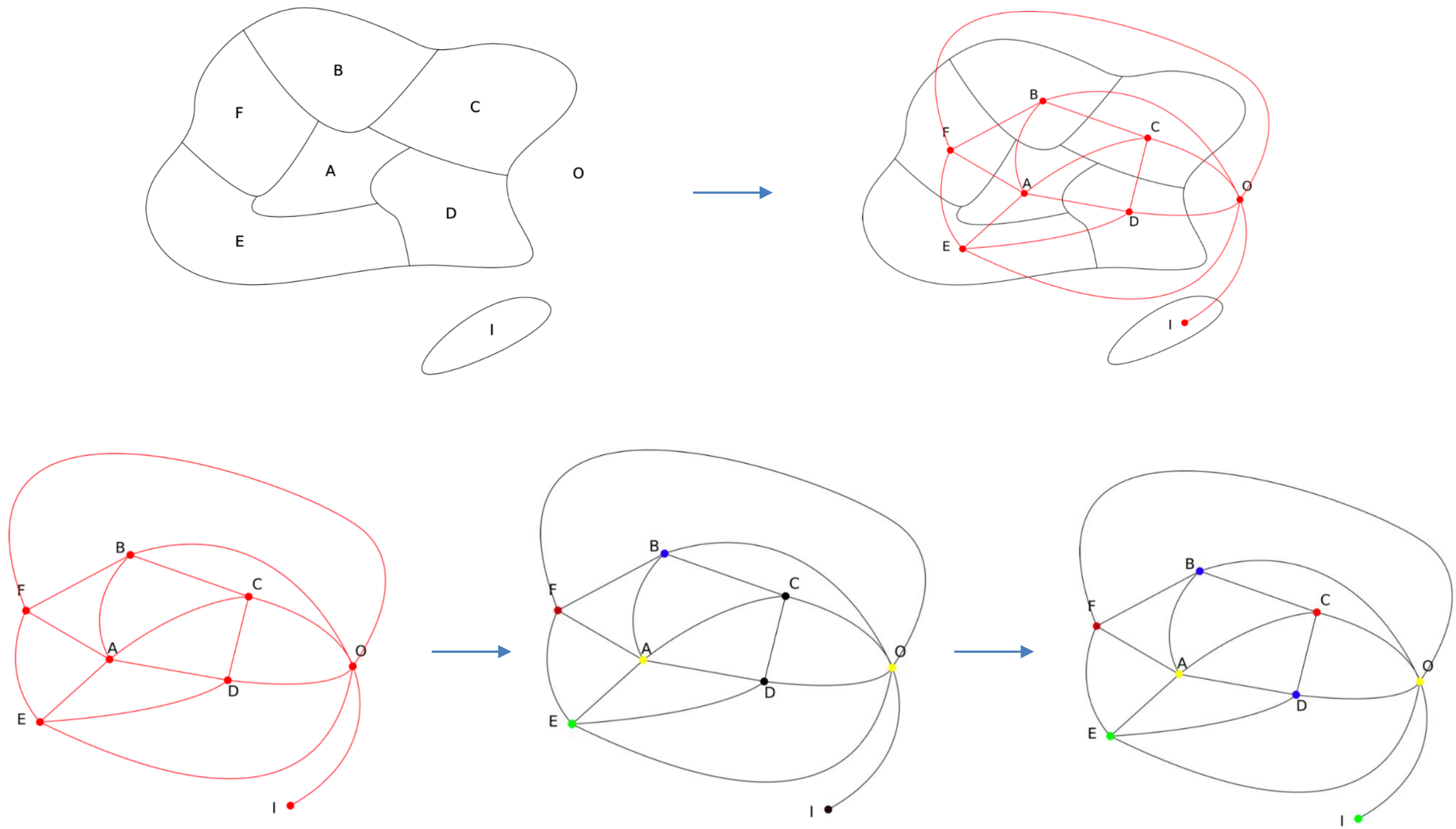
Dual Graph

Let G be a planar graph and assume we take a planar representation of G that we denote also G. The **dual of** G is the graph G^* that has a vertex for each face of G and an edge connecting two vertices if the corresponding faces in G have a common edge in their boundary.



Remark: The dual of a planar simple graph is not necessarily simple.

Coloring a Map

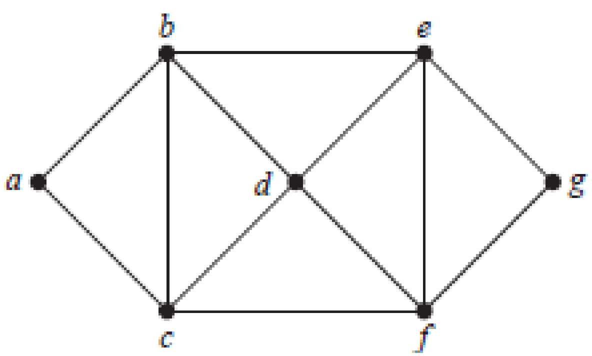


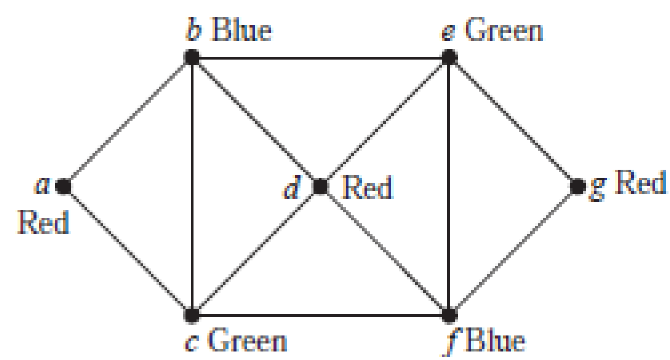
Coloring regions of the map ⇔ Coloring vertices of the dual graph

Graph Coloring

DEFINITION: Let $G = (V, E)$ be a simple graph. A ***k*-coloring**$_{k\text{-着色}}$ of G is a map $f: V \to [k]$ such that $f(u) \neq f(v)$ whenever $\{u, v\} \in E$.

- **chromatic number ($\chi(G)$)**$_{色数}$**:** the least k s.t. G has a k-coloring.





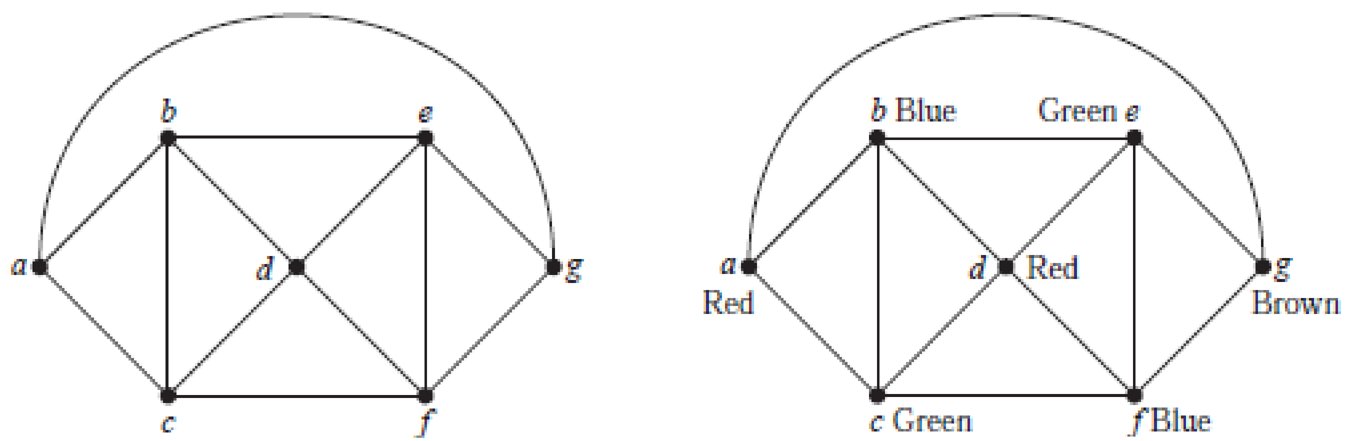
$\chi(G) = 3$

The chromatic number is at least 3 because *a; b; c* is a circuit of length 3

Graph Coloring

DEFINITION: Let $G = (V, E)$ be a simple graph. A ***k*-coloring**$_{k\text{-着色}}$ of G is a map $f: V \to [k]$ such that $f(u) \neq f(v)$ whenever $\{u, v\} \in E$.

- **chromatic number ($\chi(G)$)**$_{\text{色数}}$**:** the least k s.t. G has a k-coloring.

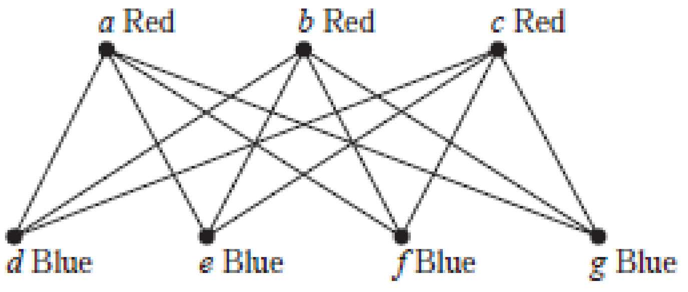


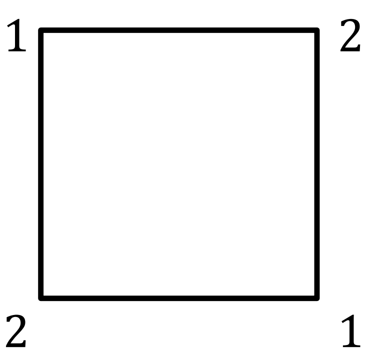
$\chi(G) = 4$

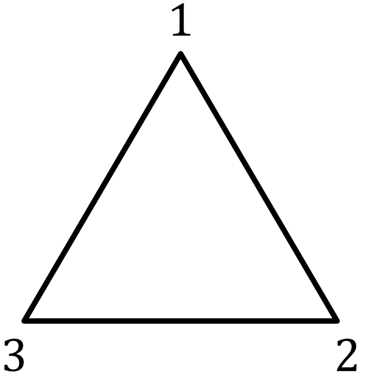
Graph Coloring

THEOREM: Let $G = (V, E)$ be a simple graph.

- $1 \leq \chi(G) \leq |V|$
- $\chi(G) = 1$ iff $E = \emptyset$
- $\chi(G) = 2$ iff G is bipartite and $|E| \geq 1$.
- $\chi(K_n) = n$ for every integer $n \geq 1$.
  - $\chi(G) \geq n$ if G has a subgraph isomorphic to K_n
- $\chi(C_n) = 2$ if $2|n$; $\chi(C_n) = 3$ if $2|(n-1)$; $(n \geq 3)$
- $\chi(G) \leq \Delta(G) + 1$, where $\Delta(G) = \max\{\deg(v) : v \in V\}$.

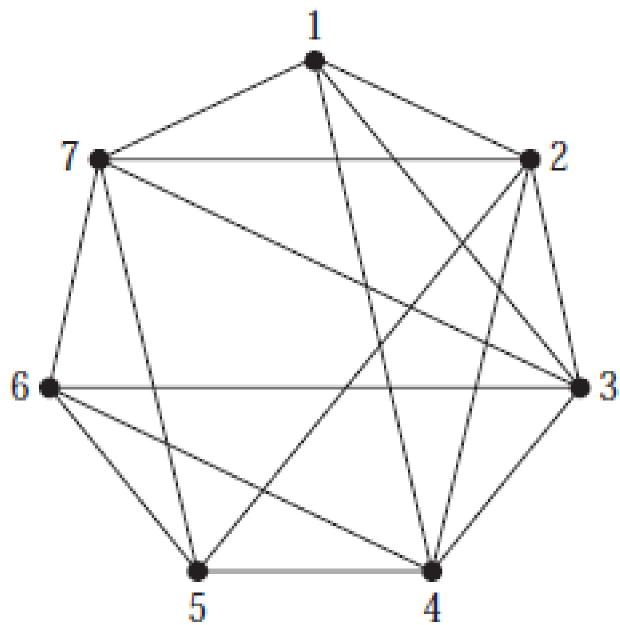


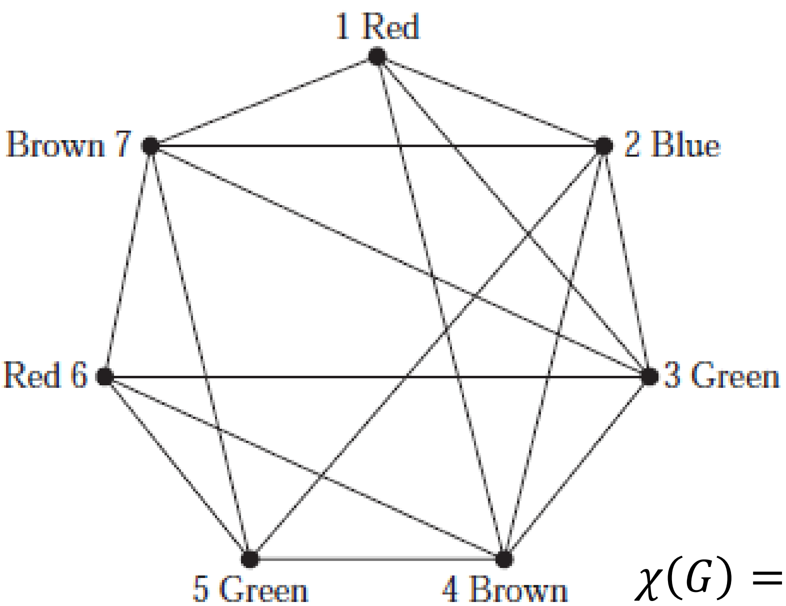




Application

PROBLEM: How can the final exams at a university be scheduled so that no student has two exams at the same time?





$\chi(G) = 4$

- There are 7 different courses, they are vertices of a graph.
- Two courses are adjacent if there is a student registered both courses.
- Choose time slots for the courses such that no two adjacent courses take place at the same time.
  - $\chi(G)$ time slots is needed.

$1 \leq \chi(G) \leq 7$

$1 \leq \chi(G) \leq \Delta(G) + 1 = 6$

$\chi(G) \geq 4$: G has a subgraph isomorphic to K_4

4-coloring Theorem

Theorem (Four coloring Theorem)

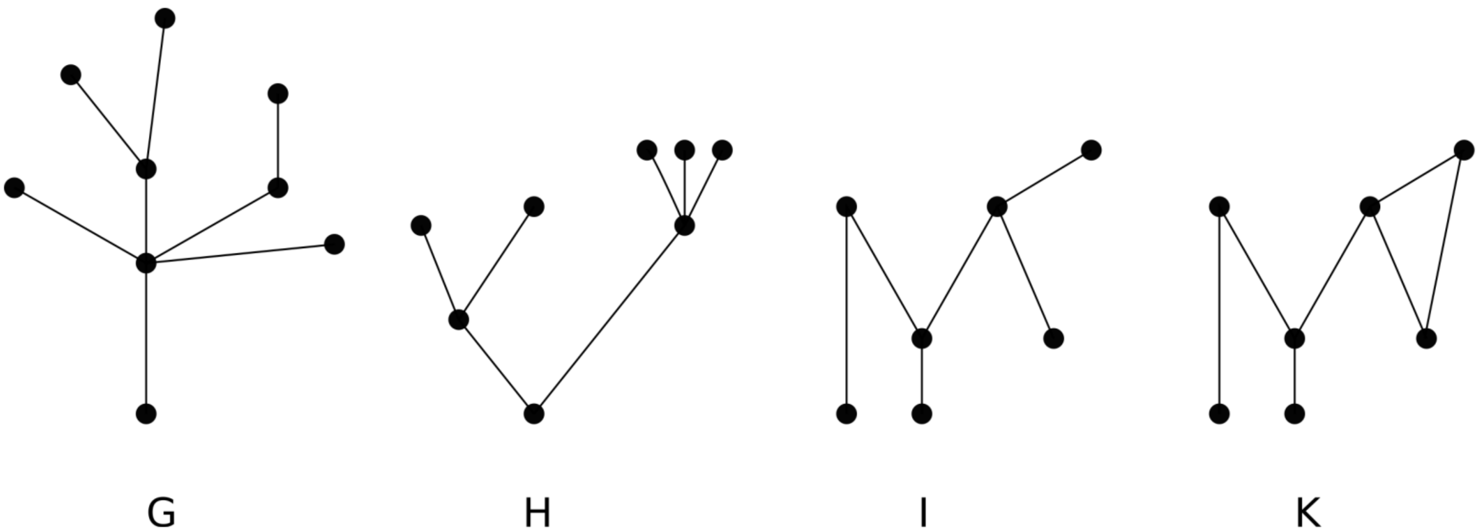
The chromatic number of a simple planar graph is no greater than 4.

Remarks: The proof of the 4-coloring Theorem depends on a computer. The two previous theorems are true for planar graphs only. A non planar graph can have an arbitrarily large chromatic number.

Tree

Definition

- A **tree** is a connected undirected graph with no simple circuits.
- A **forest** is an graph such that each of its connected components is a tree.



G, H, I are trees, but K is not a tree.

Characterization of Tree

Theorem

An undirected graph is a tree if and only if there is a unique simple path between any two of its vertices.

Proof: ($\Rightarrow$) Assume T is a tree and let u and v be two vertices. T is connected so there is a *simple path* P_1 from u to v. Assume there is a second simple path P_2 from u to v.

Claim: There is a simple circuit in T.
Let $u = x_0, x_1, \ldots, x_n = v$ denote the vertices of P_1 and $u = y_0, y_1, \ldots,$ $y_m = v$ the vertices of P_2.
P_1 and P_2 start at u but are not equal so must diverge at some point.

• If they diverge after one of them has ended, then the remaining part of the other path is a circuit from v to v.

Characterization of Tree

- Otherwise, we can assume

$$x_0 = y_0, x_1 = y_1, \ldots, x_i = y_i$$

and $x_{i+1} \neq y_{i+1}$.

We follow then $y_{i+1}, y_{i+2}, \ldots$ until we reach a vertex of P_1.
Then go back to x_i following P_1 forwards or backwards.

This gives a circuit which is simple because P_1 and P_2 are, and we stop using edges of P_2 as soon as we hit P_1.

($\Leftarrow$) Assume there is a unique simple path between any two vertices of the graph T. Then:

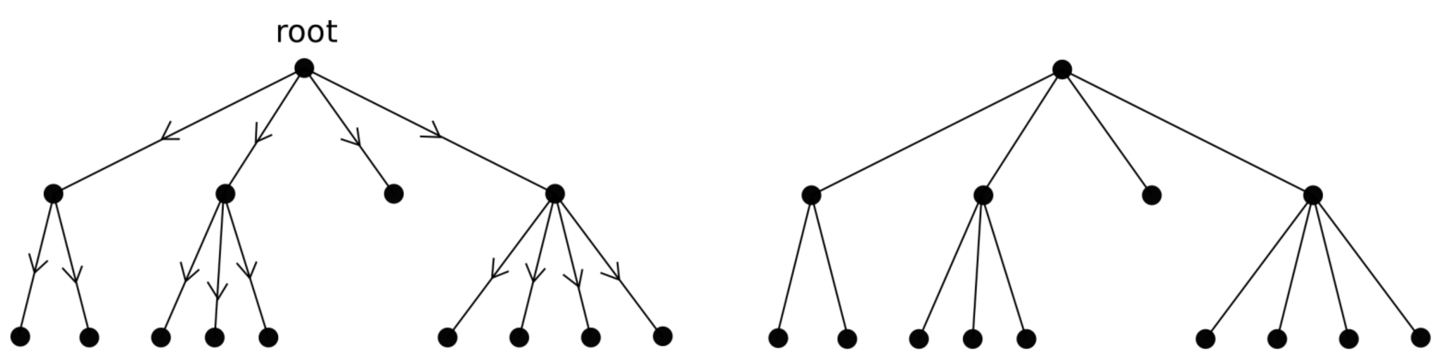
- T is connected (by definition)
- if T has a simple circuit containing the vertices x and y $\rightsquigarrow$ two simple paths between x and y. $\square$

Rooted Tree

Definition

A **rooted tree** is a tree in which one vertex has been designated as the root and every edge is directed away from the root.

Remarks: • A rooted tree is a directed graph.
• We usually draw a rooted tree with its root at the top of the graph.
• We usually omit the arrows on the edges to indicate the direction because it is uniquely determined by the choice of the root.
• Any non rooted tree can be changed to a rooted tree by choosing a vertex for the root.

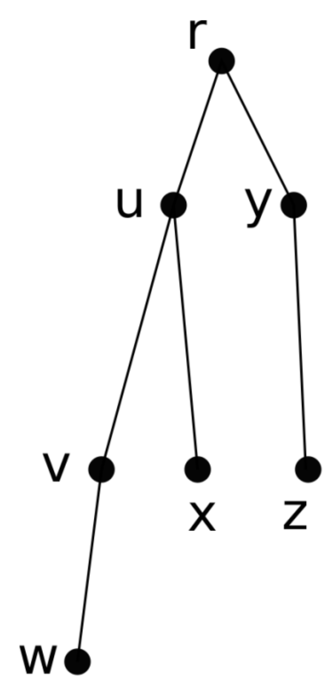


Rooted Tree

Definition

Let T be a rooted tree and v a vertex which is not the root. We call

- **parent** of v the *unique* vertex u such that there is an edge from u to v,
- **child** of v a vertex w such that there is an edge from v to w,
- **siblings** vertices with the same parent,
- **ancestors** of v all vertices in the path from the root to v,
- **descendants** of v all vertices that have v as an ancestor,
- **leaf** a vertex which has no children,
- **internal vertex** a vertex that has children,
- **subtree with** v **at its root** the subgraph of T consisting of v and its descendants and the edges incident to them.

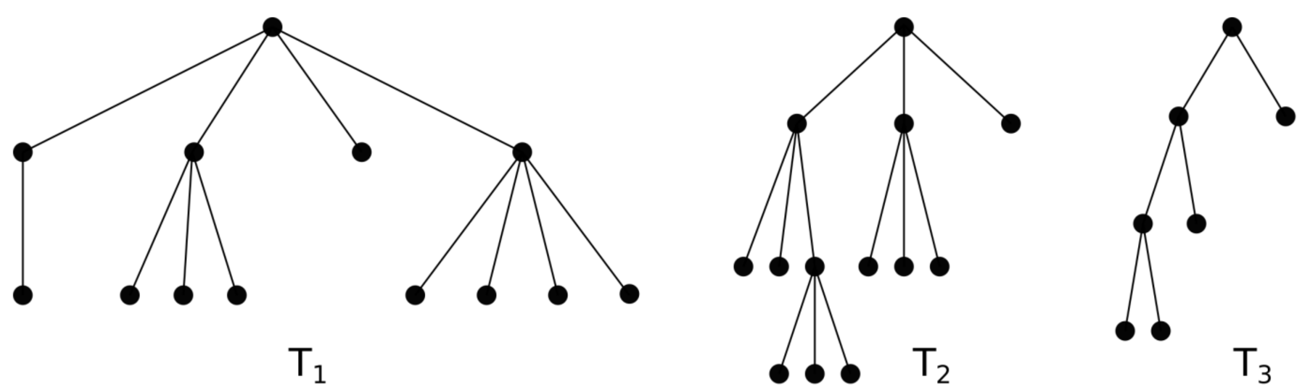


- r is the root
- v is child of u and parent of w
- v and x are siblings

Rooted Tree

Definition

• A rooted tree is called an m-**ary tree** if every internal vertex has no more than m children.
• A rooted tree is called a **full m-ary tree** if every internal vertex has exactly m children.
• An m-ary tree with $m = 2$ is called a **binary tree**. In this case if an internal vertex has two children, they are called **left child** and **right child**. The subtree rooted at the left (resp. right) child of a vertex is called the **left (resp. right) subtree** of this vertex.



T_1 is a 4-ary tree, T_2 a full 3-ary tree, T_3 a full binary tree.

Properties of Tree

Theorem

A tree with n vertices has $n-1$ edges.

Properties of Tree

Theorem

A tree with n vertices has $n-1$ edges.

Proof: By induction on the number of vertices.

- $n = 1$: A tree with one vertex has no edge.
- $k \rightsquigarrow k+1$: Assume every tree with k vertices has $k-1$ edges.

Let T be a tree with $k+1$ vertices, and v a leaf (which exists because the tree has a finite number of vertices).

Let T' be the tree obtained from T by removing v (and the edge incident to it). T' is a connected tree with k vertices $\Rightarrow$ it has $k-1$ edges by induction hypothesis.

$\Rightarrow$ T has $k+1$ vertices and k edges.

Properties of Tree

Tree = connected with no simple circuit (definition)

(1) connected
(2) no simple circuit
(3) $(n-1)$ edges (n=nb of vertices)

Previous theorem: $(1) + (2) \Rightarrow (3)$

We also have: $(1) + (3) \Rightarrow (2)$
$(2) + (3) \Rightarrow (1)$

Example: For what value of m, n the complete bipartite graph $K_{m,n}$ is a tree?

$K_{m,n}$ is connected, has $m+n$ vertices and $m \times n$ edges.
It is a tree if:

$$m \times n = m + n - 1 \Longleftrightarrow (n-1)m = n - 1$$

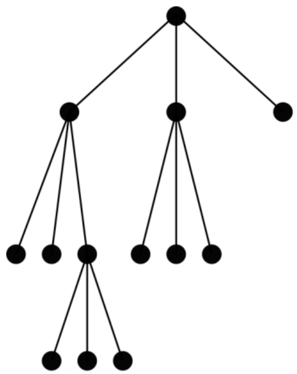
If $n \neq 1$: $m = 1$
If $n = 1$: $m \in \mathbb{N}^*$

Properties of Tree

Theorem

A full m-ary tree with i internal vertices contains $n = mi + 1$ vertices.

Proof: Each vertex (except the root) is the child of an internal vertex.
There are i internal vertices, each with m children
$\Rightarrow mi$ vertices + root $= mi + 1$ vertices



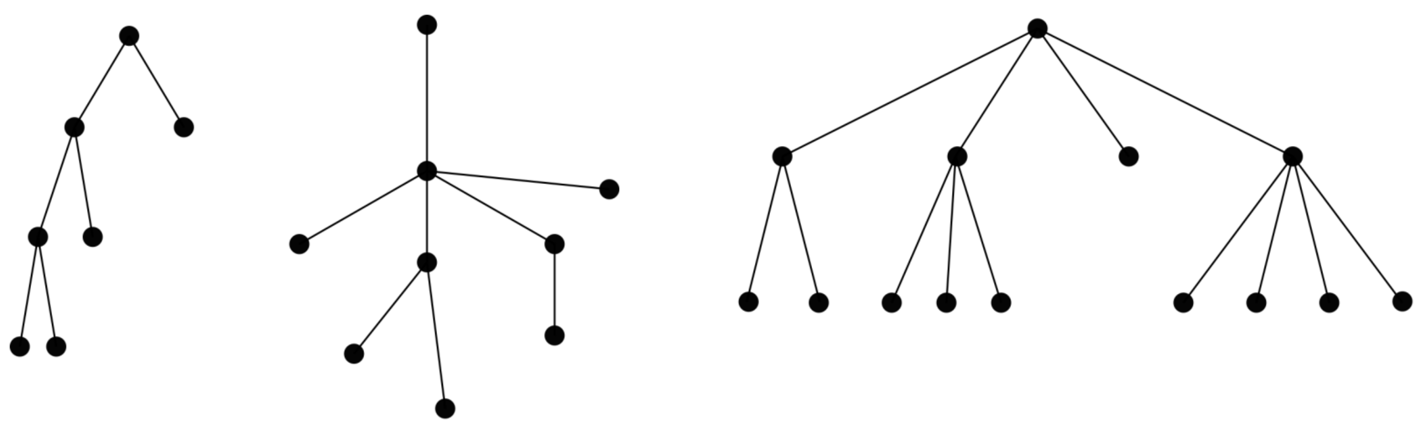
A full m-ary tree with

1. *n vertices has $i = (n-1)/m$ internal vertices and $\ell = \big((m-1)n+1\big)/m$ leaves,*
2. *i internal vertices has $n = mi + 1$ vertices and $\ell = (m-1)i + 1$ leaves,*
3. *ℓ leaves has $n = (m\ell - 1)/(m-1)$ vertices and $i = (\ell - 1)/(m-1)$ internal vertices.*

Balanced m-ary Tree

Definition

• The **level** of a vertex v in a rooted tree is the length of the unique path from the root to this vertex.
• The **height** of a rooted tree is the maximum of the levels of its vertices.
• A rooted m-ary tree of height h is **balanced** if all leaves are at levels h or $h-1$.



Balanced m-ary Tree

Theorem

There are at most m^h leaves in an m-ary tree of height h.

Proof: Induction again!

Corollary

If an m-ary tree of height h has l leaves, then $h \geq \lceil \log_m l \rceil$. If moreover the m-ary tree is full and balanced, then $h = \lceil \log_m l \rceil$.

Balanced m-ary Tree*

Theorem

There are at most m^h leaves in an m-ary tree of height h.

Proof: Induction again!

- An m-ary tree of height 1 consists of a root and its children (at most m) that are leaves. So the tree has at most $m^1 = m$ leaves.
- Assume all m-ary tree of height less or equal to h have at most m^h leaves.

Let T be an m-ary tree of height $h + 1$ and denote r its root.
Consider the subtrees rooted at the children of r. Each of them is an m-ary tree of height less or equal to h, so by inductive hypothesis they have at most m^h leaves.
There are at most m of such trees because r has at most m children.
So in total T has at most $m \times m^h$ leaves.

Discrete Mathematics: Lecture 29

Tree, Tree Traversals, Spanning Trees, DFS, BFS

Xuming He
Associate Professor

School of Information Science and Technology
ShanghaiTech University

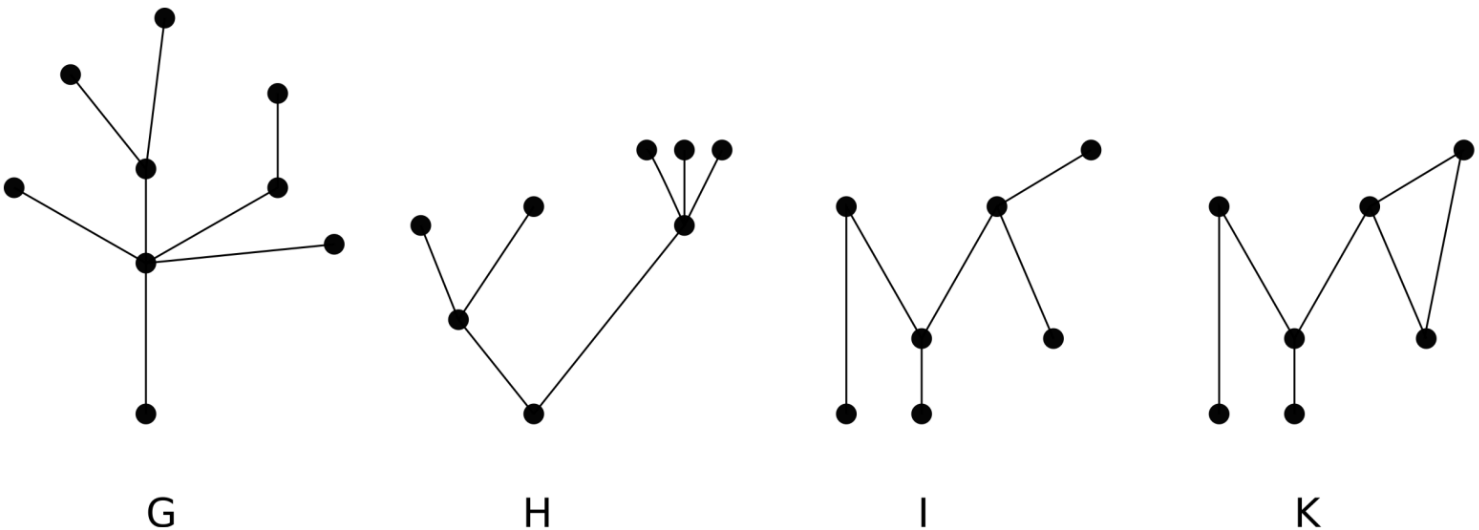
Spring Semester, 2022

Notes by Prof. Liangfeng Zhang

Tree

Definition

- A **tree** is a connected undirected graph with no simple circuits.
- A **forest** is an graph such that each of its connected components is a tree.



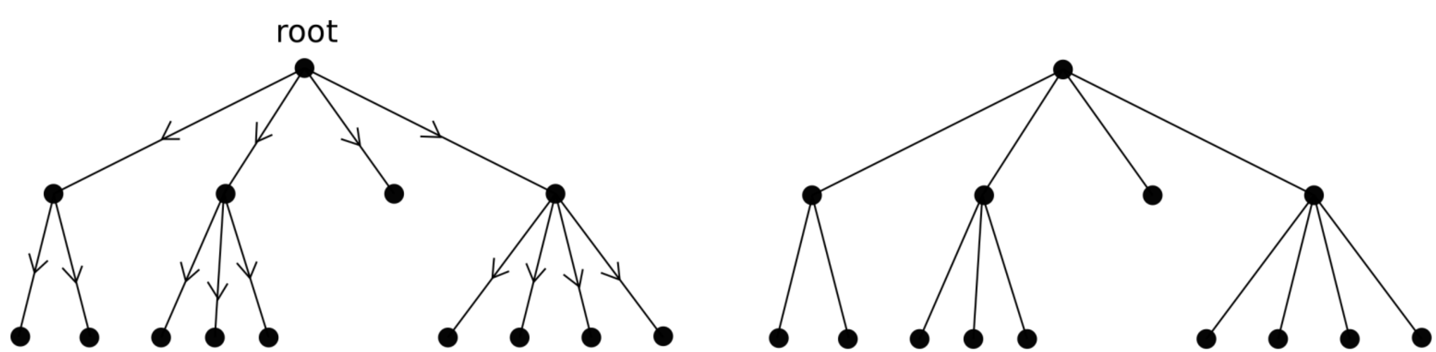
G, H, I are trees, but K is not a tree.

Rooted Tree

Definition

A **rooted tree** is a tree in which one vertex has been designated as the root and every edge is directed away from the root.

Remarks: • A rooted tree is a directed graph.
• We usually draw a rooted tree with its root at the top of the graph.
• We usually omit the arrows on the edges to indicate the direction because it is uniquely determined by the choice of the root.
• Any non rooted tree can be changed to a rooted tree by choosing a vertex for the root.



Properties of Tree

Tree = connected with no simple circuit (definition)

(1) connected
(2) no simple circuit
(3) $(n-1)$ edges (n=nb of vertices)

Previous theorem: (1) + (2) $\Rightarrow$ (3)

We also have: (1) + (3) $\Rightarrow$ (2)
(2) + (3) $\Rightarrow$ (1)

Example: For what value of m, n the complete bipartite graph $K_{m,n}$ is a tree?

$K_{m,n}$ is connected, has $m+n$ vertices and $m \times n$ edges.
It is a tree if:

$$m \times n = m + n - 1 \Longleftrightarrow (n-1)m = n - 1$$

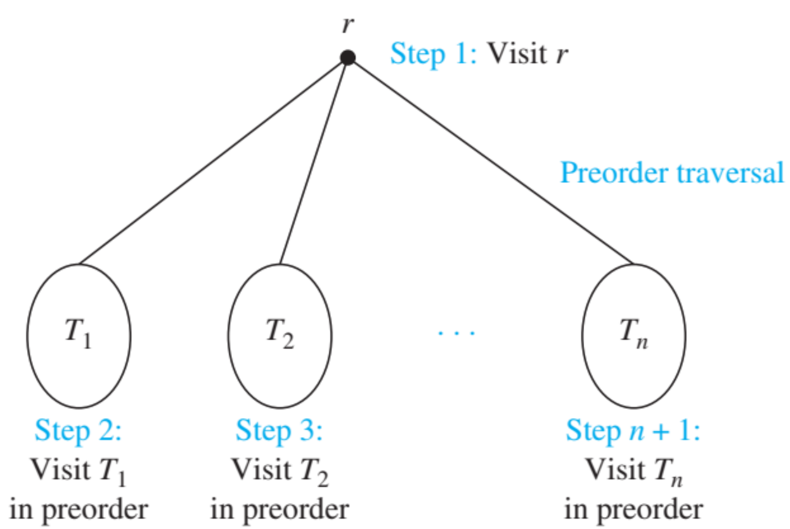
If $n \neq 1$: $m = 1$
If $n = 1$: $m \in \mathbb{N}^*$

Tree Traversals

Preorder traversal algorithm

Recursive definition: Let T be a rooted tree with root r

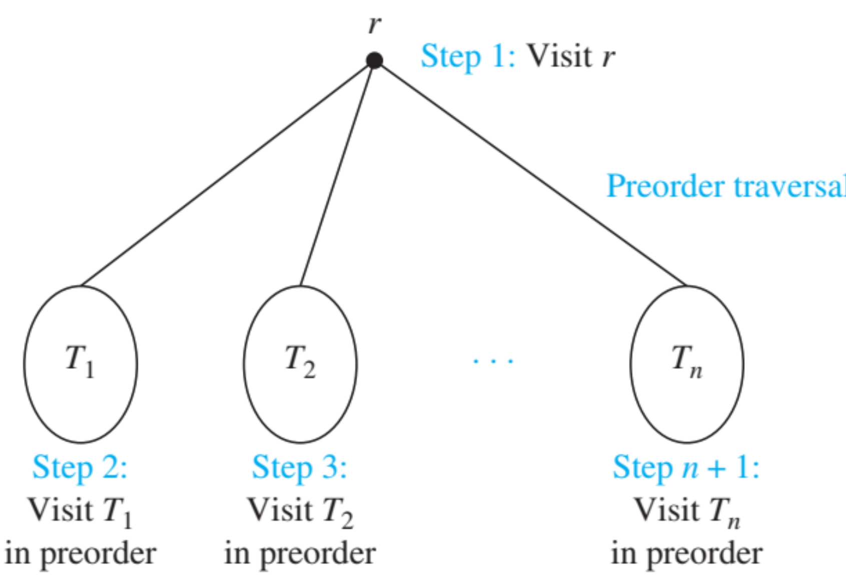
- if T consists only on r: r is the preorder traversal of T.
- otherwise, denote by $T_1, \ldots, T_n$ the subtrees rooted at the children of r, from left to right.
  The preorder traversal of T begins by visiting r, then traverses T_1 in preorder, then T_2 in preorder,..., and finally T_n in preorder.



Tree Traversals

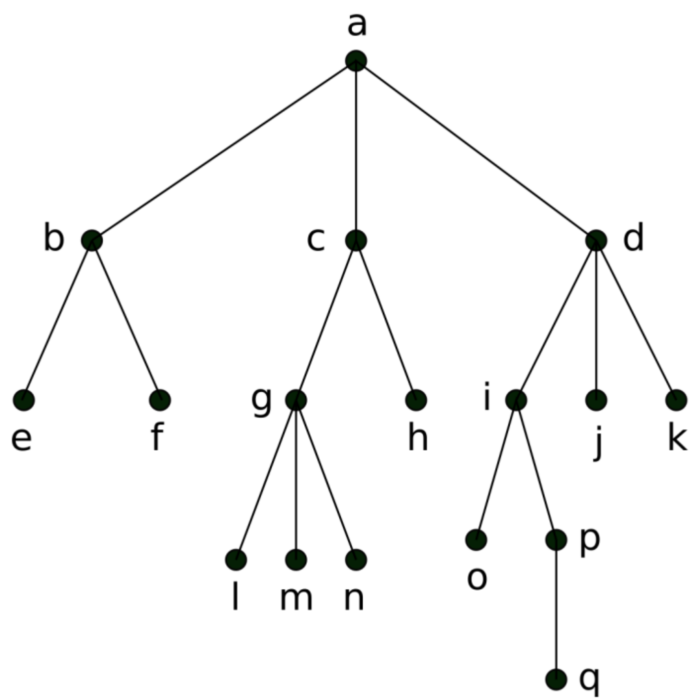
Recursive algorithm:

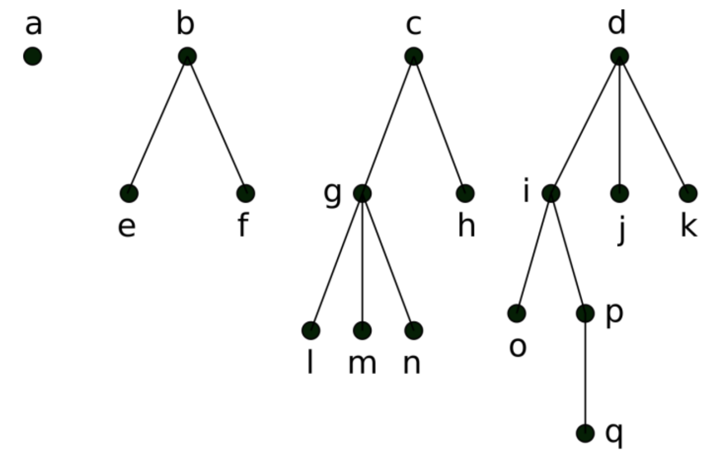
preorder(T: ordered rooted tree)
r:=root of T
list r (add r in the preorder list of the vertices of T)
for each child c of r from left to right
    $T(c)$:= subtree of T with c as its root
    preorder(T(c))

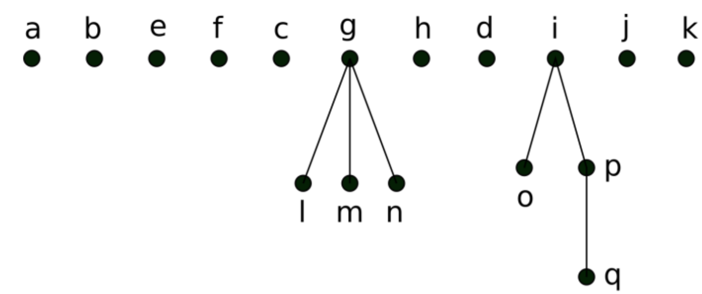


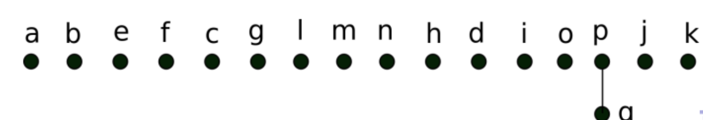
Tree Traversals

Preorder traversal algorithm







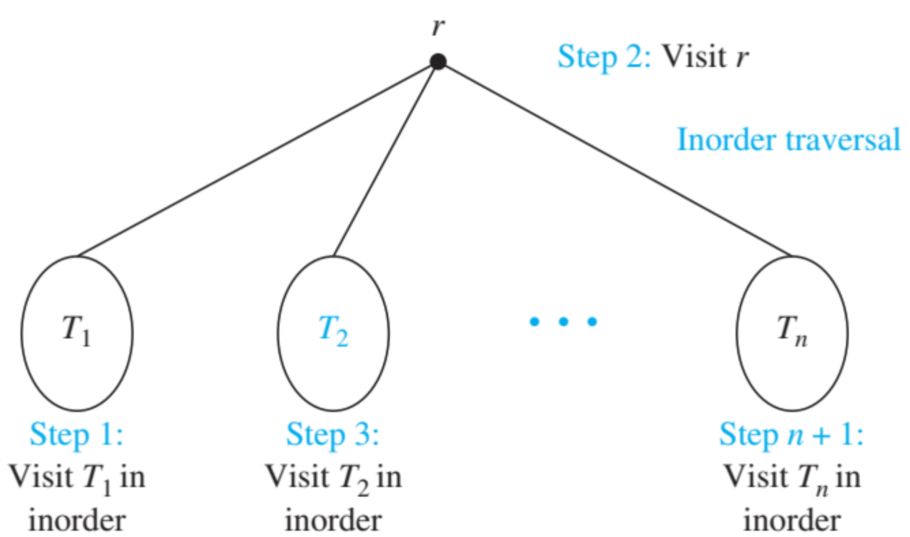


Tree Traversals

Inorder traversal algorithm

Recursive definition: Let T be a rooted tree with root r

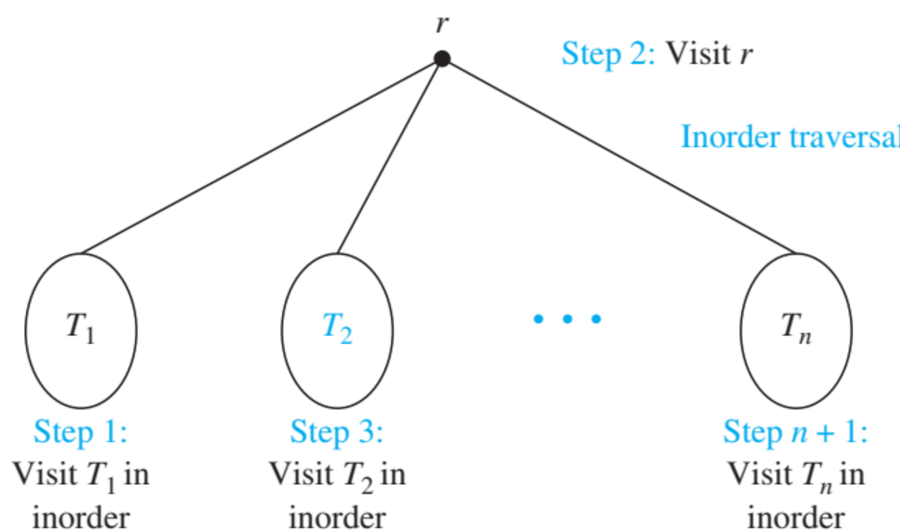
- if T consists only on r: r is the inorder traversal of T.
- otherwise, denote by $T_1, \ldots, T_n$ the subtrees rooted at the children of r, from left to right.
  The inorder traversal of T begins by traversing T_1 in inorder, then visiting r, then traversing T_2 in inorder, then T_3 in inorder,..., and finally T_n in inorder.



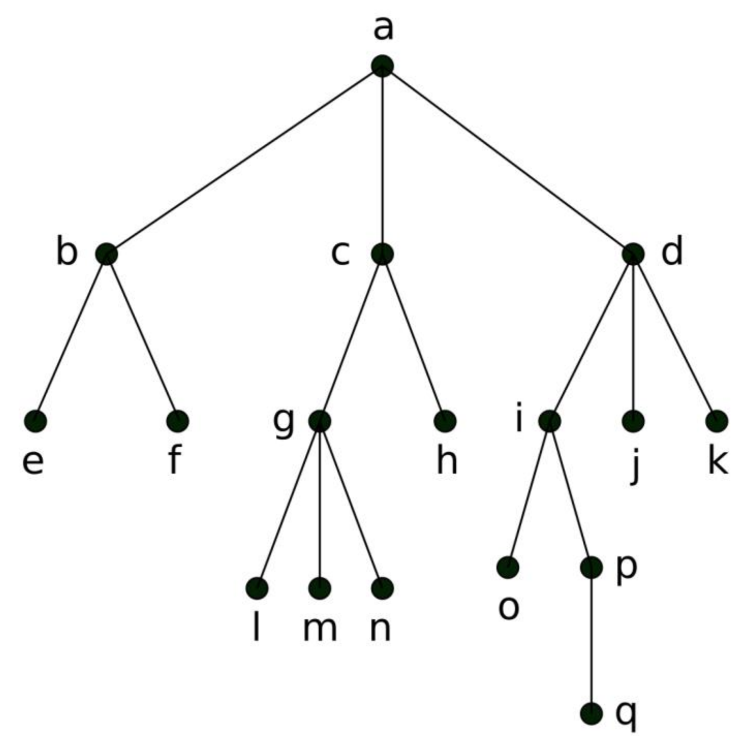
Tree Traversals

Recursive algorithm:

inorder(T: ordered rooted tree)
r:=root of T
if r is a leaf **then** list r
else l := first child of r from left to right
    $T(l)$:= subtree of T with l as its root
    inorder(T(l))
    list r
    for each child c of r from left to right except l
        $T(c)$:= subtree of T with c as its root
        inorder(T(c))



Tree Traversals



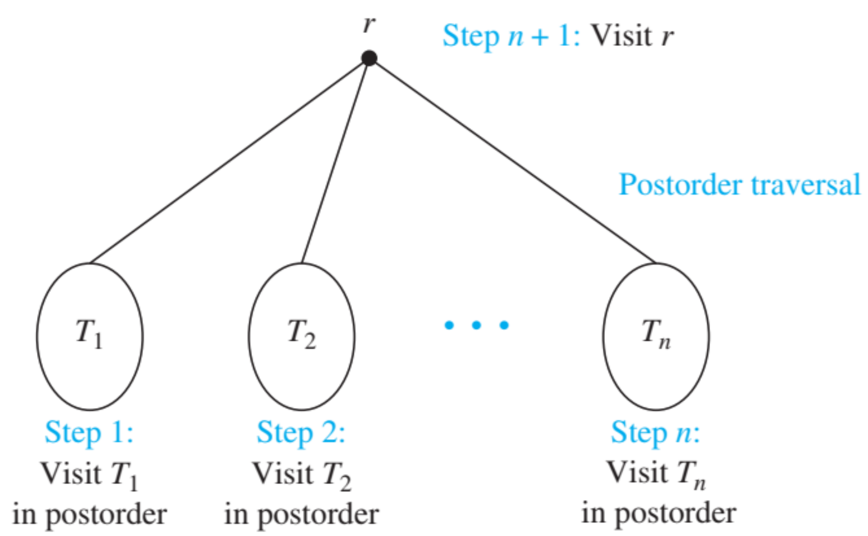
Inorder traversal: e, b, f, a, l, g, m, n, c, h, o, i, q, p, d, j, k

Tree Traversals

Postorder traversal algorithm

Recursive definition: Let T be a rooted tree with root r

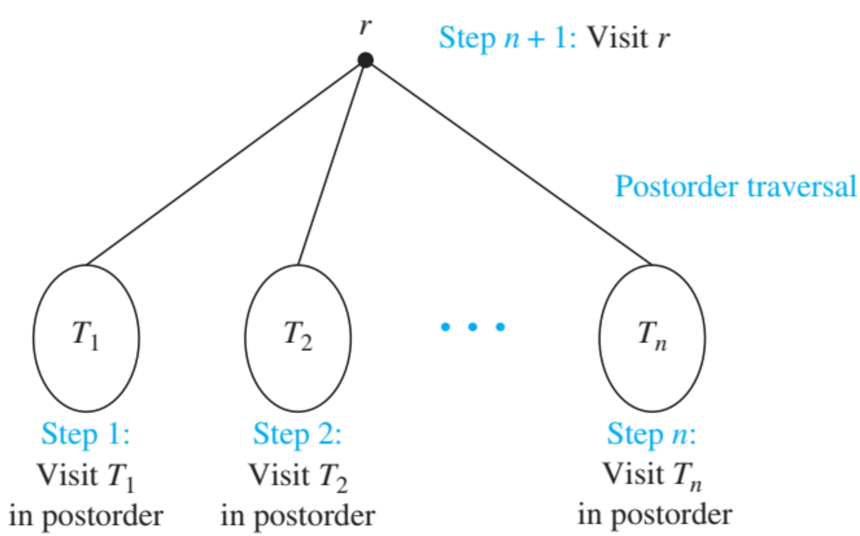
- if T consists only on r: r is the postorder traversal of T.
- otherwise, denote by $T_1, \ldots, T_n$ the subtrees rooted at the children of r, from left to right.
  The postorder traversal of T begins by traversing T_1 in postorder, then T_2 in postorder,..., then T_n in postorder, and ends by visiting the root r.



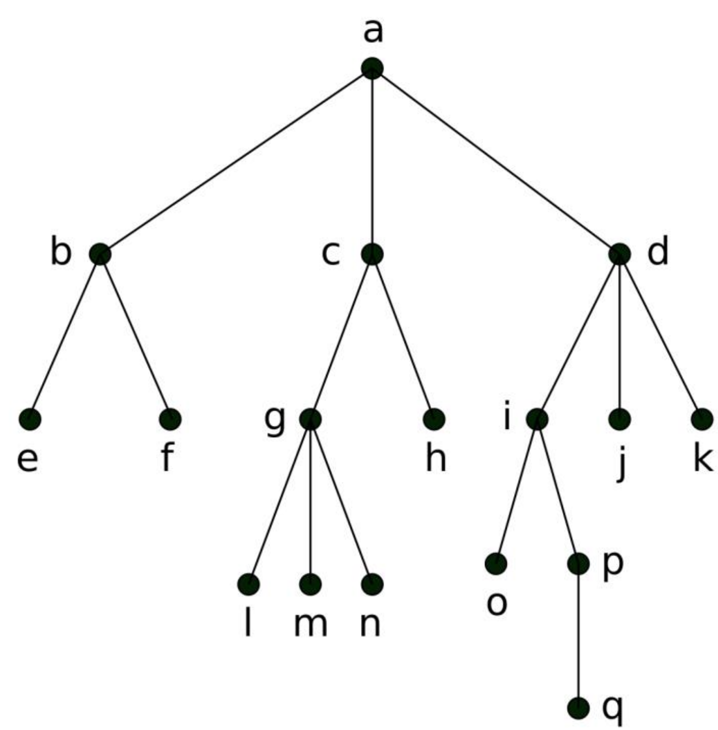
Tree Traversals

Recursive algorithm:

postorder(T: ordered rooted tree)
r:=root of T
for each child c of r from left to right
    $T(c)$:= subtree of T with c as its root
    postorder(T(c))
list r



Tree Traversals



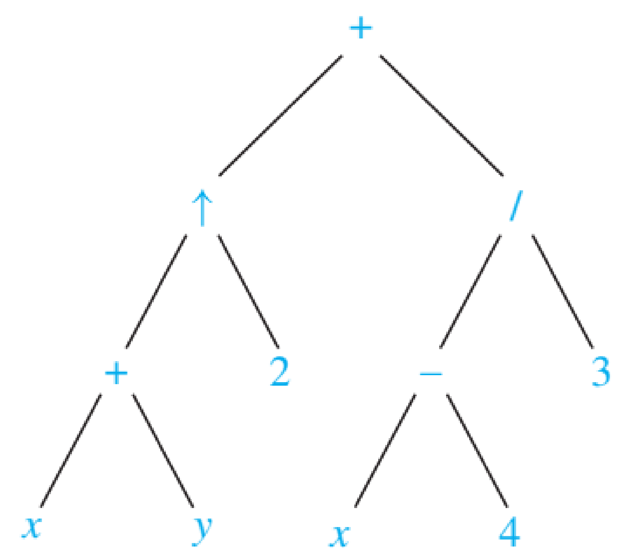
Postorder traversal: e, f, b, l, m, n, g, h, c, o, q, p, i, j, k, d, a

Infix, Prefix, Postfix Notation

Goal: Using ordered rooted trees to represent arithmetic expressions or compound propositions.

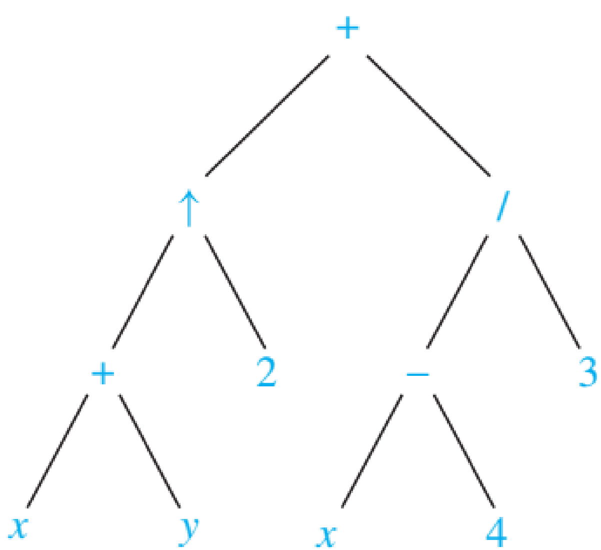
- leaves: numbers or variables,
- internal vertices: operations, where each operation operates on its left and right subtrees in that order (or its only subtree if it is a unary operation).

$$((x + y) \uparrow 2) + ((x - 4)/3)$$



Infix, Prefix, Postfix Notation

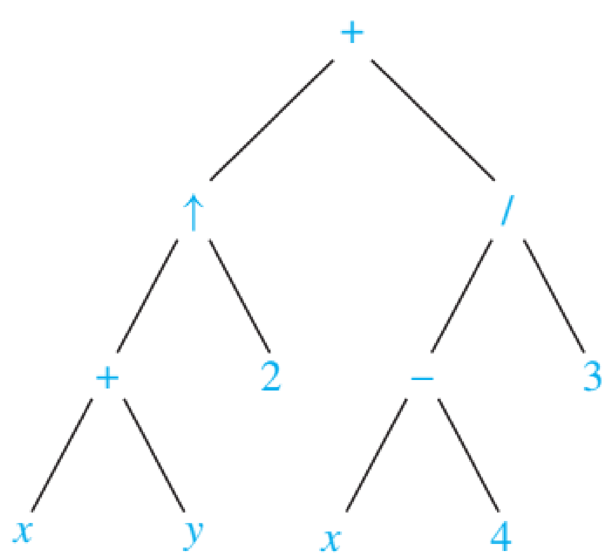
$\Rightarrow$ An inorder traversal of a binary tree representing an expression produces the original expression with the elements and operations in the same order as they originally appear, except for unary operation.
But: inorder traversals give ambiguous expressions $\Rightarrow$ need to include parentheses $\Rightarrow$ **infix form** (fully parenthesized)



$$((x + y) \uparrow 2) + ((x - 4)/3)$$

Infix, Prefix, Postfix Notation

The **prefix form (Polish notation)** of an expression is obtained by traversing its corresponding rooted tree in preorder.
An expression in prefix form (where each operation has a specified number of operands) is unambiguous.

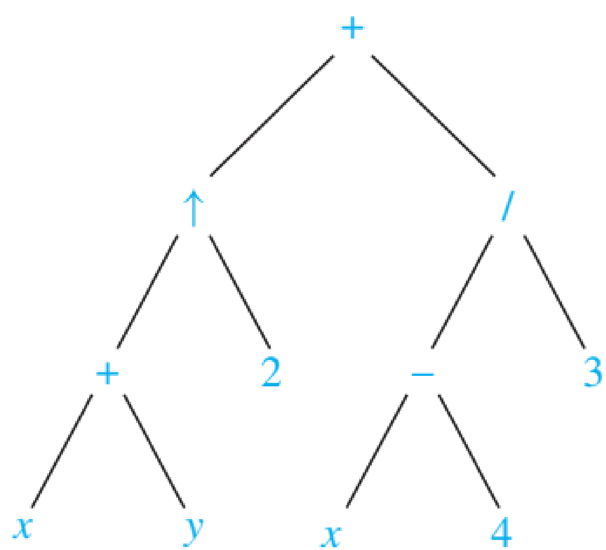


$$+ \uparrow + x\ y\ 2\ /\ -\ x\ 4\ 3$$

- Evaluate an expression in prefix form by working from right to left.
- When we encounter an operator, we perform the corresponding operation with the two operands immediately to the right of this operand.

Infix, Prefix, Postfix Notation

The **postfix form (reverse Polish notation)** of an expression is obtained by traversing its corresponding rooted tree in postorder. An expression in postfix form (where each operation has a specified number of operands) is unambiguous.



$$x\ y + 2 \uparrow x\ 4 - 3\ / +$$

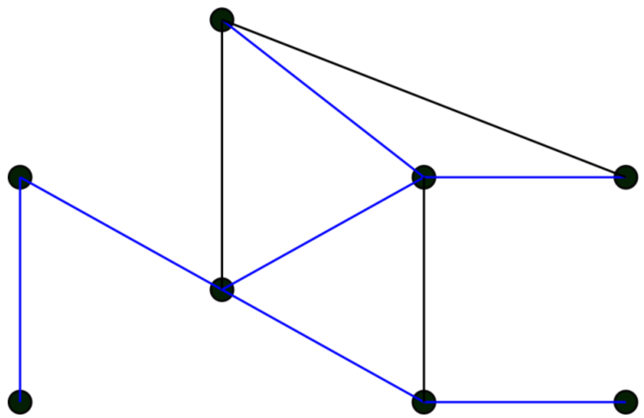
- Work from left to right, carrying out operations whenever an operator follows two operands.
- After an operation is carried out, the result of this operation becomes a new operand.

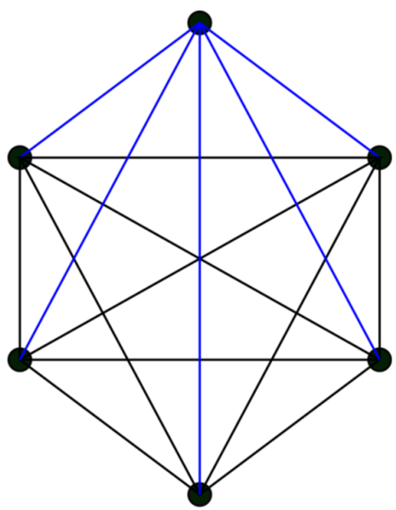
Spanning Trees

Definition

Let G be a simple graph. A **spanning tree** of G is a subgraph of G that is a tree containing every vertex of G.

Example:



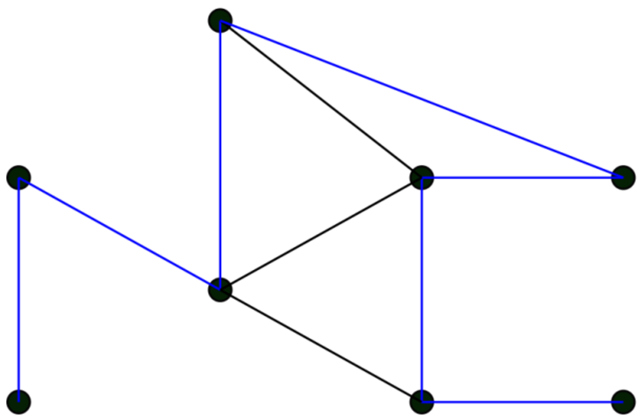


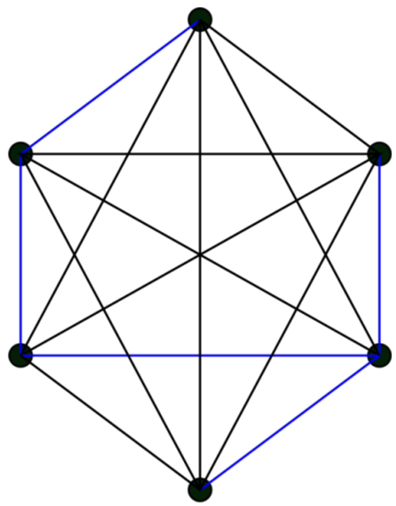
Spanning Trees

Definition

Let G be a simple graph. A **spanning tree** of G is a subgraph of G that is a tree containing every vertex of G.

Example:





Spanning Trees

Theorem

A simple graph is connected if and only if it has a spanning tree.

Proof:

"$\Leftarrow$" Assume G is a simple graph admitting a spanning tree T:

- T subgraph of G containing all vertices of G,
- by definition of tree, their is a path between any two vertices of T

So their is a path between any two vertices of G.

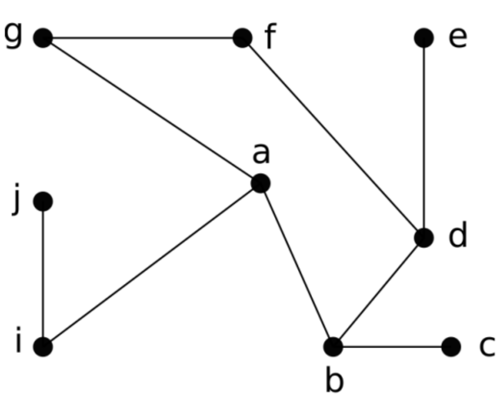
"$\Rightarrow$" Assume G is a simple connected graph.
If it is not a tree, it contains a circuit. Denote G' the subgraph of G obtained by removing one edge of the circuit with endpoints u and v.
There is still a path from u to $v \Rightarrow G'$ is connected.
If G' is not a tree, it contains a circuit, and again take a subgraph removing one edge of the circuit.
Repeat this process until there is no more circuit.
The graph obtained is connected and has no circuit, it is a spanning tree.

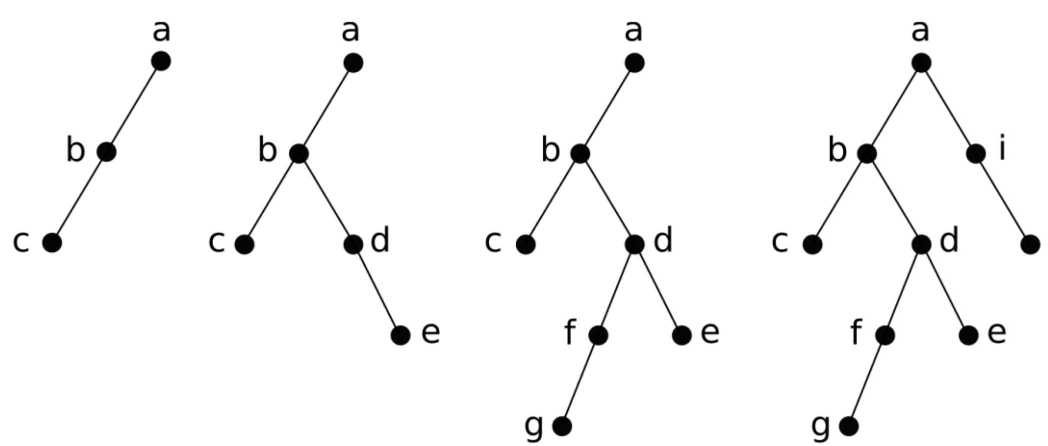
Depth-first Search

Recursive algorithm

DFS(G: connected graph with vertices $v_1, v_2, \ldots, v_n$)
$T :=$ tree consisting only of the vertex v_1
visit(v_1)

visit(v: vertex of G)
for each vertex w adjacent to v and not yet in T
    add vertex w and edge (v, w) to T
    visit(w)





Breadth-first Search

Algorithm

BFS(G: connected graph with vertices $v_1, v_2, \ldots, v_n$)
T:= tree consisting only of vertex v_1
L:= empty list
put v_1 in the list L of unprocessed vertices
while L is not empty
    remove the first vertex v from L
    for each neighbour w of v
        if w is not in L and not in T **then**
            add w to the end of the list L
            add w and the edge (v, w) to T