



数理逻辑

(Mathematical Logic)

上海交通大学软件学院

吴刚

2013年春



离散数学简介

- 离散数学(Discrete Mathematics)
 - 是数学的几个分支的总称，以研究离散量的结构和相互间的关系为主要目标，其研究对象一般是有限个或可数无穷个元素；因此它充分描述了计算机科学离散性的特点。



离散数学简介

- 研究对象
 - 离散个体及其结构
- 研究思想
 - 以集合和映射为工具、体现公理化和结构的思想
- 研究内容：包含不同的数学分支
 - 集合论：离散结构的表示、描述工具
 - 数理逻辑：符号化的推理和证明
 - 代数结构：离散结构的代数模型
 - 图论：离散结构的关系模型
 - 组合数学：离散结构的存在性、计数、枚举、优化、设计
 - 计算建模：有限状态机、图灵机
 - 离散概率：二项分布、泊松分布、超几何分布



离散数学与计算机科学的关系

- 数理逻辑：系统建模、人工智能、程序正确性证明及验证
- 集合论：关系数据库模型
- 图论：数据结构、数据库模型、网络模型等
- 代数结构：软件规范、形式语义、编译系统、编码理论、密码学、数据仓库



学习数理逻辑的目的

- 掌握数理逻辑的描述方法和推理方法
 - 为其它专业课程的学习准备必要的数学工具
- 学习现代数学的思想方法
 - 结构化、公理化
- 培养分析问题解决问题的能力



学习数理逻辑的方法

了解问题背景



弄清基本概念



掌握基本方法

把握四个环节：

课前（浏览预习，只求了解）

课堂（认真听讲，当场消化）

课后（立即复习，融会贯通）

课余（多做习题，举一反三）

勤动脑、勤动手



数理逻辑

- What is Logic
- What is Mathematical Logic
- A Reasoning Example
 - 如果你好好弹琴，爸爸下午带你去科技馆
 - “如果我不好好弹琴，爸爸就不带我去了”
- Paradox（悖论）
 - 理发师只帮那些不为自己理发的人理发



课程内容

- 朴素集合论
- 命题逻辑
- 谓词逻辑
- 逻辑理论



课程安排

- 教材:

- 离散数学及其应用（英文精编版. 第6版）
 - 《离散数学及其应用》第6版
 - Discrete Mathematics and Its Applications
 - Kenneth H. Rosen
 - 机械工业出版社 (影印版)
- 《面向计算机科学的数理逻辑》第二版，陆钟万，科学出版社，2002



课程安排

- 成绩构成

- 平时成绩**40%**: 课堂、作业、课堂小测验
- 期末考试**60%**



联系方式

- 主讲：吴刚
 - 软件学院1208室，Tel: 34204046
 - wugang@cs.sjtu.edu.cn
- 助教：
 - 陈晨阳 757638733@qq.com
 - 王春 wangchun0109@126.com
 - TA hours: 周三下午4点-5点半 软院5320

- 
-
- Enjoy the course!



Logic & Proof (1)

Wu Gang
School of Software, SJTU



Logic

- Logic gives precise meaning to mathematical statements, and allows consistent math reasoning
 - Many applications in CS, construction and verification computer programs, circuit design, etc
- Contents
 - Propositions
 - Propositional Equivalence
 - Predicates and Quantifiers
 - Nested Quantifiers



Propositions

- Definition

- A declarative statement that is either true (T) or false (F), use p, q, r to denote proposition usually

- Examples

- $1+1=2$
- $1+1=3$
- Expo2014 will be held in Shanghai
- What is your name?
- Be quiet
- $1+X = 5$



Compound Propositions

- Definition

- New propositions formed by existing propositions and logical operators

- Logical Opetators

- \neg 、 \wedge 、 \vee 、 \oplus 、 \rightarrow 、 \leftrightarrow

not and or xor implies biconditional
(negation/conjunction/disjunction/exclusive
or/implication/biconditional implication)



True Table

- Let “ p, q ” be propositions

P	$\neg P$
T	F
F	T

P	Q	$P \wedge Q$	$P \vee Q$	$P \oplus Q$
T	T	T	T	F
T	F	F	T	T
F	T	F	T	T
F	F	F	F	F



OR and XOR

- Examples

1. Soup or salad comes with an entrée
2. We want a student who can play piano or violin
3. I will go to the moon or the mars tomorrow

- How to express them

- OR, inclusive or: 2
- XOR, exclusive or: 1, 3



Implications

- Implication: $P \rightarrow Q$, P IMPLIES Q.
 - P is hypothesis, Q is consequence.
 - Some ways: if P then Q, Q when P, Q follows from P, P only if Q, Q is the necessary for P
 - example: If you make no mistakes, then you'll get an A.
- Bidirectional implication: $P \leftrightarrow Q$,
 - P if and only if (iff) Q
 - $(P \rightarrow Q \text{ and } Q \rightarrow P)$
- Implications are often used in mathematical proofs



Implications

P	Q	$P \rightarrow Q$	$P \leftrightarrow Q$	
T	T	T	T	
T	F	F	F	
F	T	T	F	} <i>weird?</i>
F	F	T	T	

Consider: $P \rightarrow Q$.

(1) converse: $Q \rightarrow P$.

(2) contra-positive: $(\text{NOT } Q) \rightarrow (\text{NOT } P)$ (equiv.)

(3) inverse: $(\text{NOT } P) \rightarrow (\text{NOT } Q)$.



Precedence of logical operators

- Order of precedence: NOT, AND, OR, XOR, \rightarrow , \leftrightarrow
- example: $P \rightarrow Q \text{ AND NOT } R = P \rightarrow (Q \text{ AND } (\text{NOT } R))$.
- We usually use parentheses to make the proposition clear



Other logical operators

- \uparrow 、 \downarrow 、 \nrightarrow : not and、not or、not implies
- How many operators we could have
- How many operators are enough?

P	Q	Operator
T	T	U1
T	F	U2
F	T	U3
F	F	U4

- T、F、P、Q;
- \neg 、 \rightarrow 、 \nrightarrow : each one covers 2 results
- \wedge 、 \vee 、 \oplus 、 \leftrightarrow 、 \uparrow 、 \downarrow



Translating

- Sentence to logical expressions
- Examples
 - You can apply for an email account only if you are a CS major or you are not a freshman
 - P: You can apply for an email account
 - Q: you are a CS major
 - R: you are a freshman
 - $P \rightarrow (Q \vee \neg R)$



Using logic (1)

- Search engine
- System specification
 - Promotion rules
 - Send the email
- The specifications in one system should be consistent



Using Logic(2)

- Logic Puzzles (using logical reasoning)
 - Two kinds of inhabitants in a island. Knights always tell the truth, knaves always lie. You meet 2 people A & B. A says "B is a knight", B says "the two of us are opposite types". Who is knave?
 - "This statement is false"
logical paradox



Logic and bit operations

- Bits are units of information. 1=T, 0=F.
- Bit-strings are sequences of bits:
00011100101010
- Computer bit operations correspond to the logical connectives
- Bitwise OR, bitwise AND, bitwise XOR
 - 2 strings of the same length
 - Corresponding bits' OR/AND/XOR
 - Example: interrupt register and interrupt mask register



homework1

(1) Find 3 sets of logical operators

- The operator in each set can not be expressed by others in the same set
- The operators in each set can express all the other operators



homework1

(2) Page 13, No.5

Page 17, No.9 ver6

(3) Page 16, No.26

Page 20, No.51 ver6

(4) Page 16, No.28, 29

Page 20, No.55,59 ver6



Translation

- He is not a freshman.
- You will get a ticket if you live in Rm.405 or Rm. 406.
- if you are available, we can play basketball on this afternoon, unless it were running at that time.



Logic & Proof (2)

Wu Gang
School of Software, SJTU



Propositional Equivalence

- **Tautology:** *Proposition that is always true.* for example: $P \text{ OR } (\text{NOT } P)$.
- **Contradiction:** *Proposition that is always false.* for example: $P \text{ AND } (\text{NOT } P)$.
- Others: *Contingencies.*



Logical equivalences

- Two propositions are logically equivalent if $P \leftrightarrow Q$ is always true (tautology).
- This is denoted by $P \equiv Q$



Logical equivalences

- Example: De Morgan's Law

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

P	Q	$\neg(P \vee Q)$	$\neg P \wedge \neg Q$
T	T	F	F
T	F	F	F
F	T	F	F
F	F	T	T



Logical equivalences

- Proving equivalences by truth tables can easily become computationally demanding
 - equivalence with 2 prop.: truth table has columns of size 4.
 - equivalence with n prop.: truth table has columns of size 2^n .
- *Solution:* we use a list of known logical equivalences (building blocks) and manipulate the expression.



Logical equivalences

- Identity laws: $p \wedge T \equiv p, p \vee F \equiv p$
- Domination laws: $p \wedge F \equiv F, p \vee T \equiv T$
- Idempotent laws: $p \wedge p \equiv p, p \vee p \equiv p$
- Double negation law: $\neg(\neg p) \equiv p$
- Commutative laws: $p \vee q \equiv q \vee p$
- Associative laws: $(p \vee q) \vee r \equiv p \vee (q \vee r)$
- Distributive laws: $p \vee (q \wedge r) \equiv (p \vee q) \wedge (p \vee r)$
- De Morgan's laws: $\neg(p \vee q) \equiv \neg p \wedge \neg q$
- Absorption laws: $p \vee (p \wedge q) \equiv p$
- Negation laws: $p \vee \neg p \equiv T$



Logical equivalences

- The Dual of a compound prop. that contains only \neg , \wedge , \vee is the prop. obtained by replacing each \wedge by \vee , each \vee by \wedge , each T by F, each F by T, denoted by p^*
- If $p \equiv q$ then $p^* \equiv q^*$



Logical equivalences

- $p \rightarrow q \equiv \neg p \vee q$
- $p \leftrightarrow q \equiv (p \rightarrow q) \wedge (q \rightarrow p)$
- $p \vee q \equiv \neg p \rightarrow q$
- $p \wedge q \equiv \neg (p \rightarrow \neg q)$
- $(p \rightarrow q) \wedge (p \rightarrow r) \equiv p \rightarrow (q \wedge r)$
- $(p \rightarrow r) \vee (q \rightarrow r) \equiv (p \wedge q) \rightarrow r$



Normal Form

- Disjunctive Normal Form
 - $p \vee (\neg p \wedge q) \vee (p \wedge q \wedge \neg r)$
- Full disjunctive normal form
 - $(p \wedge q \wedge r) \vee (\neg p \wedge q \wedge r) \vee (\neg p \wedge q \wedge \neg r)$
- Conjunctive Normal Form



Predicates

- consider statements with *variables*: $x > 3$
 - x is the subject.
 - >3 is the *predicate* or property of the subject.
- We introduce a *propositional function* $P(x)$, that denotes >3 .
- If x is a specific number, the function becomes a proposition (T or F). For example:
 $P(2) = F$, $P(4) = T$.
- P is called a *predicate*



Predicates

- More generally, we can have “functions” of more than one variable.
- For each input value it assigns either T or F.
- example: $Q(x,y) = (x=y+3)$.
 $Q(1,2) = (1=2+3) = F$
 $Q(3,0) = (3=0+3) = T$



Quantifiers

- We do not always have to insert specific values. We can make propositions for general values in a *domain* (or *universe of discourse*):
- This is called: *quantification*.
- **Universal Quantification:** $P(x)$ is true for all values of x in the domain: $\forall x P(x)$
- **Existential Quantification:** There exists an element x in the domain such that $P(x)$ is true: $\exists x P(x)$



Quantifiers

- example: domain x is real numbers. $P(x)$ is $x > -1$.
 - $\forall x P(x)$ (F) *counter-example* : $x=-2$
 - $\exists x P(x)$ (T)
- example: domain is positive real numbers, $P(x)$ is $x > -1$.
 - $\forall x P(x)$ (T)
 - $\exists x P(x)$ (T)



Binding Variables

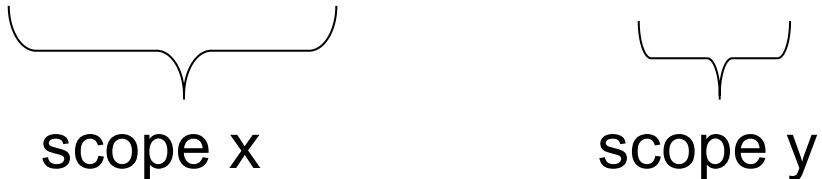
- A variable is bound if it has a value or a quantifier is “acting” on it.
- A statement can only become a proposition if all variables are bound.
- example: $\exists x P(x,y)$, x is bound, y is free.



Scope of a quantifier

- The *scope* of a quantifier is the part of the statement on which it is acting.
- example

$$\exists x (P(x) \wedge Q(x)) \vee \forall y R(y)$$





Negation of a quantified Prop.

- We can also negate propositions with quantifiers.
- Two important equivalences:
 - $\neg \forall x P(x) \equiv \exists x \neg P(x)$, It is not the case that for all x $P(x)$ is true \equiv there must be an x for which $P(x)$ is not true
 - $\neg \exists x P(x) \equiv \forall x \neg P(x)$, It is not true that there exists an x for which $P(x)$ is true \equiv $P(x)$ must be false for all x



Using Predictions and Quantifiers

- Translation sentence into logical expression
- Example: every students in this class has studied c language
 - $\forall x C(x)$ the universe of discourse for x is the students in this class
 - $\forall x (S(x) \rightarrow C(x))$ the universe of discourse for x is all the people
 - $\forall x (S(x) \rightarrow L(x, c))$



Using Predictions and Quantifiers

- Logic programming
 - Prolog language
 - Facts and rules
 - Example:
 - instructor (john, math202)
 - enrolled (tom, math202)
 - enrolled (jack, math202)
 - teacher (P,S) :- instructor (P,C), enrolled (S,C)
 - ?enrolled (tom, math202) *yes*
 - ?enrolled (X, math202)
 - ?teacher (X, tom)



homework2

(1) Use 2 ways to prove that
$$(p \vee q) \wedge (\neg p \vee r) \rightarrow (q \vee r)$$
is a tautology



homework2

(2) Page 37, No.6,8

Page 47, No.11,16 ver6

- No.11: let $P(x)$ be the statement " $x=x^2$ ". if the domain consists of the integers, what are the truth values? $P(0)$; $P(1)$; $P(2)$; $P(-1)$; $\exists x P(x)$; $\forall x P(x)$
- No.16: determine the truth value of each statements if the domain consists of real number? $\exists x(x^2=2)$; $\exists x(x^2=-1)$; $\forall x(x^2+2 \geq 1)$; $\forall x(x^2 \neq x)$



homework2

(3) Page 39, No.21, 22

Page 49, No.41,43 ver6

- No. 41: Express each of these system specifications
 - a) At least one mail message, among the nonempty set of messages, can be saved if there is a disk with more than 10 kilobytes of free space
 - b) whenever there is an active alert, all queued messages are transmitted.
 - c) The diagnostic monitor tracks the status of all systems except the main console
 - d) each participant on the conference call whom the host the call did not put on a special list was billed



homework2

(3)

- No. 43: Determine whether $\forall x(P(x) \rightarrow Q(x))$ and $\forall xP(x) \rightarrow \forall xQ(x)$ are logically equivalent. Justify your answer.



Quiz

- How to prove the equivalence of two logical statements
- Determine whether $\exists x(P(x) \rightarrow Q(x))$ and $\exists xP(x) \rightarrow \exists xQ(x)$ are logically equivalent. Justify your answer.
- $\neg (\forall x \exists y P(x) \rightarrow \neg Q(y))$
let the Negation act on the predicts directly



Nested Quantifiers

- **Nested Quantifier:** *Quantifier that appears within the scope of another quantifier*
- Examples: x, y, z are real numbers
 - $\forall x \exists y (x+y=0)$
 - $\forall x \forall y \forall z (x+(y+z) = (x+y)+z)$
 - $\forall x \forall y (x>0) \wedge (y<0) \rightarrow (xy<0)$
 - For all x and for all y if x is positive and y is negative then their product must be negative.
 - The product of a positive and a negative real number is negative.



Using nested quantifiers

- Translate this sentence into a logical expressions.
 - "If a person is female and is a parent, then she is someone's mother."
- $F(x)$ is "x is female", $P(x)$ is "x is someone's parent", $M(x,y)$, "x is the mother of y"
 - $\forall x \exists y (F(x) \wedge P(x)) \rightarrow M(x,y)$



Negating nested quantifiers

- $\forall x \exists y (x+y=0)$
- $\neg \forall x \exists y (x+y=0)$
 $\equiv \exists x \neg \exists y (x+y=0)$
 $\equiv \exists x \forall y \neg (x+y=0)$
 $\equiv \exists x \forall y (x+y \neq 0)$



Order of quantifiers

- The order in which quantifiers occur can be very *important!*
 - $\forall x \exists y p(x,y) \leftrightarrow \exists y \forall x p(x,y)$ *not always true*
- Example $\exists y \forall x (x + y = 0) \leftrightarrow \forall x \exists y (x + y = 0)$
 - Left side: F, Right side: T
 - The true value of the proposition is false



Order of quantifiers

- The following compound propositions are always true, Left and right are equivalent.
 - $\forall x \forall y p(x,y) \leftrightarrow \forall y \forall x p(x,y)$
 - $\exists x \exists y p(x,y) \leftrightarrow \exists y \exists x p(x,y)$



Thinking quantification as loop

- **Tip:** You can think of expression with quantifiers as executing “loops” in a computer program
 - $\exists y \forall x p(x,y)$
 - First loop over y and for every y loop over x ; For every value of y , check if $p(x,y)$ is true for all x . If you found one “ y ”, the proposition must be true.

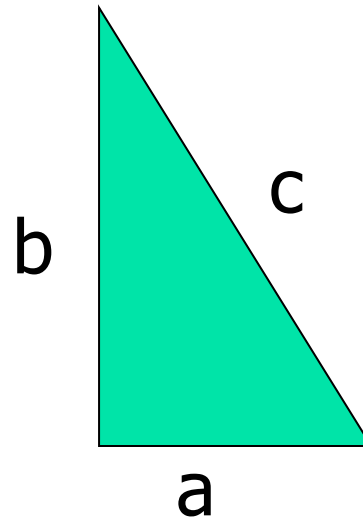


Logic & Proof (3)

Wu Gang
School of Software, SJTU



Proof (Pythagorean theorem)

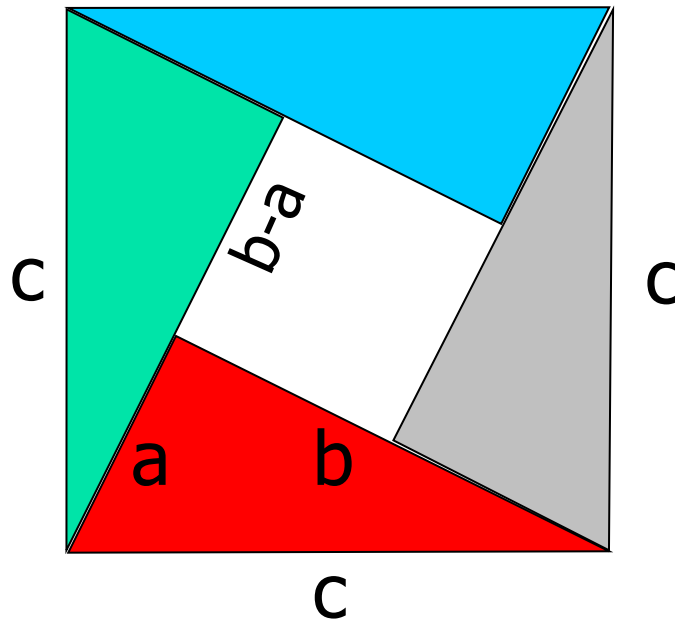


$$a^2 + b^2 = c^2$$



Proof (Pythagorean theorem)

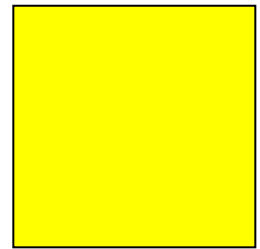
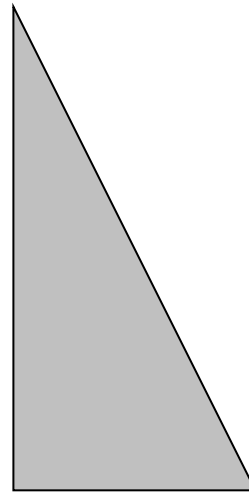
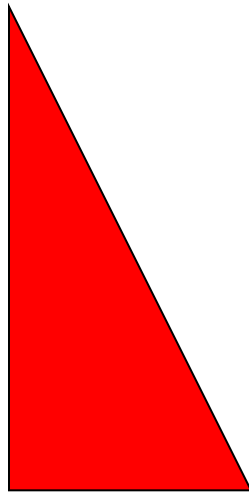
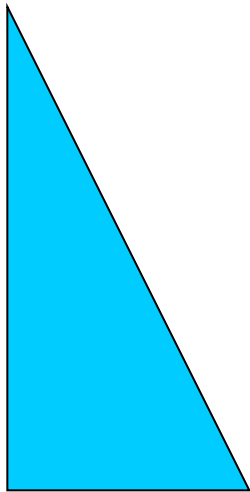
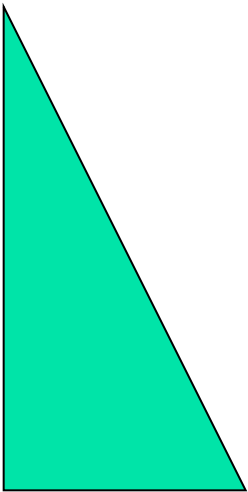
- A cool proof





Proof (Pythagorean theorem)

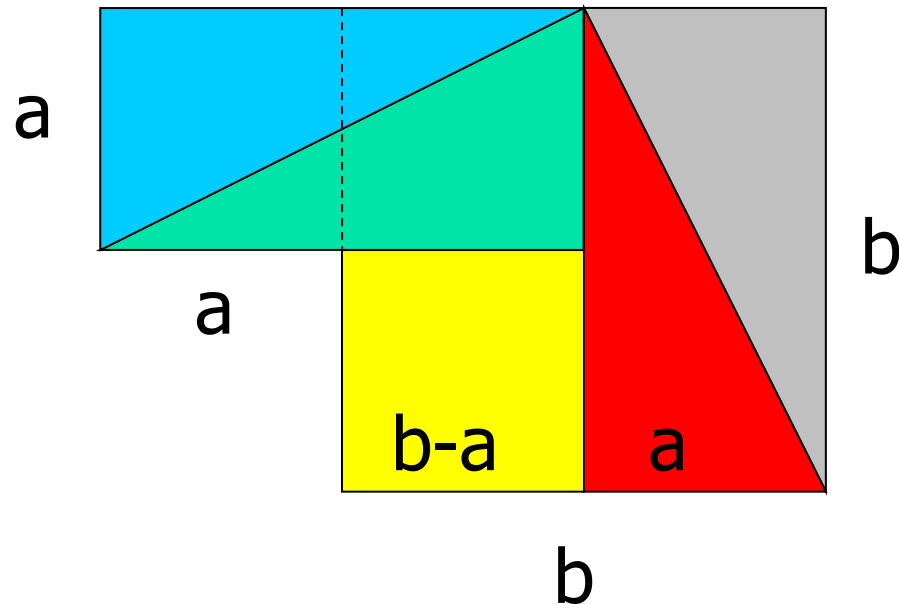
- A cool proof





Proof (Pythagorean theorem)

- A cool proof





Proof

- What's wrong with this?

a, b positive integers

$$a = b \Leftrightarrow$$

$$a^2 = ab \Leftrightarrow$$

$$a^2 - b^2 = ab - b^2 \Leftrightarrow$$

$$(a - b)(a + b) = b(a - b) \Leftrightarrow$$

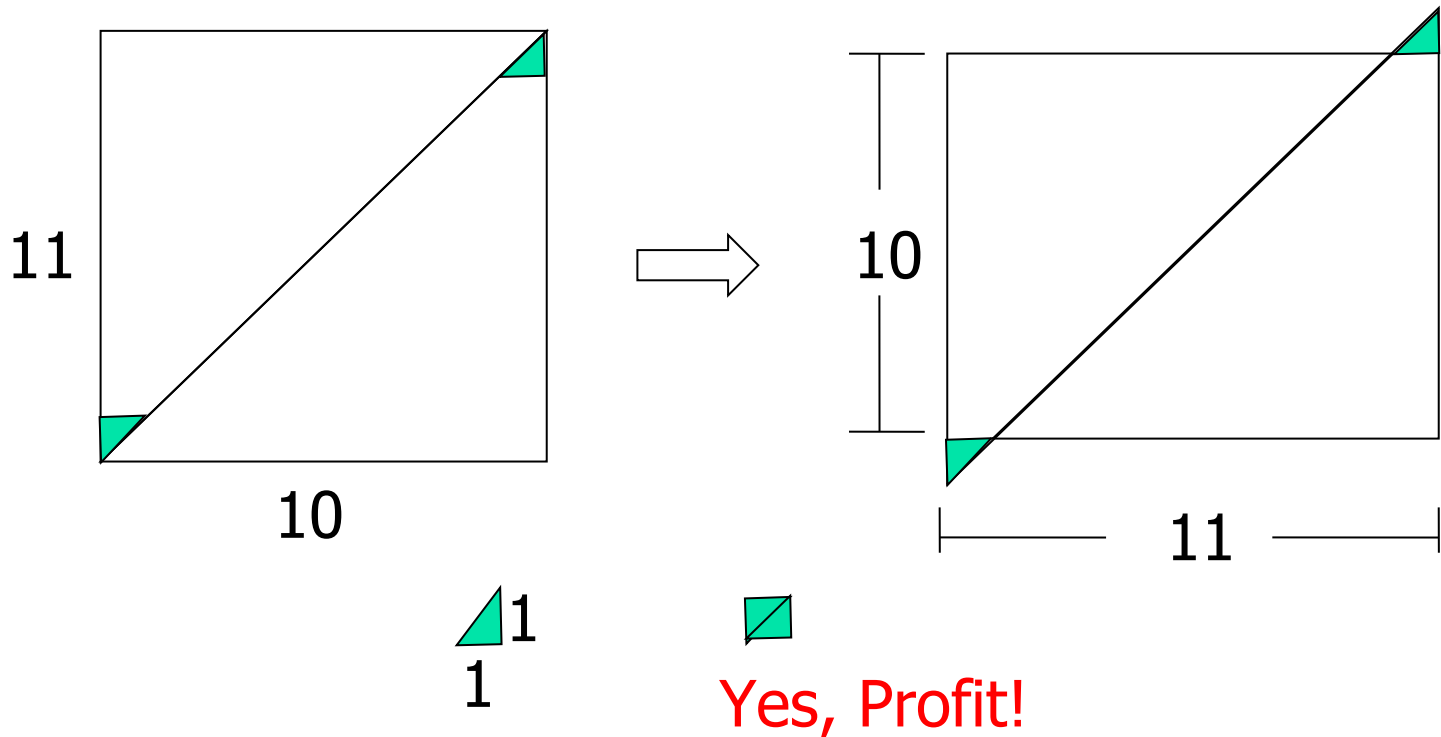
$$a + b = b \Leftrightarrow$$

$$2b = b \Leftrightarrow$$

$$2 = 1$$

Proof

- What's wrong with this?





Proof

- Definitions

- Theorem: statement that can be proved
- Proof: a sequence of statements to demonstrate that a theorem is true
- Axioms/postulates: the basic assumptions on which the proof is based
- Rules of inference: used to draw conclusions from other assertions



Proof

- Definitions

- Lemma: a simple theorem used to proof other theorem
- Corollary: Result that is directly follows from a theorem you just proved
- conjecture: Statement with unknown truth value



Rule of inference (Proposition Logic)

- Base: modus ponens (law of detachment)
 - $(p \wedge (p \rightarrow q)) \rightarrow q$ is a tautology

$$\begin{array}{c} p \\ p \rightarrow q \\ \hline \therefore q \end{array}$$

That is: if the premises p and $p \rightarrow q$ are both true, then q can only be true.



Rule of inference

- Examples

- it snows today

- If it snows today we go skiing

- Therefore: we go skiing

p

$p \rightarrow q$

$\therefore q$



Rules of inference

$$p \rightarrow (p \vee q)$$

$$(p) \wedge (q) \rightarrow (p \wedge q)$$

$$p \wedge q \rightarrow p$$

$$[\neg q \wedge (p \rightarrow q)] \rightarrow \neg p$$

$$\frac{p}{\therefore p \vee q}$$

addition

$$\frac{p}{q} \\ \therefore p \wedge q$$

conjunction

$$\frac{p \wedge q}{\therefore p}$$

simplification

$$\frac{\neg q}{p \rightarrow q} \\ \therefore \neg p$$

modus tollens

$$[(p \rightarrow q) \wedge (q \rightarrow r)] \rightarrow (p \rightarrow r)$$

$$[(p \vee q) \wedge \neg p] \rightarrow q$$

$$[(p \vee q) \wedge (\neg p \vee r)] \rightarrow (q \vee r)$$

$$\frac{p \rightarrow q}{q \rightarrow r} \\ \therefore p \rightarrow r$$

hypothetical syllogism

$$\frac{p \vee q}{\neg p} \\ \therefore q$$

disjunctive syllogism

$$\frac{p \vee q}{\neg p \vee r} \\ \therefore q \vee r$$

resolution



Valid arguments

- An argument form is called **valid** If whenever all the hypotheses are true, the conclusion is also true
 - Showing q follows from the hypotheses p_1, p_2, \dots, p_n is the same as showing that the implication $(p_1 \wedge p_2 \wedge \dots \wedge p_n) \rightarrow q$ is true

For an argument to be true all the premises must be true;

Several rules of inference need to be used



Valid arguments (example)

- Hypotheses

- If you love me, then you will go with me
- If you don't love me, then you will go with him
- If you go with him, then I will cry

- Conclusion

- You didn't go with me, then I will cry



Valid arguments (example)

- p : you love me
- q : you will go with me
- r : you will go with him
- s : I will cry
- Hypotheses: $p \rightarrow q$, $\neg p \rightarrow r$, $r \rightarrow s$
- Conclusion: $\neg q \rightarrow s$



Valid arguments (example)

- Hypotheses: $p \rightarrow q$, $\neg p \rightarrow r$, $r \rightarrow s$
- Conclusion: $\neg q \rightarrow s$

1	$p \rightarrow q$	hypothesis
2	$\neg q \rightarrow \neg p$	contrapositive of step1
3	$\neg p \rightarrow r$	hypothesis
4	$\neg q \rightarrow r$	hypothetical syllogism(2,3)
5	$r \rightarrow s$	hypothesis
6	$\neg q \rightarrow s$	hypothetical syllogism(4,5)



Resolution (rule of inference)

- Tautology $((p \vee q) \wedge (\neg p \vee r)) \rightarrow (q \vee r)$
- Important in Programming languages based on rules of logic, like Prolog
- Clause: disjunction of variables or their negations
- Example: prove $(p \wedge q) \vee r, r \rightarrow s$ imply $p \vee s$
 - $(p \vee r) \wedge (q \vee r) \wedge (\neg r \vee s)$



Fallacies

- Not based on tautology
- Normal fallacies
 - *fallacy of denying hypothesis*
 - *fallacy of affirming conclusion*

$$p \rightarrow q$$

$$\underline{p}$$

$$\therefore q$$

correct

$$p \rightarrow q$$

$$\underline{\neg p}$$

$$\therefore \neg q$$

wrong

$$p \rightarrow q$$

$$\underline{q}$$

$$\therefore p$$

wrong



Inference for Quantified Statements

$$\frac{\forall x P(x)}{}$$

$\therefore P(c)$ for arbitrary c

Universal instantiation

$$\frac{P(c) \text{ for arbitrary } c}{}$$

$\therefore \forall x P(x)$

Universal generalization

$$\frac{\exists x P(x)}{}$$

$\therefore P(c)$ for some element c

Existential instantiation

$$\frac{P(c) \text{ for some element } c}{}$$

$\therefore \exists x P(x)$

Existential generalization



Example1

- Prove

- Everyone in this math class has taken a CS course;
Marla is in this math class; Therefore Marla has taken
a CS course

- Let

- $D(x)$ = x has taken this math class
- $C(x)$ = x has taken a CS class.
- premises: $\forall x (D(x) \rightarrow C(x))$; $D(\text{Marla})$
- conclusion: $C(\text{Marla})$

1. $\forall x (D(x) \rightarrow C(x))$ *Premise*
2. $D(\text{Marla}) \rightarrow C(\text{Marla})$ *Universal instantiation (1)*
3. $D(\text{Marla})$ *Premise*
4. $C(\text{Marla})$ *Modus ponens (2)(3)*



Example2

- Prove

- “A student in this class has not read the book” and “Everyone in this class passed the quiz” imply “someone who passed the quiz has not read the book”

- Let

- $C(x)$ = x is in this class;
- $B(x)$ = x has read the book;
- $P(x)$ = x passed the quiz;
- Premises: $\exists x(C(x) \wedge \neg B(x)); \forall x(C(x) \rightarrow P(x))$
- Conclusion: $\exists x(P(x) \wedge \neg B(x));$



Example2

- Premises: $\exists x(C(x) \wedge \neg B(x)); \forall x(C(x) \rightarrow P(x))$
- Conclusion: $\exists x(P(x) \wedge \neg B(x));$

1. $\exists x (C(x) \wedge \neg B(x))$

premise

2. $C(a) \wedge \neg B(a)$

existential instantiation (1)

3. $\forall x (C(x) \rightarrow P(x))$

premise

4. $C(a) \rightarrow P(a)$

universal instantiation (3)

5. $C(a)$

simplification from (2)

6. $P(a)$

modus ponens (4)(5)

7. $\neg B(a)$

simplification (2)

8. $P(a) \wedge \neg B(a)$

conjunction (6)(7)

9. $\exists x (P(x) \wedge \neg B(x))$

existential generalization(8)



Method of Proving theorems

- To prove the implication $p \rightarrow q$
 - Direct Proof: Assume p is true and use rules of inference to prove that q must be true.
 - Indirect Proof: $\neg q \rightarrow \neg p \equiv p \rightarrow q$, proof by contraposition
 - Vacuous Proof: to prove p is always false
 - Trivial Proof: to prove q is always true



Method of Proving theorems

- Proof by contradiction
 - If $\neg p \rightarrow F$ can be proved, then **p** is true
- Proof by cases
 - $(p_1 \vee p_2 \vee \dots \vee p_n) \rightarrow q$
 $\equiv (p_1 \rightarrow q) \wedge (p_2 \rightarrow q) \wedge \dots \wedge (p_n \rightarrow q)$
- Proof of equivalence
 - $p \leftrightarrow q \equiv (p \rightarrow q) \wedge (q \rightarrow p)$



Method of Proving theorems

- Existence proofs
 - Constructive
 - Non constructive
- Uniqueness proofs
 - Existence & Uniqueness
- Counter examples
 - To show $\forall x P(x)$ is false



Proof strategy

- Include
 - Forward and backward reasoning
 - Leveraging proof by cases
 - Adapting existing proofs
 - Conjecture\proof\counterexample
- Read by yourself!



Homework

(1) Page 58, No.3

Page 72, No.5 ver6

Use rules of inference to show that the hypotheses "Randy works hard," "if Randy works hard then he is a dull boy" "If Randy is a dull boy, then he will not get the job" imply the conclusion "Randy will not get the job"



Homework

(2) Page 58, No.7 ver5 OR *Page 73, No.13 ver6*

- For each argument, explain which rules of inference are used, give the steps
 - Doug, a student in this class, knows how to write programs in Java. Everyone who knows how to write programs in Java can get a high-paying job. Therefore someone in this class can get a high-paying job.
 - Someone in this class enjoys whale watching. Every person who enjoy whale watching cares about ocean pollution. Therefore there is a person in this class who cares about ocean pollution.
 - Each of the 93 students in this class owns a PC. Everyone who owns a PC can use a word processing program. Therefore, Zeke, a students in this class, can use a word processing program.
 - Everyone is NJ lives within 50 miles of the ocean. Someone in NJ has never seen the ocean. Therefore someone who lives within 50 miles of the ocean has never seen the ocean.



Homework

(3) Page 59, No.9

Page 74, No.17 ver6

What is wrong with the argument?

Let $H(x)$ be “ x is happy”, Given the premise
 $\exists x H(x)$, we conclude that $H(\text{Lola})$. Therefore Lola is
happy.



Homework

(4) Page 69, No.9

Page 85, No.17 ver6

Show that if n is an integer and n^3+5 is odd, then n is even using

- A proof by contraposition
- A proof by contradiction



Basic Structures (1)

Wu Gang
School of Software, SJTU



Basic Discrete Structures

- Sets
- Functions



Sets

- Definition

- An unordered collection of objects
- Georg Cantor 1895, Naïve set theory
- Russell's Paradoxes, logical inconsistency
- Axiom set theory
- Usually use uppercase letters to denote the set, as "A,B,C..."



Set and elements

- The objects in a set are called the elements, usually use lowercase letters to denote them, as “a,b,c...”
- $a \in A$: a is an element of the set A, or A contains a
- $a \notin A$: a is not an element of the set A



Methods to describe a set

- List all the elements between braces
 - $A = \{0, 1, 2, 3, 4, 5, 6, 7, 8, 9\}$
 - $B = \{a, b, c, \dots, x, y, z\}$
 - $C = \{1, 2, 3, 4, \dots\}$
 - $D = \{\{1, 2\}, \{3, 4, 5\}\}$
- Set builder notation
 - $A = \{x \mid x \text{ is the natural number less than } 10\}$
 - $B = \{x \mid x \text{ is the English alphabet}\}$
 - $C = \{x \mid P(x)\}$
- Venn diagrams
 - Usually used to indicate the relationship between sets



Special sets

- Universal set denoted as "U"
- Empty set denoted as " \emptyset "
- $N = \{0, 1, 2, 3, \dots\}$ natural numbers
- $Z = \{\dots, -2, -1, 0, 1, 2, \dots\}$ integers
- $Z^+ = \{1, 2, 3, \dots\}$ positive integers
- $Q = \{x \mid x \text{ is a rational number}\}$
- $R = \{x \mid x \text{ is a real number}\}$

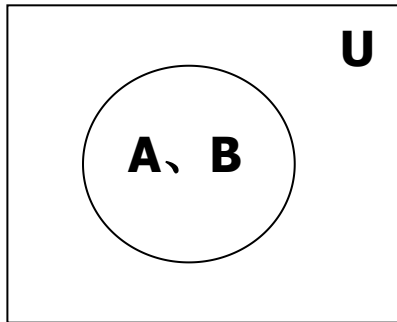


Relationship between sets

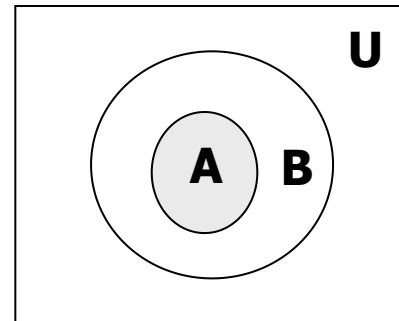
- Equality
 - Have the same elements
 - $\{1,2,3\}, \{2,1,3\}, \{1,1,3,2,2\}$
- Subset
 - A is a subset of B if and only if every element of A is also a element of B
 - $A \subseteq B$ if and only if $\forall x (x \in A \rightarrow x \in B)$
 - Theorem: For any set A, $\emptyset \subseteq A$ and $A \subseteq A$
 - Proper subset $A \subset B$: $A \subseteq B$ & $A \neq B$
- $A=B$ iff $A \subseteq B$ & $B \subseteq A$ " $\forall x (x \in A \leftrightarrow x \in B)$ "



Vann Diagram



$A=B$



$A \subset B$



Cardinality

- Exactly n distinct elements in set S where n is a nonnegative integer
 - We say S is a **finite set** and n is the **cardinality** of S , denoted as $|S|=n$
- A set is **infinite** if it is not finite
 - \aleph_0, \aleph



Power set

■ Definition

- Given a set S , the power set of S is a set of all subsets of S , denoted by $P(S)$
- $S = \{1, 2\}$ then $P(S) = \{\emptyset, \{1\}, \{2\}, \{1, 2\}\}$
- $S = \emptyset$ then $P(S) = \{\emptyset\}$
- $S = \{\emptyset\}$ then $P(S) = \{\emptyset, \{\emptyset\}\}$
- If S has n elements, then $P(S)$ has 2^n elements



Cartesian Products

- Ordered n-tuples
 - (a_1, a_2, \dots, a_n)
 - $(a_1, a_2, \dots, a_n) = (b_1, b_2, \dots, b_n)$ iff $a_i = b_i$ for $i = 1, 2, \dots, n$
 - **Ordered pair:** $(a, b), (c, d)$
- Cartesian product of sets A and B
 - Denoted by $A \times B$
 - $A \times B = \{(a, b) \mid a \in A \wedge b \in B\}$ **It is a Set of ordered pairs**
 - Rene Descartes

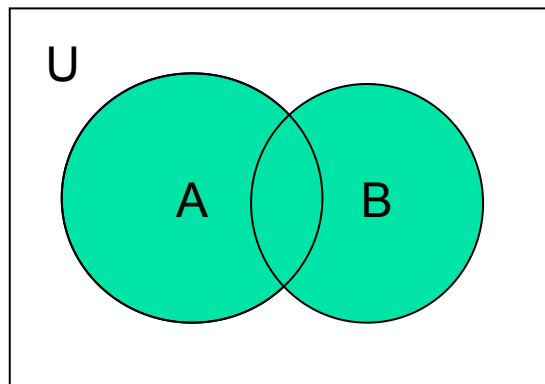


Cartesian Products

- Cartesian product of sets A_1, A_2, \dots, A_n
 - Denoted by $A_1 \times A_2 \times \dots \times A_n$
 - $A_1 \times A_2 \times \dots \times A_n = \{(a_1, a_2, \dots, a_n) \mid a_i \in A_i \text{ for } i=1, 2, \dots, n\}$
- Example
 - $A = \{1, 2\}, B = \{a, b, c\}$
 $A \times B = \{(1, a), (1, b), (1, c), (2, a), (2, b), (2, c)\}$
 $B \times A = \{(a, 1), (b, 1), (c, 1), (a, 2), (b, 2), (c, 2)\}$
 - $A = \emptyset$ then $A \times B = \emptyset$

Set operations

- Let A and B are sets
 - Union of A&B, denoted by $A \cup B$
$$A \cup B = \{x \mid x \in A \vee x \in B\}$$
 - Intersection of A&B, denoted by $A \cap B$
$$A \cap B = \{x \mid x \in A \wedge x \in B\}$$



Venn Diagram for $A \cup B$



Set Operations

- Let A and B are sets
 - Difference of A&B, denoted by A-B

$$A-B = \{x \mid x \in A \wedge x \notin B\}$$

- Let U be the universal set
 - Complement of set A, denoted by

$$\bar{A} = \{x \mid x \notin A\}$$



Set Operations

- Two sets are called disjoint if their intersection is the empty set
- Cardinality of the union of sets
(Principle of inclusion-exclusion)

$$|A \cup B| = |A| + |B| - |A \cap B|$$



Set Identities

- Identity laws: $A \cap U = A, A \cup \emptyset = A$
- Domination laws: $A \cap \emptyset = \emptyset, A \cup U = U$
- Idempotent laws: $A \cap A = A, A \cup A = A$
- Complementation law: $\overline{\overline{A}} = A$
- Commutative laws: $A \cup B = B \cup A$
- Associative laws: $(A \cup B) \cup C = A \cup (B \cup C)$
- Distributive laws: $A \cup (B \cap C) = (A \cup B) \cap (A \cup C)$
- De Morgan's laws: $\overline{A \cup B} = \overline{A} \cap \overline{B}$
- Absorption laws: $A \cup (A \cap B) = A$
- Complement laws: $A \cup \overline{A} = U$



Proof of the set identities

$$\overline{A \cap B} = \{x \mid x \notin A \cap B\}$$

$$\neg(x \in (A \cap B))$$

$$\neg(x \in A \wedge x \in B)$$

$$x \notin A \vee x \notin B$$

← De Morgan's laws
for logical equivalence

$$x \in \overline{A} \vee x \in \overline{B}$$

$$x \in \overline{A} \cup \overline{B}$$



Proof of the set identities

- Let A and B be sets, $A=B$ iff $A \subseteq B$ and $B \subseteq A$
- Membership tables

A	B	$A \cup B$
1	1	1
1	0	1
0	1	1
0	0	0

Generalized Unions and intersection

- Generalized Union: The union of a collection of sets

$$\bigcup_{i=1}^n A_i = A_1 \cup A_2 \cup \dots \cup A_n$$

- Generalized intersection: The intersection of a collection of sets

$$\bigcap_{i=1}^n A_i = A_1 \cap A_2 \cap \dots \cap A_n$$

- Example

- $A_i = \{i, i+1, i+2, \dots\}$

$$\bigcup_{i=1}^n A_i = \{1, 2, 3, \dots\}$$

$$\bigcap_{i=1}^n A_i = \{n, n+1, n+2, \dots\}$$

Computer representation of sets



- Store the elements of the set in an unordered fashion
 - Computing Union/intersection/difference of two sets would be time-consuming
- Store elements using an arbitrary ordering of the elements of the universal set U , for instance a_1, a_2, \dots, a_n
 - Represent a subset of U with a bit string of length n (bitmap)

Computer representation of sets

■ Examples

- Let $U=\{1,2,3,4,5,6,7,8,9\}$, the ordering of elements has the elements in increasing order
- $A=\{1,3,5,7,9\}$ 101010101
- $B=\{2,4,5,6\}$ 010111000
- $A \cup B$ bitwise or 111111101
- $A \cap B$ bitwise and 000010000



homework

(1) Page 97, No.3 (what about $\{2\}$?)

Page 120, No.5, 6 ver6

- For each of the following sets, determine whether 2 is the element of it.
 - a) $\{x \in \mathbb{R} \mid x \text{ is an integer greater than } 1\}$
 - b) $\{x \in \mathbb{R} \mid x \text{ is the square of an integer}\}$
 - c) $\{2, \{2\}\}$
 - d) $\{\{2\}, \{\{2\}\}\}$
 - e) $\{\{2\}, \{2, \{2\}\}\}$
 - f) $\{\{\{2\}\}\}$
- Determine whether $\{2\}$ is the element of above sets



homework

(2) Page 97, No.15, 16

Page 120, No.29, 31 ver6

- How many different elements does $A \times B$ have if A has m elements and B has n elements
- Explain why $A \times B \times C$ and $(A \times B) \times C$ are not the same



homework

(3) Page 106, No.10, 15

Page 131, No.19, 29 ver6

- Show that if A and B are sets, then

$$A - B = A \cap \overline{B}$$

- What can you say about the sets A and B if we know that
 - $A \cup B = A$? $A \cap B = A$? $A - B = A$?
 - $A \cap B = B \cap A$? $A - B = B - A$?



Basic Structures (2)

Wu Gang
School of Software, SJTU



Functions

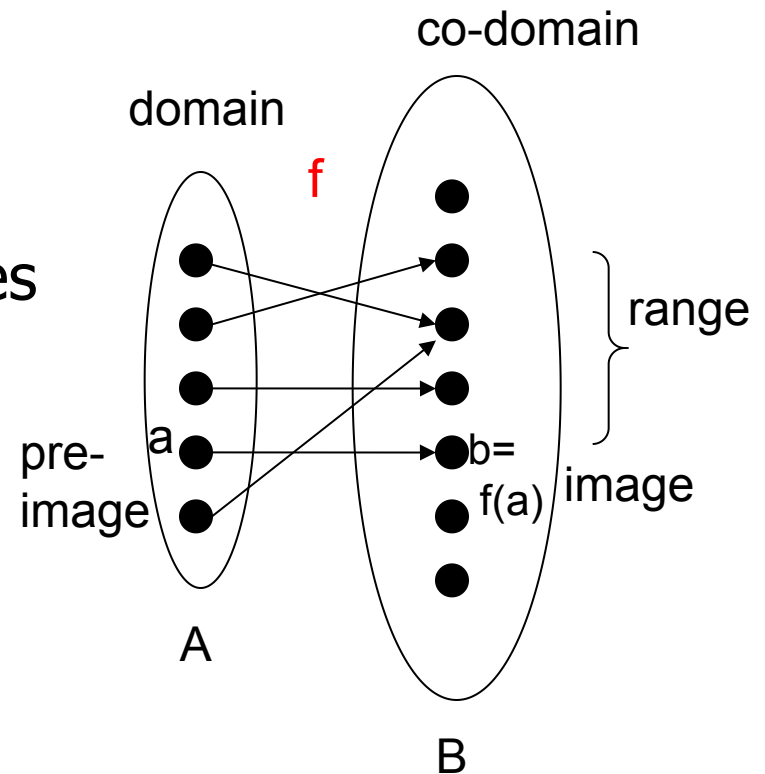
- Definition

- *The assignment of exactly one element of the set B to each element of the set A .
 $f:A \rightarrow B$ or $f(a)=b$.*
- *Function f maps A to B*

Functions

- A is the domain of f.
- B is the co-domain of f.
- b is the image of a.
- a is the pre-image of b.
- range of f: set of all images of elements of A.

Example: $f: \mathbb{Z} \rightarrow \mathbb{Z}$, $f(x) = x^2$
domain/co-domain: \mathbb{Z}
range: $\{0, 1, 4, 9, \dots\}$





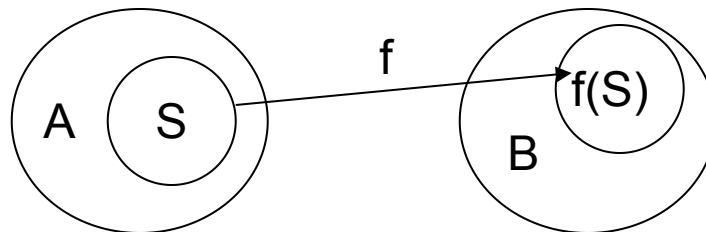
Functions

- If f_1 and f_2 are two functions from A to \mathbf{R} (real numbers), then $g=f_1+f_2$ and $h=f_1*f_2$ are also functions from A to \mathbf{R} defined by:
 - $(f_1+f_2)(x) = f_1(x) + f_2(x)$
 - $(f_1*f_2)(x) = f_1(x)*f_2(x)$
- Example: $f_1(x) = x$, $f_2(x) = x^2$.
 - $(f_1+f_2)(x) = x+x^2$
 - $(f_1*f_2)(x) = x^3$.



Functions

- $f:A \rightarrow B$, and S is a subset of A . Then we can define $f': S \rightarrow \text{Image}(S)=f(S)$



$$f(S) = \{f(s) \mid s \in S\}$$



One-to-one function

- **One-to-one or injective function:**

- *A function f is one-to-one if and only if $f(x)=f(y)$ implies $x=y$ for all x, y in the domain of f .*

$$\forall x \forall y (f(x) = f(y) \rightarrow x = y)$$

$$\forall x \forall y (x \neq y \rightarrow f(x) \neq f(y))$$

equivalent since: $(p \rightarrow q) \equiv (\neg q \rightarrow \neg p)$

Example: $f: \mathbb{Z} \rightarrow \mathbb{Z}$, $f(x) = x^2$ one-to-one?

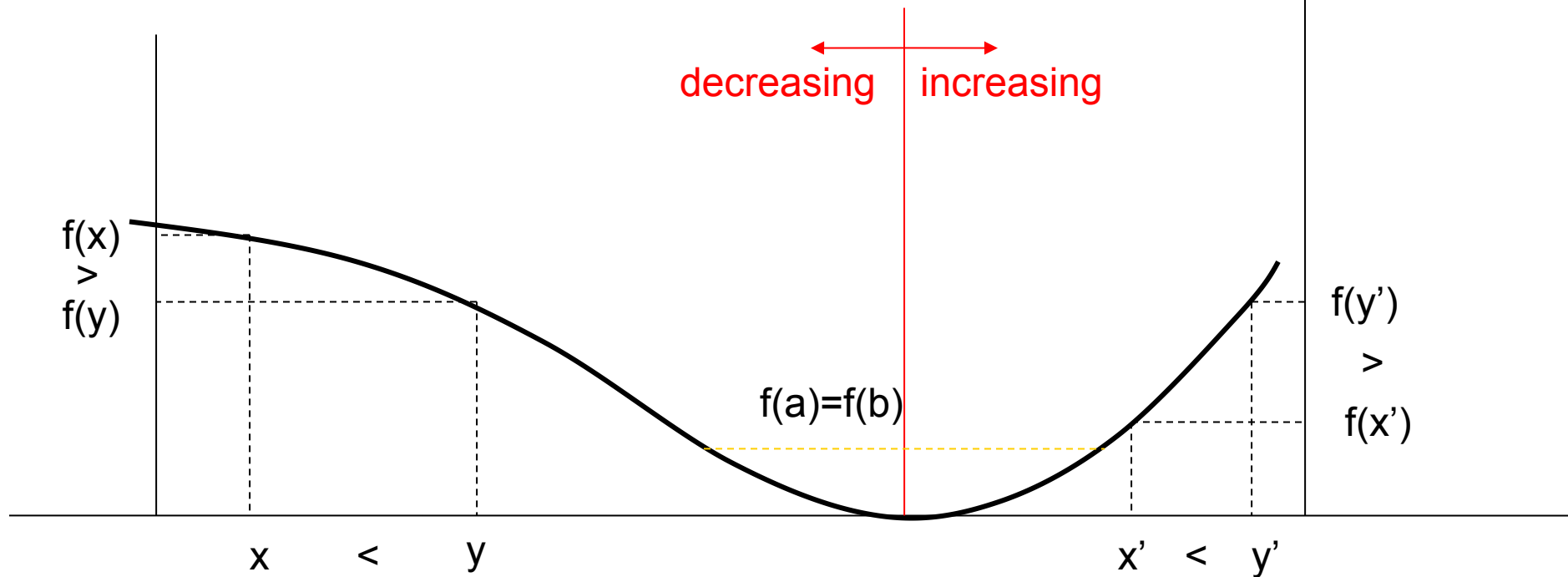
No! $x = -1$ & $x = 1$ map both to $f(1) = f(-1) = 1$.

One-to-one function

$\forall x \forall y (x < y \rightarrow f(x) < f(y))$ strictly increasing

$\forall x \forall y (x < y \rightarrow f(x) > f(y))$ strictly decreasing

x, y real number



Strictly increasing and strictly decreasing functions are one-to-one

Onto function

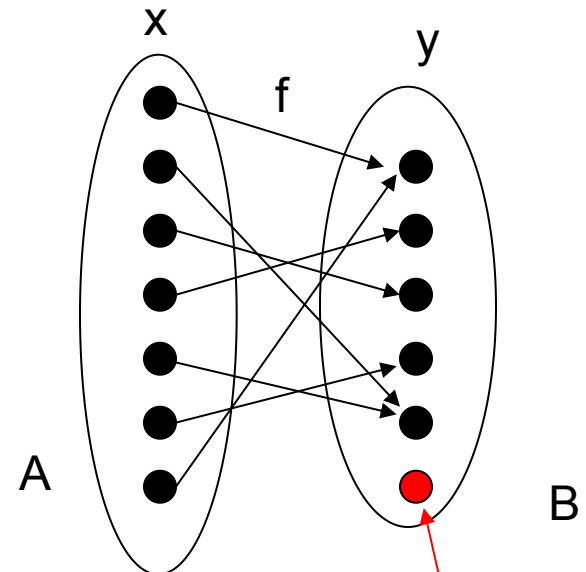
■ Onto or surjective functions:

- *A function f from A to B is onto if for every element b in B there is an element a in A with $f(a)=b$.*

$$\forall y \exists x (f(x) = y)$$

Example: $F: \mathbb{Z} \rightarrow \mathbb{Z}, f(x) = x^2$
onto?

No, $y = -1$ has no pre-image.



There is no element without incoming arrows



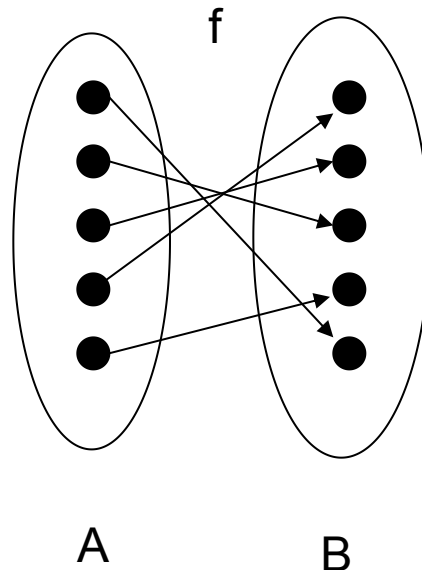
One-to-one correspondence

- **One-to-one correspondence or bijection:**

- *A function f is in one-to-one correspondence if it is both one-to-one and onto.*
- Example: $f:\mathbb{R}\rightarrow\mathbb{R}$, $f(x) = -x$ bijection!

One-to-one correspondence

- *Number of elements in A and B must be the same. Every element in A is uniquely associated with exactly one element in B .*





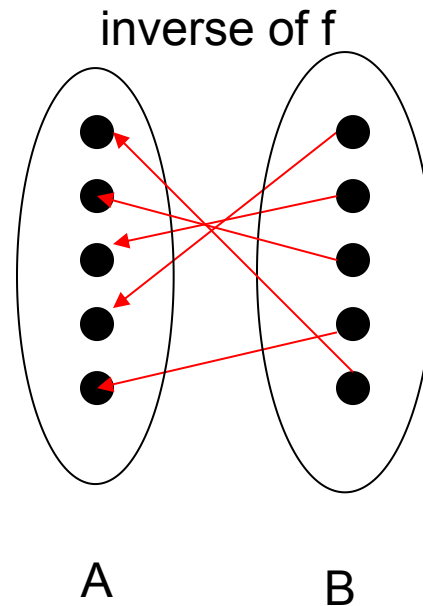
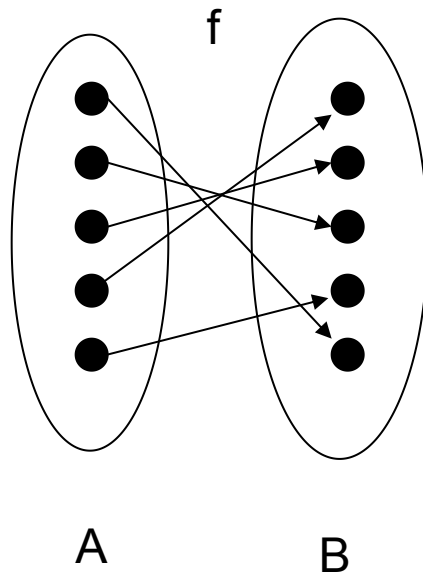
Inverse function

- **Inverse function:**

- *The inverse function of a bijection is the function that assigns to b in B the element a in A such that $f(a)=b$.*

Inverse function

$$f^{-1} : B \rightarrow A, f^{-1}(b) = a$$



If a function is not a bijection it is not invertible
example: $f: \mathbb{R} \rightarrow \mathbb{R}, f(x) = x^2$.



Composition of the functions

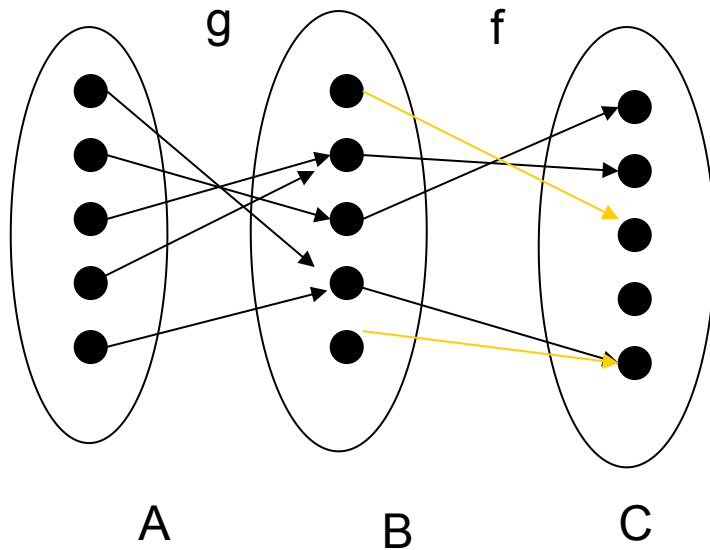
- **Composition:**

- *A composition of 2 functions $g:A \rightarrow B$ and $f:B \rightarrow C$ is defined by:*

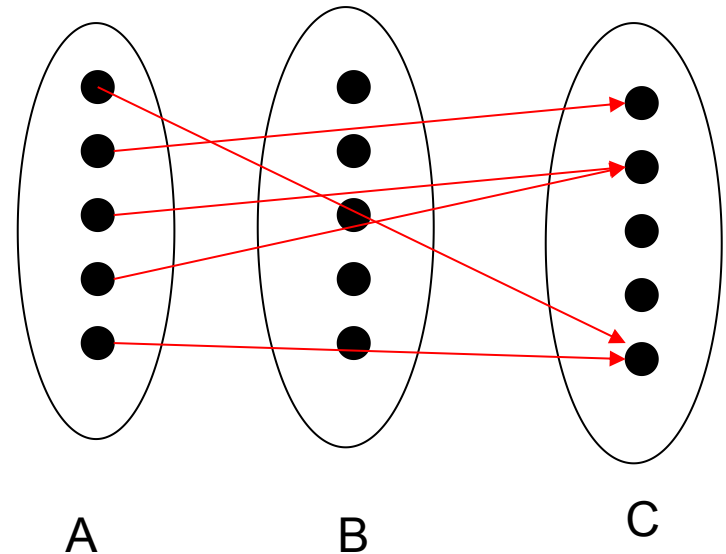
$$(f \circ g)(x) = f(g(x))$$

Composition of the functions

Range of g must be
subset of domain of f



$$(f \circ g)(x) = f(g(x))$$





Graph of a function

- **Graph of a function:**

- *The graph of a function f is the set of ordered pairs $\{(a,b) | a \text{ in } A, f(a)=b\}$.*
- This is a subset of the Cartesian product **$A \times B$** (i.e. it is a “relation”).



Some important functions

$\lfloor x \rfloor$ = largest integer smaller or equal to x :Floor

$\lceil x \rceil$ = smallest integer larger or equal to x : Ceiling.

$n! = 1 \cdot 2 \cdot 3 \cdot \dots \cdot n$: Factorial.

$b^n = b \cdot b \cdot b \cdot \dots \cdot b$ (n times): Exponential.

$\log_b(x)$ = inverse of exponential ($x > 0$).



Q&A 1

- Let $A_1 = \{a, b\}$, $B_1 = \{1, 2, 3\}$,
 $A_2 = \{a, b, c\}$, $B_2 = \{1, 2\}$,
 $A_3 = \{a, b, c\}$, $B_3 = \{1, 2, 3\}$,

How many one-to-one/onto/bijection functions can be defined from $A_1 \rightarrow B_1$, $A_2 \rightarrow B_2$, $A_3 \rightarrow B_3$



Q&A 2

- Let $A_1 = \{a, b\}$, $B_1 = \{1, \{2\}, 3\}$,
 $A_2 = \{\{a, b\}\}$, $B_2 = \{1, 2\}$,

Give the Cartesian Product

$A_1 \times B_1$, $A_2 \times B_2$, $A_2 \times A_1 \times B_2$



homework

- Page118 3, 6, 13

Page146 5, 15,29 ver6

- ◆ Find the domain and range of the functions. Note that in each case, to find the domain, determine the set of elements assigned values by the function
 - (a) The function that assigns to each bit string the number of ones minus the number of zeros
 - (b) The function that assigns to each bit string twice the number of zeros in that string
 - (c) The function that assigns to each bit string the number of bits left over when a bit string is split into bytes
 - (d) The function that assigns to each positive integer the largest perfect square not exceeding this integer



homework

- ♦ Determine whether the function $f: \mathbb{Z} \times \mathbb{Z} \rightarrow \mathbb{Z}$ is onto if
 - (a) $f(m,n)=m+n$
 - (b) $f(m,n)=m^2+n^2$
 - (c) $f(m,n)=m$
 - (d) $f(m,n)=|n|$
 - (e) $f(m,n)=m-n$



homework

- Suppose that g is a function from A to B and f is a function from B to C
 - (a) Show that if both f and g are one-to-one functions, then $f \circ g$ is also one-to-one
 - (a) Show that if both f and g are onto functions, then $f \circ g$ is also onto



Relations (1)

Wu Gang
School of Software, SJTU



Relations

- Binary relation
 - *Let A and B be sets, a binary relation from A to B is a subset of $A \times B$*
 - It is a set R of ordered pairs
 - Notation aRb to denote that $(a,b) \in R$, a is said to be related to b by R



Binary relation

- Examples

- A: set of the students in this class, B: set of the courses, R is the relation to say Student a enrolled in Course b.
(Obama, CS112), (Bush, CS911)
- Let $A=\{0,1,2\}$ $B=\{a,b\}$, then $\{(0,a), (0,b), (1,a), (2,b)\}$ is a relation from A to B

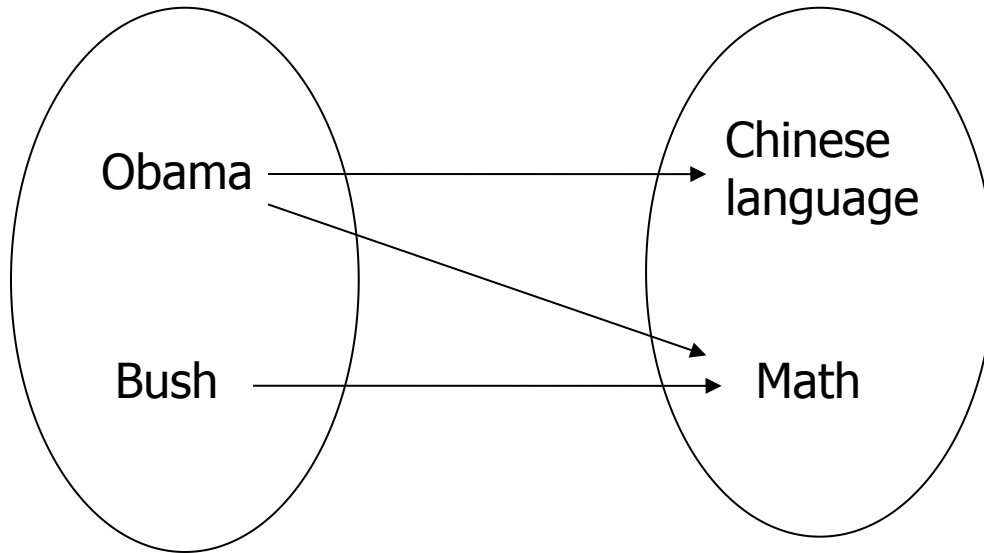


Functions & Relations

- Functions are special relations, Relations are a generalization of functions
 - The graph of a function is a set of ordered pairs and has the property that **every elements** of A is the first element of **exactly one** ordered pair



Functions & Relations



It is a relation but not a function



Relations on a set

- Relation on the set A is a relation from A to A
- Example
 - $A = \{1, 2, 3, 4\}$, Give the relation R on set A
 $R = \{(a, b) \mid a \text{ divides } b\}$
 $R = \{(1, 1), (1, 2), (1, 3), (1, 4), (2, 2), (2, 4), (3, 3), (4, 4)\}$
 - How many relations are there on a set with n elements??



Properties of relations on a set

relation R on the set A

- Reflexive

- $\forall a ((a,a) \in R), a \in A$

- Irreflexive

- $\forall a ((a,a) \notin R), a \in A$

- Symmetric

- $\forall a \forall b ((a,b) \in R \rightarrow (b,a) \in R) \quad a, b \in A$

- Antisymmetric

- $\forall a \forall b ((a,b) \in R \wedge (b,a) \in R \rightarrow (a=b)) \quad a, b \in A$



Properties of relations on a set

relation R on the set A

- Transitive

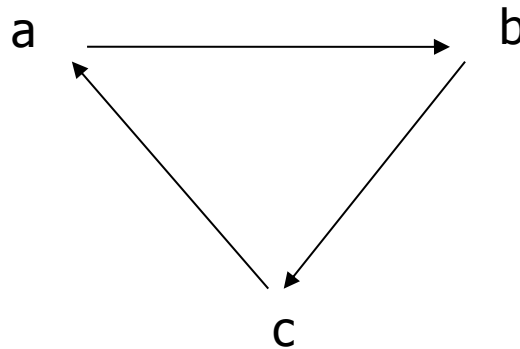
- $\forall a \forall b \forall c ((a,b) \in R \wedge (b,c) \in R \rightarrow (a,c) \in R)$
 $a,b,c \in A$



Properties of relations on a set

- Examples

- Relations “ = ” 、 “ > ” 、 “ ≥ ” on **N**
- Relation on \emptyset





Combining relations

- $R1$ and $R2$ are relations from A to B
 - $R1 \cup R2$
 - $R1 \cap R2$
 - $R1 - R2$
 - $R2 - R1$



Combining relations

- R is a relation from A to B, S is a relation from B to C, then
 - The composite of R and S is the relation from A to C denoted as $S \circ R$, for each $(a,c) \in S \circ R$, there exists an element $b \in B$ such as $(a,b) \in R$, $(b,c) \in S$



Combining relations

- R is a relation on set A

the powers R^n , $n=1,2,3\dots$, are defined recursively by

$$R^1 = R \quad \text{and} \quad R^{n+1} = R^n \circ R$$

- **Theorem:** the relation R on set A is transitive iff $R^n \subseteq R$ $n=1,2,3\dots$



N-ary relations

- Which trains start from shanghai to beijing after 9pm?
- who in your class get more than 2 A's in last semester?
- N-ary relations and computer database



N-ary relations

- Definition

- Let A_1, A_2, \dots, A_n be sets, an n -ary relation on these sets is a subset of $A_1 \times A_2 \times \dots \times A_n$, A_1, A_2, \dots, A_n are called domains of the relation, n is called its degree

- Examples

- (D, 305, shanghai, beijing, 9:00pm)
- (Zhangsan, G10101, english, A, 2009, Fall)



Databases and relations

- Relational data model!
- Database
 - Table
 - Record
 - Field
 - Primary key/composite key



Operations on N-ary relations

- Selection operator σ_C
 - Maps the n-ary relation R to the n-ary relation of all n-tuples from R that satisfy the condition C
- Projection operator $\pi_{i_1, i_2, \dots, i_m}$
 - Maps the n-tuple (a_1, a_2, \dots, a_n) to the m-tuple $(a_{i_1}, a_{i_2}, \dots, a_{i_m})$, where $m \leq n$



Operations on N-ary relations

- Join operator $J_p(R, S)$
 - Let R be a relation of degree m , S a relation of degree n , $p \leq m, p \leq n$. $J_p(R, S)$ is a relation of degree $m+n-p$ that consists of all $(m+n-p)$ -tuples $(a_1, a_2, \dots, a_{m-p}, c_1, c_2, \dots, c_p, b_1, b_2, \dots, b_{n-p})$, where $(a_1, a_2, \dots, a_{m-p}, c_1, c_2, \dots, c_p)$ belongs to R and $(c_1, c_2, \dots, c_p, b_1, b_2, \dots, b_{n-p})$ belongs to S



homework

- Page 253 11

Page 528 25 ver 6

let R be the relation $R = \{(a, b) \mid a \text{ divides } b\}$ on the set of positive integers. Find

- a) R^{-1} b) \bar{R}



homework

- Page253 14

Page528 31 ver6

let R be the relation on the set of people consisting of pair (a,b) , where a is a parent of b . let S be the relation on the set of people consisting of pair (a,b) , where a and b are siblings(brothers or sisters). What are $S \circ R$ and $R \circ S$



homework

- Page 253 23

Page 528 49 ver 6

show that the relation R on a set A is symmetric if and only if $R = R^{-1}$, where R^{-1} is the inverse relation



homework

- Page253 26

Page528 55 ver6

let R be a reflexive relation on a set A ,
show that R^n is reflexive for all positive
integers n .



Relations (2)

Wu Gang
School of Software, SJTU



Representing Relations

- Represent a relation between finite sets
 - List its ordered pairs
 - Use 0-1 matrices (used in computer programs)
 - Directed graphs (for people to understand the relations better)



Using matrices

- Suppose that R is a relation from $A = \{a_1, a_2, \dots, a_m\}$ to $B = \{b_1, b_2, \dots, b_n\}$.
 - the elements have been listed in an arbitrary order (if $A=B$, then the order is the same).

use matrix $M_R = [m_{ij}]$ to represent R

$$M_{ij} = \begin{cases} 1 & \text{if } (a_i, b_j) \in R \\ 0 & \text{if } (a_i, b_j) \notin R \end{cases}$$



Using matrices

- Example

- $A=\{1,2,3\}$ $B=\{1,2\}$, R is the relation from A to B where $a>b$ for every pair (a,b)

$$R=\{(2,1),(3,1),(3,2)\}$$

$$M_R = \begin{bmatrix} 0 & 0 \\ 1 & 0 \\ 1 & 1 \end{bmatrix}$$



Using matrices

- The matrix of a relation **on a set** is a square matrix. Can be used to determine relation's properties
 - Reflexive “elements on main diagonal = 1”
 - Irreflexive
 - Symmetric “ $M_R = (M_R)^t$ ”
 - Antisymmetric



Using matrices

$$M_R = \begin{bmatrix} 1 & 1 & 0 \\ 1 & 1 & 1 \\ 0 & 1 & 1 \end{bmatrix}$$

Reflexive?	YES
Symmetric ?	YES
Antisymmetric ?	NO



Using matrices

- Represent the union and intersection of two relations
 - Using the 0-1matrix's boolean operations join \vee and meet \wedge
 - $M_{R1 \cup R2} = M_{R1} \vee M_{R2}$
 - $M_{R1 \cap R2} = M_{R1} \wedge M_{R2}$



Using matrices

■ Example

$$M_{R1} = \begin{bmatrix} 1 & 0 & 1 \\ 1 & 0 & 0 \\ 0 & 1 & 0 \end{bmatrix}$$

$$M_{R2} = \begin{bmatrix} 1 & 0 & 1 \\ 0 & 1 & 1 \\ 1 & 0 & 0 \end{bmatrix}$$

$$M_{R1 \cup R2} = M_{R1} \vee M_{R2} =$$

$$\begin{bmatrix} 1 & 0 & 1 \\ 1 & 1 & 1 \\ 1 & 1 & 0 \end{bmatrix}$$

$$M_{R1 \cap R2} = M_{R1} \wedge M_{R2} =$$

$$\begin{bmatrix} 1 & 0 & 1 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix}$$



Using matrices

- Represent the composite of two relations
 - Using the 0-1matrix's boolean product operations \odot
 - R is a relation from A to B, S is a relation from B to C, A/B/C have **m/n/p** elements
 - $M_{S \circ R} = M_R \odot M_S$
 - the elements have been listed in an arbitrary order



Using matrices

- Example

$$M_R = \begin{bmatrix} 1 & 0 & 1 \\ 1 & 1 & 0 \\ 0 & 0 & 0 \end{bmatrix}$$

$$M_S = \begin{bmatrix} 0 & 1 & 0 \\ 0 & 0 & 1 \\ 1 & 0 & 1 \end{bmatrix}$$

$$M_{S \circ R} = M_R \odot M_S = \begin{bmatrix} 1 & 1 & 1 \\ 0 & 1 & 1 \\ 0 & 0 & 0 \end{bmatrix}$$



Using matrices

- Represent the powers R^n
 - use the 0-1matrix's boolean powers operations

$$M_{R^n} = M_R^{[n]}$$



Using directed graph

- A directed graph consists of a set V of vertices (nodes) and a set E of edges (arcs)
 - Edge (a,b): a is initial vertex, b is terminal vertex
 - Edge (a,a) is called a loop



Using directed graph

- Using the directed graph of a relation to determine relation's properties
 - Reflexive “loops at every node”
 - Irreflexive “no loops”
 - Symmetric
 - Antisymmetric



Closures of relations

- Let \underline{R} be a relation on a set \underline{A} . Then \underline{R} may or may not have some property \underline{P} , like reflexivity/symmetry/transitivity.
 - If there is a relation \underline{S} with property \underline{P} containing \underline{R} , such that \underline{S} is the subset of **every** relation with property \underline{P} containing \underline{R}
 - **\underline{S} is called the closure of \underline{R} with respect to \underline{P}**



Closures

- Reflexive closure
 - $R \cup \Delta$, where $\Delta = \{(a, a) | a \in A\}$ is called diagonal relation on A
- Symmetric closure
 - $R \cup R^{-1}$
- Transitive closure
 - More complicated



Paths in directed graph

- A path from a to b in a directed graph is a sequence of edges

$(a, x_1)(x_1, x_2)(x_2, x_3) \dots (x_n, b)$

- The path is denoted by $a, x_1, x_2, \dots, x_n, b$
- Its length is $n+1$
- If $a=b$, the path is called a circuit/cycle
- **Attention:** A path can pass through a node more than once, an edge can occur more than once in a path



Paths in directed graph

- Theorem

- Let R be a relation on a set A . there is a path of length n (positive integer) from a to b iff $(a,b) \in R^n$



homework

- Page265 1 (*Page543 1 ver6*)

Represent each of these relations on $\{1,2,3\}$ with a matrix (the elements are in increasing order)

a) $\{(1,1), (1,2), (1,3)\}$ b) $\{(1,2), (2,1), (2,2), (3,3)\}$

c) $\{(1,1), (1,2), (1,3), (2,2), (2,3), (3,3)\}$

d) $\{(1,3), (3,1)\}$

- *Draw the directed graphs representing relations from above Exercise 1*



homework

- Page265 8,9 (*Page543 16,17 ver6*)

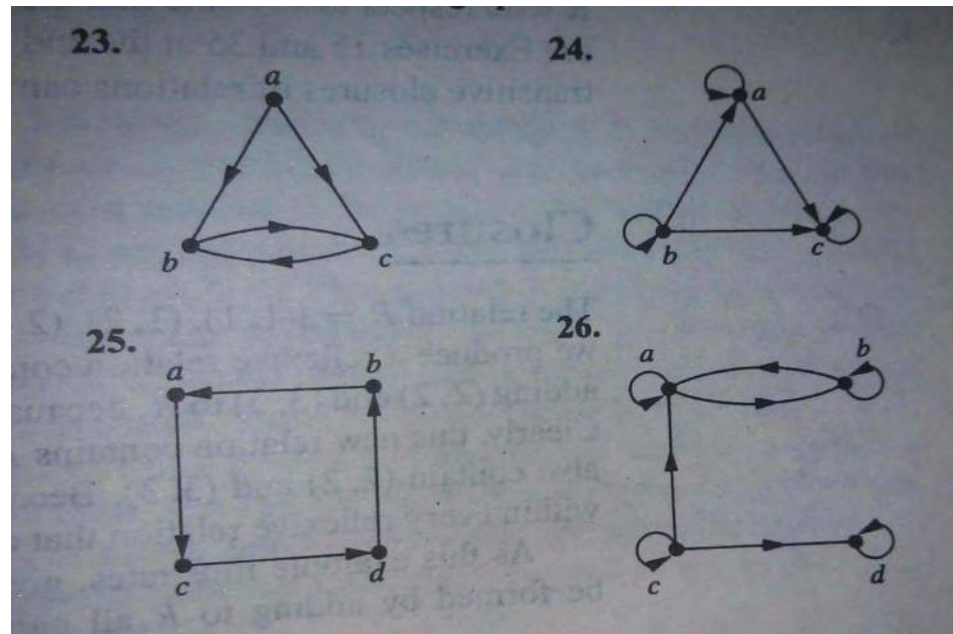
Let R be a relation on a set A with n elements. If there are k nonzero entries in M_R , the matrix representing R , how many nonzero entries are there in the matrix of the inverse of R ?

how many nonzero entries are there in the matrix of the complement of R ?

homework

- Page265 14 (*Page543 31 ver6*)

Determine whether the relations represented by the following graphs are reflexive, irreflexive, symmetric, antisymmetric, and/or transitive.





Relations (3)

Wu Gang
School of Software, SJTU



Transitive closures

- Connective relation R^*
 - Let R be a relation on a set A
 - R^* consists of the pairs (a,b) such that there is a path of length at least one from a to b in R

$$R^* = \bigcup_{n=1}^{\infty} R^n$$



examples

- Let R be the relation on the set of all people in the world that contains (a,b) if a has met b
 - What is R^n ? What is R^* ?
 - $(\text{you}, \text{Obama}) \in R^*$?



examples

- Let R be the relation on the set of all subway stops in shanghai that contains (a,b) if it is possible to travel from a to b without changing trains.
 - What is R^n ? What is R^* ?
 - $(\text{Dongchuan Road}, \text{Jianchuan Road}) \in R^5$?
 - $(\text{Dongchuan Road}, \text{Jianchuan Road}) \in R^4$?



Transitive closures

- Theorem

- The transitive closure of a relation R equals the connectivity relation R^*
 - R^* contains R
 - R^* is transitive
 - If there is a S , which is transitive and contains R , then S contains R^*



Transitive closures

- Lemma

- Let A be a set with n elements, R be a relation on A . If there is a path from a to b , then there is such a path with length not exceeding n . Furthermore, if $a \neq b$ then there is such a path with length not exceeding $n-1$

- $R^* = R \cup R^2 \cup R^3 \cup \dots \cup R^n$



Transitive closures

- $R^* = R \cup R^2 \cup R^3 \cup \dots \cup R^n$

$$M_{R^*} = M_R \vee M_R^{[2]} \vee M_R^{[3]} \dots \vee M_R^{[n]}$$



Equivalence relations

- Definition: A relation on a set A is an equivalence relation if it is reflexive, symmetric and transitive.
- Examples
 - Let R be the relation on the set of students in this class, aRb iff a and b come from the same province--- “Lao Xiang”
 - Congruence Modulo.
 - $R = \{(a, b) | a \equiv b \pmod{m}\}$ $m > 1$ is positive integer, a/b are integers



Equivalence classes

- Let R be an equivalence relation on set A . The set of elements that are related to an **element a** of A is called the equivalence class of **a** . denoted by $[a]_R$
 - $[a]_R = \{s \mid (a,s) \in R\}$
 - If there is only one relation under consideration, it can be denoted by $[a]$



Equivalence classes

- Examples

- [zhangsan] for relation “Lao xiang”
- $[0], [1], [5]$ for congruence modulo 4
 - $[0] = \{\dots -8, -4, 0, 4, 8 \dots\}$
 - $[1] = \{\dots -7, -3, 1, 5, 9 \dots\}$
 - $[5] = ?$
- Congruence classes modulo m , denoted by $[a]_m$, like $[5]_4$ and $[1]_3$



Equivalence classes and partitions

- Theorem1

- Let R be an equivalence relation on set A , these statements are equivalent

(1) aRb ; (2) $[a]=[b]$; (3) $[a] \cap [b] \neq \emptyset$



Equivalence classes and partitions

- Partition of a set S
 - a collection of disjoint nonempty subsets of S that have S as their union
 - $A_i \neq \emptyset \quad A_i \cap A_j = \emptyset \quad S = \bigcup_{i \in I} A_i$
- Theorem 2
 - The equivalence classes of an equivalence relation on a set form a partition of the set
 - Give a partition $\{A_i | i \in I\}$ of set A , there is an equivalence relation R that has the sets A_i as its equivalence classes



Partial orderings

- Definition

- A relation R on a set S is called partial order if it is reflexive, antisymmetric and transitive.
- A set S together with a partial order R is called a *partially ordered set* or *poset*, denoted by (S, R)



Partial orderings

- Examples
 - (\mathbb{Z}, \geq)
 - Inclusion relation on the power set of set S ,
 $(P(S), \subseteq)$
- (S, R)
 - It is not necessary that either aRb or bRa for every a and b in the set S



Partial orderings

- The elements a and b of a poset (S, \leq) are called **Comparable** if either $a \leq b$ or $b \leq a$, otherwise, incomparable
- **Total ordering**
 - If (S, \leq) is a poset, and every two elements of S are comparable, S is called totally ordered set or linearly ordered set
 - $(P(S), \subseteq)$ Total ordering?



Partial orderings

- Well-ordered set
 - a poset (S, \leq) is totally ordered, and such that every nonempty subset of S has a least element
 - Example: (\mathbb{Z}^+, \leq) (\mathbb{Z}, \leq)
- Lexicographic ordering
 - \leq on Cartesian product of two posets (A_1, \leq_1) and (A_2, \leq_2)
 - $(a_1, a_2) < (b_1, b_2)$ if $a_1 <_1 b_1$ or $(a_1 = b_1 \text{ and } a_2 <_2 b_2)$
 - $a < b$ denotes that $a \leq b$ and $a \neq b$



Partial orderings

- Lexicographic ordering
 - \leq on Cartesian product of N posets
 $(A_1, \leq_1) (A_2, \leq_2) \dots (A_n, \leq_n)$
 - $(a_1, a_2, \dots, a_n) < (b_1, b_2, \dots, b_n)$ if $a_1 <_1 b_1$ or there is an integer $i > 0$ such that $a_1 = b_1, a_2 = b_2, \dots, a_i = b_i$, and $a_{i+1} <_{i+1} b_{i+1}$
 - $(a_1, a_2, \dots, a_n) = (b_1, b_2, \dots, b_n)$ if $a_i = b_i, i = 1, 2, \dots, n$
- How to compare 8-bit binary number?



Partial orderings

- Lexicographic ordering of strings
 - $a_1a_2\dots a_m, b_1b_2\dots b_n$
 - let t be the minimum of m/n , then
$$(a_1, a_2, \dots, a_m) < (b_1, b_2, \dots, b_n) \text{ if}$$
$$(a_1, a_2, \dots, a_t) <_t (b_1, b_2, \dots, b_t) \text{ or}$$
$$(a_1, a_2, \dots, a_t) = (b_1, b_2, \dots, b_t) \text{ and } m < n$$
- How to compare “good”, “goodbye”, “gate”, “go”



Hasse Diagrams

- Give a directed graph of a finite poset,
 - Since every vertex has a loop, we can delete them (reflexive)
 - Since many edges must be present because it is transitive, we can remove them.
 - Arrange each edge so that its initial vertex is below its terminal vertex, we can remove the arrows
- Finally we get a hasse diagram



Hasse Diagrams

- Examples

- (\mathbb{Z}, \leq)
- $(\{2, 3, 6, 12, 24, 36\}, |)$
- (power set of $\{a, b\}$, \subseteq)



Maximal and minimal elements

- Maximal element
 - Poset (S, \leq) , $a \in S$, there is no $b \in S$ such that $a < b$
- Minimal element
- Greatest element
 - Poset (S, \leq) , $a \in S$, $b \leq a$ for every $b \in S$
- Least element



Maximal and minimal elements

- Upper bound
 - Poset (S, \leq) , $A \subseteq S$ $a \in S$, $b \leq a$ for every $b \in A$
- Lower bound
- Least upper bound
 - x is a upper bound of A , if there are other upper bound y of A , $x \leq y$
- Greatest lower bound



Maximal and minimal elements

- Examples

- ($S = \text{power set of } \{a, b, c\}, \subseteq$)

- (1) $A = \{\{a, b\}, \{b, c\}, \{b\}, \{c\}, \emptyset\}$

- (2) $A = \{\{a\}, \{c\}\}$

Give the Maximal/Minimal/Greatest/least element, and the Upper/lower/least upper/greatest lower bound

- Have upper bound, but no least upper bound???



homework

- Page 282 5 (*Page 563 9 ver 6*)
 - *Suppose that A is a nonempty set, and f is a function that has A as its domain. Let R be the relation on A consisting of all ordered pairs (x, y) such that $f(x)=f(y)$.*
 - a) show that R is an equivalence relation on A*
 - b) what are the equivalence classes of R ?*



homework

- Page 282 8 (*Page 563 15 ver 6*)
 - *Let R be the relation on the set of ordered pairs of positive integers such that $((a,b),(c,d)) \in R$ if and only if $a+d = b+c$. Show that R is an equivalence relation.*



homework

- Page 294 2 (*Page578 3 ver6*)
 - *Is (S,R) a poset if S is the set of all people in the world and $(a,b) \in R$, where a and b are people, if*
 - a) a is taller than b ?*
 - b) a is not taller than b ?*
 - c) $a=b$ or a is an ancestor of b ?*
 - d) a and b have a common friend?*



homework

- Page 294 17 (*Page 578 33 ver6*)
 - *Poset $(\{3, 5, 9, 15, 24, 45\}, |)$ 整除关系*
 - a) Find the maximal elements; b) Find the minimal elements; c) Is there a greatest element? d) Is there a least element? e) Find all upper bounds of $\{3, 5\}$; f) Find the least upper bound of $\{3, 5\}$, if it exists; g) Find all lower bounds of $\{15, 45\}$; h) find the greatest lower bounds of $\{15, 45\}$, if it exists*



Lattices & Boolean Algebra

Wu Gang
School of Software, SJTU



Lattice

- Poset (S, \leq)
- if Every pair of element has both a “least upper bound” and a “greatest lower bound”, then (S, \leq) is called
Lattice



Lattice

- Examples: which is lattice
 - $(\mathbb{Z}^+, |)$
 - $(\{1, 2, 3, 4, 5\}, |)$
 - $(\{1, 2, 4, 8, 16\}, |)$
 - $(P(S), \subseteq)$, S is a set



Lattice

- Let (S, \leq) be a lattice, Define
 - meet operation (\wedge) : $a \wedge b = \text{glb}(a, b)$
 - join operation (\vee) : $a \vee b = \text{lub}(a, b)$
- for all elements x, y, z of lattice S
 - $x \wedge y = y \wedge x, x \vee y = y \vee x$ (commutative law)
 - $(x \wedge y) \wedge z = x \wedge (y \wedge z)$ (associative law)
 - $x \wedge (x \vee y) = x, x \vee (x \wedge y) = x$ (absorption law)
 - $x \wedge x = x, x \vee x = x$ (idempotent law)



Lattice & Algebra structure

- Lattice (S, \leq)
- (S, \vee, \wedge) is an algebra structure derived from the lattice
 - \vee, \wedge operations have the properties of commutative laws, associative laws, absorption laws and idempotent laws



Distributive Lattice

- Give a lattice (S, \leq) , $\forall a, b, c \in S$,
 - $a \vee (b \wedge c) \leq (a \vee b) \wedge (a \vee c)$
 - $(a \wedge b) \vee (a \wedge c) \leq a \wedge (b \vee c)$
 - if
 - $a \vee (b \wedge c) = (a \vee b) \wedge (a \vee c)$
 - $(a \wedge b) \vee (a \wedge c) = a \wedge (b \vee c)$
- then (S, \leq) is distributive



Distributive Lattice

- Examples: which lattice is distributive
 - $(\mathbb{Z}^+, |)$
 - $(P(S), \subseteq)$, S is a set
 - (S, \subseteq) , $S = \{\emptyset, \{1\}, \{2\}, \{3\}, \{1,2\}, \{2,3\}, \{1,2,3\}\}$



bounded Lattice

- Give a lattice (S, \leq) ,
- if there is an Upper Bound, denoted by 1, $\forall a \in S$, such that $a \leq 1$; and if there is an Lower Bound, denoted by 0, $\forall a \in S$, such that $0 \leq a$, this lattice is bounded
 - $a \vee 1 = 1$; $a \wedge 1 = a$
 - $a \vee 0 = a$; $a \wedge 0 = 0$



Complement & bounded lattice

- Give a **Bounded lattice** (S, \leq) ,
- Complement of an element a is an element b such that
 - $a \vee b = 1; a \wedge b = 0$
- a lattice is complemented if every element has a complement.



Complement & bounded lattice

- Examples: is complemented?
 - $(P(S), \subseteq)$, S is a finite set
 - (S, \subseteq) , $S = \{\emptyset, \{1\}, \{2\}, \{3\}, \{1,2\}, \{2,3\}, \{1,2,3\}\}$



Lattice & Boolean Algebra

- A lattice is called Boolean Lattice if it is complemented and distributive.
- The algebra structure derived from the boolean lattice is called Boolean Algebra (S, \vee, \wedge, \sim) , " \sim " denotes a unary operation which compute the complement of an element.



Boolean Function

- Boolean set $\{0,1\}$
- 3 operations
 - Complementation $\bar{0} = 1 \quad \bar{1} = 0$
 - Boolean sum “denoted by + or OR”
 - $1+1=1 \quad 1+0=1 \quad 0+1=1 \quad 0+0=0$
 - Boolean product “denoted by . or AND”
 - $1 \cdot 1 = 1 \quad 1 \cdot 0 = 0 \quad 0 \cdot 1 = 0 \quad 0 \cdot 0 = 0$
 - Rules of precedence: complementation, AND, OR



Boolean Function

- Boolean variable
- Boolean function of degree n
 - From B^n to B
 - $B = \{0, 1\}$, $B^n = \{(x_1, x_2, \dots, x_n) \mid x_i \in B\}$
- Boolean expression
 - Are defined recursively as
 - $0, 1, x_1, x_2, \dots, x_n$ are boolean expression
 - If E is boolean expression, the complement of E is boolean expression
 - If E_1, E_2 are boolean expression then $E_1 E_2, E_1 + E_2$ are boolean expression



Boolean Function

- Each boolean expression represents a boolean function
 - $F(x,y,z)=xy+\sim z$ (using \sim as complement)
- Relationship between boolean expression and proposition logic expression
- Boolean value table



Boolean Function

- Boolean functions F and G of n variables are equal if and only if $F(b_1, b_2, \dots, b_n) = G(b_1, b_2, \dots, b_n)$. Two different Boolean expressions that represent the same function are called **equivalent**. Like $xy = xy + 0$
- How many different boolean functions of degree n are there?



Boolean Function

- Boolean identities
 - Table 5 on Page 753 of ver6
 - Commutation laws, associative laws, distributive laws, De Morgan's laws.....
- Duality
 - Interchanging Boolean sums and products, and interchanging 0s and 1s
 - Duality principle



Abstract definition of A Boolean algebra

- Definition: A Boolean algebra is a set B with two binary operations \vee and \wedge , elements 0 and 1 , and a unary operation \sim such that there properties hold for all x, y, z in B
 - Identity laws
 - Complement laws
 - Associative laws
 - Commutative laws
 - Distributed laws



Abstract definition of A Boolean algebra

- Boolean Algebra
 - Boolean expressions
 - Propositions
 - Sets



Representing boolean function

- Sum-of-Products expansions
 - Example: how to constructing a boolean expression representing a function with given values
 - Definition: A *literal* is a boolean variable or its complement. A *minterm* of the boolean variables x_1, x_2, \dots, x_n is a Boolean product $y_1 y_2 \dots y_n$ where $y_i = x_i$ or its complement
 - The minterm has the value 1 for one and only one combination of values of its variables



Representing boolean function

- Sum-of-Products expansions
 - The sum of minterms that represent the function is called the sum-of-products expansion or disjunctive normal form
 - Example: $F(x,y,z)=(x+y)\sim z$
- Product-of sum expansion or conjunctive normal form



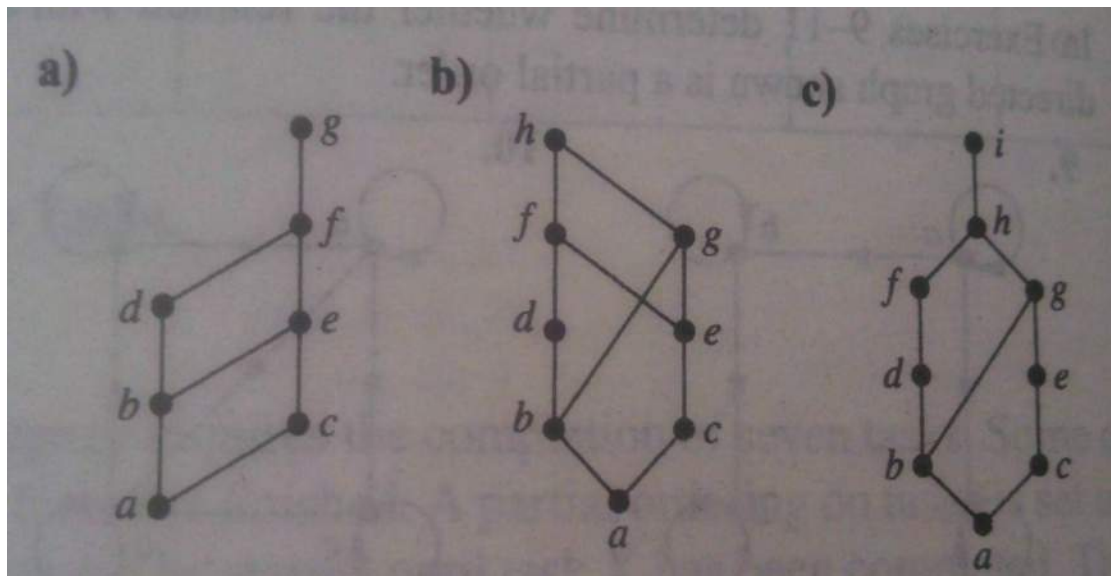
Representing boolean function

- Functional completeness
 - Every boolean function can be expressed as a Boolean sum of minterm, each minterm is the Boolean product of boolean variables or its complements. So boolean operators set "sum, product, complement" is called functional complete
 - Can we find a smaller set of functionally complete operators?

Homework

- Page 294 22 (*Page 580, 43 ver 6*)

Determine whether the posets with these Hasse diagrams are lattices





Homework

- Page 294 23 (*Page 580, 45 ver 6*)

Show that every nonempty finite subset of a lattice has a least upper bound and a greatest lower bound.



Homework

- Show that De Morgan's laws hold in a boolean algebra. That is, show that for all x and y ,

$$\overline{(x \vee y)} = \bar{x} \wedge \bar{y}$$

$$\overline{(x \wedge y)} = \bar{x} \vee \bar{y}$$



Fundament of Logic (1)

Wu Gang
School of Software, SJTU



Logic

- deductive reasoning (演绎推理)
 - 前提和结论之间的可推导性
- Inductive reasoning (归纳推理)
 - 样本到总体的一个推理
 - 所有观察到的乌鸦都是黑的
 - 所以，所有乌鸦都是黑的



deductive reasoning

- 研究形式上的可推导性关系
 - 所有大学生都听音乐之声（前提）
 - 张三不听音乐之声（前提）
 - 张三不是大学生（结论）
- 这个推导过程是正确的，虽然前提 / 结论不一定为真
- 推理是否正确与命题的真假无关，跟逻辑形式有关



deductive reasoning

- 研究形式上的可推导性关系
 - S 中的元素都有性质 R （前提）
 - a 没有性质 R （前提）
 - a 不是 S 中的元素（结论）
- 这就是前面例子的逻辑形式，跟 S 到底是什么无关



deductive reasoning

- 形式语言的必要性(Formal Language)
 - 他认识3班某同学
 - 3班某同学是足球队长
 - 他认识足球队长
- 自然语言描述可能产生误解，可以构造一种符号语言来精确描述命题的逻辑形式（形式语言）



deductive reasoning

- 命题逻辑

- 命题: P 、 Q 、 R
- 逻辑连接词: \neg 、 \wedge 、 \vee 、 \rightarrow 、...

- 谓词逻辑

- 谓词: $S(t)$, t 是项 (term)
- 量词: $\forall x$ 、 $\exists x$
- 个体: a 、 b 、 c (常量)、... u 、 v (变元)
- 个体函数: f 、 g ...
- 构造规则: 项的构造、公式的构造



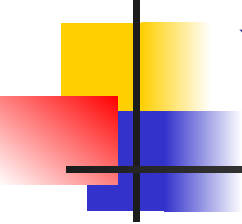
形式语言

- 语法和语义
 - 语法只涉及符号和表达式的形式结构
 - 而给符号赋予某种解释时，就有语义了
- 谓词逻辑公式的解释（赋值）
 - 论域
 - 谓词函数
 - 个体函数
 - 命题真值



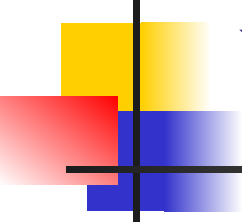
命题逻辑与谓词逻辑的性质

- **the soundness property**（可靠性）
 - 语法的推演反映出的关系在语义上是成立的
 - 形式推演对于语义的推理是可靠的
 - 形式推演的可靠性
- **The completeness property**（完备性）
 - 凡是语义推理成立的关系，语法的推演都是可以反映的
 - 形式推演的完备性



可满足性与有效性

- 语义概念
- **可满足的**：一个公式或者公式集合是可满足的，当且仅当有一个赋值（论域不为空）使得公式（或集合中每一个公式）为T。一般用 Σ 表示公式集合，大写字母A、B、P、Q标识公式
- **有效的**：公式A是有效的，当且仅当任何赋值都使之成为T（论域不为空）



可满足性与有效性

- 在论域**D**上是可满足的：当且仅当在论域**D**上有一个赋值使之为**T**
- 在论域**D**上是有效的：当且仅当在论域**D**上的任何赋值都使之为**T**
- Σ 在论域**D**上是可满足的，则 Σ 是可满足的；
- **A**是有效的，则**A**在论域**D**上是有效的



可满足性与有效性

■ 定理

- A 是可满足的，当且仅当 $\neg A$ 不是有效的
- A 是有效的，当且仅当 $\neg A$ 是不可满足的
- $A(u_1, u_2, \dots, u_n)$ 是可满足的，当且仅当
 $\exists x_1 \dots \exists x_n A(x_1, \dots, x_n)$ 是可满足的
- $A(u_1, u_2, \dots, u_n)$ 是有效的，当且仅当 $\forall x_1 \dots \forall x_n A(x_1, \dots, x_n)$ 是有效的



前束范式

- 谓词逻辑的一种范式
 - $Q_1x_1 \dots Q_nx_n A$, 其中 $Q_1 \dots Q_n$ 为量词
- 一个公式可以变换为某个前束范式, 且与之等值 \equiv
- Prenex normal form



前束范式

- $\neg \forall x A(x) \equiv \exists x \neg A(x)$
- $\neg \exists x A(x) \equiv \forall x \neg A(x)$
- $A \wedge Qx B(x) \equiv Qx (A \wedge B(x))$, x 不在 A 中出现
- $A \vee Qx B(x) \equiv Qx (A \vee B(x))$, x 不在 A 中出现
- $\forall x A(x) \wedge \forall x B(x) \equiv \forall x (A(x) \wedge B(x))$
- $\exists x A(x) \vee \exists x B(x) \equiv \exists x (A(x) \vee B(x))$
- $Q_1 x A(x) \vee Q_2 y B(y) \equiv Q_1 x Q_2 y (A(x) \vee B(y))$
- $Q_1 x A(x) \wedge Q_2 y B(y) \equiv Q_1 x Q_2 y (A(x) \wedge B(y))$



前束范式的可满足性与有效性

- 定理

- A 是可满足的，当且仅当 A 的前束范式是可满足的
- A 是有效的，当且仅当 A 的前束范式是有效的



homework

- Page 122, 3.6.1

- 将下列公式变换成前束范式
- $(\neg \exists x F(x) \vee \forall y G(y)) \wedge (F(u) \rightarrow \forall y G(y))$
- $\exists x F(u, x) \leftrightarrow \forall y G(y)$
- $\forall x (F(x) \rightarrow \exists y (G(y) \rightarrow F(x) \vee \forall z G(z)))$



Fundament of Logic (2)

Wu Gang
School of Software, SJTU



可靠性（谓词逻辑、命题逻辑）

- 定理：设 $\Sigma \subseteq \text{Form}(L)$ ， $A \in \text{Form}(L)$ 。
 - 若 $\Sigma \vdash A$ ，则 $\Sigma \models A$ 。
 - 若 $\Phi \vdash A$ ，则 $\Phi \models A$
- 即：所有形式可证明的公式都是有效的
- 一阶逻辑有**17**条推理规则，要证明每一条规则都满足可靠性



17条推理规则

- $A \vdash A$ (Ref)
- if $\Sigma \vdash A$, then $\Sigma, \Sigma' \vdash A$ (+)
- if $\Sigma, \neg A \vdash B$ and $\Sigma, \neg A \vdash \neg B$,
then $\Sigma \vdash A$ ($\neg -$)
- if $\Sigma \vdash A \rightarrow B$, $\Sigma \vdash A$, then $\Sigma \vdash B$ ($\rightarrow -$)
- if $\Sigma, A \vdash B$, then $\Sigma \vdash A \rightarrow B$ ($\rightarrow +$)
- if $\Sigma \vdash A \wedge B$, then $\Sigma \vdash A$ and $\Sigma \vdash B$ ($\wedge -$)
- if $\Sigma \vdash A$ and $\Sigma \vdash B$, then $\Sigma \vdash A \wedge B$ ($\wedge +$)



17条推理规则

- if $\Sigma, A \vdash C$ and $\Sigma, B \vdash C$, then $\Sigma, A \vee B \vdash C$ ($\vee -$)
- if $\Sigma \vdash A$, then $\Sigma \vdash A \vee B$, $\Sigma \vdash B \vee A$ ($\vee +$)
- if $\Sigma \vdash A \leftrightarrow B$, $\Sigma \vdash A$, then $\Sigma \vdash B$ ($\leftrightarrow -$)
- if $\Sigma, A \vdash B$, $\Sigma, B \vdash A$, then $\Sigma \vdash A \leftrightarrow B$ ($\leftrightarrow +$)
- if $\Sigma \vdash \forall x A(x)$, then $\Sigma \vdash A(t)$ 代入 ($\forall -$)
- if $\Sigma \vdash A(u)$, u 不在 Σ 中出现
then $\Sigma \vdash \forall x A(x)$ 代入 ($\forall +$)



17条推理规则

- if $\Sigma, A(u) \vdash B$, u 不在 Σ 或 B 中出现
then $\Sigma, \exists xA(x) \vdash B$ 代入
($\exists -$)
- if $\Sigma \vdash A(t)$ then $\Sigma \vdash \exists xA(x)$ ($\exists +$)
 x 替换 t (可以只替换部分)
- if $\Sigma \vdash A(t_1)$, $\Sigma \vdash t_1 \equiv t_2$ then $\Sigma \vdash A(t_2)$
 t_2 替换 t_1 (可以只替换部分) ($\equiv -$)
- $\Phi \vdash u \equiv u$ ($\equiv +$)



证明可靠性定理

- (Ref): $A \vdash A$ 显然成立
- (+): if $\Sigma \vdash A$, then $\Sigma, \Sigma' \vdash A$
要证: if $\Sigma \models A$, then $\Sigma, \Sigma' \models A$
- (\neg —): 要证 if $\Sigma, \neg A \models B$ and $\Sigma, \neg A \models \neg B$, then $\Sigma \models A$ 反证法
- (\vee —): 要证 if $\Sigma, A \models C$ and $\Sigma, B \models C$, then $\Sigma, A \vee B \models C$



证明可靠性定理

- $(\exists -)$: 要证 if $\Sigma, A(u) \models B$, u 不在 Σ 或 B 中出现, then $\Sigma, \exists x A(x) \models B$
 - 令 v 为以 D 为论域的任何赋值, 使 $\Sigma^v, \exists x A(x)^v$ 均为真, 则有 $a \in D$, 使 $A(a)$ 即 $A(u)^{v(u=a)}$ 为真; 由于 u 不在 Σ 中出现, 因此 $\Sigma^{v(u=a)} = \Sigma^v$ 为真; 因此 $\Sigma, A(u)$ 在解释 $v(u=a)$ 下均为真, 得到 $B^{v(u=a)}$ 为真, 又由于 u 不在 B 中出现, 则



错误推理

- $(\forall +)$ u 不在 Σ 中出现
 - “if $\Sigma \vdash A(u)$, then $\Sigma \vdash \forall xA(x)$ ”
 - if $A(u) \vdash A(u)$, then $A(u) \vdash \forall xA(x)$ 不对
- $(\exists -)$ u 不在 Σ 或 B 中出现
 - “if $\Sigma, A(u) \vdash B$, then $\Sigma, \exists xA(x) \vdash B$ ”
 - if $A(u) \vdash A(u)$, then $\exists xA(x) \vdash A(u)$ 不对



错误推理的说明

- 易证：
 - $A(u) \not\models \forall xA(x)$
 - $\exists xA(x) \not\models A(u)$
- 那么由可靠性定理得
 - $A(u) \not\vdash \forall xA(x)$
 - $\exists xA(x) \not\vdash A(u)$
 - 意思是可以证明这是无法推出的！！



协调性

- 定义： $\Sigma \subseteq \text{Form}(L)$ 是协调的，当且仅当不存在 $A \in \text{Form}(L)$ ，使得 $\Sigma \vdash A$ 并且 $\Sigma \vdash \neg A$
 - 语法概念



可靠性定理

- 另一种表达：语义和语法之间
- 设 $\Sigma \subseteq \text{Form}(L)$, $A \in \text{Form}(L)$,
 - 如果 Σ 是可满足的, 则 Σ 是协调的;
 - 如果 A 是可满足的, 则 A 是协调的。



协调性的一个定理

- $\Sigma \subseteq \mathbf{Form}(L)$ 是协调的，当且仅当存在 A ，使得 $\Sigma \vdash A$



homework

- 证明上一页的定理
- 证明：可靠性定理证明中的
 $(\rightarrow -)$ 、 $(\wedge +)$ 、 $(\forall +)$ 情况



Fundament of Logic (3)

Wu Gang
School of Software, SJTU



极大协调性

- 定义：公式集 Σ 是极大协调的，当且仅当 Σ 满足以下两点，
 - Σ 是协调集
 - 对于任何 $A \notin \Sigma$ ， $\Sigma \cup \{A\}$ 是不协调的。
即不存在以 Σ 为真子集的协调集
- 如果对于任何 A ， $\Sigma \vdash A$ 蕴涵 $A \in \Sigma$ ，称 Σ 是封闭于形式可推演的。



极大协调性

- 定理1：设 Σ 是极大协调集，那么对于任何 A ， $\Sigma \vdash A$ 当且仅当 $A \in \Sigma$
 - 设 $\Sigma \vdash A$ ，如果 $A \notin \Sigma$ ，由 Σ 是极大协调的可知 $\Sigma \cup \{A\}$ 是不协调的。因此 $\Sigma \vdash \neg A$ ，导出 Σ 不协调，矛盾！因此， $A \in \Sigma$ 。
 - 设 $A \in \Sigma$ ，则由推理规则可到 $\Sigma \vdash A$
- 极大协调集 Σ 是封闭于形式可推演的



极大协调性

- 定理2: 设 Σ 是极大协调集, 则对于任何 A 和 B , 有
 - $\neg A \in \Sigma$, 当且仅当 $A \notin \Sigma$;
 - $A \wedge B \in \Sigma$, 当且仅当 $A \in \Sigma$ 且 $B \in \Sigma$
 - $A \vee B \in \Sigma$, 当且仅当 $A \in \Sigma$ 或 $B \in \Sigma$
 - $A \rightarrow B \in \Sigma$, 当且仅当 $A \in \Sigma$ 蕴涵 $B \in \Sigma$
 - $A \leftrightarrow B \in \Sigma$, 当且仅当 $A \in \Sigma$ 等值于 $B \in \Sigma$



极大协调性

- 推论：设 Σ 是极大协调集，则对于任何 A ， $\Sigma \vdash \neg A$ 当且仅当 $\Sigma \not\vdash A$
 - $\Sigma \vdash \neg A$ ，若 $\Sigma \vdash A$ ， Σ 不协调，矛盾！
 - $\Sigma \not\vdash A$ ，则 $A \notin \Sigma$ ，则 $\neg A \in \Sigma$ ，则 $\Sigma \vdash \neg A$

极大协调集

- 定理 (Lindenbaum) : 任何协调的公式集 Σ 能够扩充为极大协调集。记为 Σ^*

- 设 $\text{Form}(L)$ 是可数无限集, 令 A_0, A_1, A_2, \dots 是其一个排列。定义 $\Sigma_n (n \geq 0)$ 如下:

$$\Sigma_0 = \Sigma, \quad \Sigma_{n+1} = \Sigma_n \cup \{A_n\} \text{ (若 } \Sigma_n \cup \{A_n\} \text{ 协调),}$$

$$\text{否则 } \Sigma_{n+1} = \Sigma_n$$

- 则有 $\bigcup_{n \in \mathbb{N}} \Sigma_n \subseteq \Sigma_{n+1}$, 且 Σ_n 是协调的。

- $\Sigma^* = \bigcup_{n \in \mathbb{N}} \Sigma_n$ 先证 Σ^* 是协调的;

再证 Σ^* 是极大的



极大协调集

- 定理（Lindenbaum）：任何协调的公式集 Σ 能够扩充为极大协调集。记为 Σ^*
 - 注明：一阶语言不可数时， $\text{Form}(L)$ 是不可数无限集。这种情况不做要求。



命题逻辑的完备性

- 完备性证明有多种不同的方法
- 方法一的引理

设 $A \in \text{Form}(L^p)$ 含不同的原子公式 p_1, p_2, \dots, p_n , v 是真假赋值。对于 $i=1, 2, \dots, n$, 令

若 $p_i^v=1$, 则 $A_i=p_i$; 否则, $A_i=\neg p_i$ 。则有

- 若 $A^v=1$, 则 $A_1, A_2, \dots, A_n \vdash A$ 。
- 若 $A^v=0$, 则 $A_1, A_2, \dots, A_n \vdash \neg A$



命题逻辑的完备性

- 方法一的定理：设 $A \in \text{Form}(L^p)$, $\Sigma \subseteq \text{Form}(L^p)$, 且 Σ 是有限集, 则
 - 若 $\Phi \models A$, 则 $\Phi \vdash A$ (A 是重言式的话, 则 A 是形式上可证明的)
 - 若 $\Sigma \models A$, 则 $\Sigma \vdash A$



命题逻辑的完备性

- 证明若 $\Phi \models A$, 则 $\Phi \vdash A$:

设 A 含不同的原子公式 p_1, p_2, \dots, p_n 。令 v_1, v_2 是真假赋值, 且 $p_1^{v_1}=1, p_1^{v_2}=0, p_2^{v_1}=p_2^{v_2}=1, p_i^{v_1}=p_i^{v_2} (i=3, \dots, n)$ 。由引理, 由于 $A^{v_1}=1$, 所以 $p_1, p_2, \dots, A_n \vdash A$ 。由于 $A^{v_2}=1$, 所以 $\neg p_1, p_2, \dots, A_n \vdash A$ 。可得, $p_1 \vee \neg p_1, p_2, \dots, A_n \vdash A$, $p_2, \dots, A_n \vdash (p_1 \vee \neg p_1) \rightarrow A$, 又因为 $\Phi \vdash p_1 \vee \neg p_1$, 所以 $p_2, \dots, A_n \vdash p_1 \vee \neg p_1$ 。可得, $p_2, \dots, A_n \vdash A$ 。

再令 u_1, u_2 是真假赋值, 且 $p_1^{u_1}=1, p_1^{u_2}=0, p_2^{u_1}=p_2^{u_2}=0, p_i^{u_1}=p_i^{u_2}, i=3, \dots, n$ 。同理可得, $\neg p_2, \dots, A_n \vdash A$ 。

可得, $A_3, \dots, A_n \vdash A$ 。进一步指定真假赋值, 可得 $\Phi \vdash A$ 。



命题逻辑的完备性

- 方法二的引理：
 - $\Sigma^* \subseteq \text{Form}(L^p)$ 是极大协调集，构造真假赋值 v 使得对任何原子公式 p ， $p^v = 1$ 当且仅当 $p \in \Sigma^*$ ，于是对于任何的 $A \in \text{Form}(L^p)$ ， $A^v = 1$ 当且仅当 $A \in \Sigma^*$



命题逻辑的完备性

- 方法二的定理1： 设 $\Sigma \subseteq \text{Form}(L^p)$, $A \in \text{Form}(L^p)$, 则有
 - 如果 Σ 是协调的, 则 Σ 是可满足的;
 - 如果 A 是协调的, 则 A 是可满足的;
- 证明: Σ 是协调的, 任取 $B \in \Sigma$, 把 Σ 扩为极大协调集 Σ^* , 使用引理中的真假赋值 v , 就有 $B^v = 1$ 。因此, v 满足 Σ 。



命题逻辑的完备性

- 方法二的定理2： 设 $\Sigma \subseteq \text{Form}(L^p)$, $A \in \text{Form}(L^p)$, 则有
 - 若 $\Sigma \models A$, 则 $\Sigma \vdash A$
 - 若 $\Phi \models A$, 则 $\Phi \vdash A$ (A 是重言式的话, 则 A 是形式上可证明的)
- 证明: $\Sigma \models A$, 则 $\Sigma \cup \{\neg A\}$ 是不可满足的。又定理1可知 $\Sigma \cup \{\neg A\}$ 是不协调的。因此 $\Sigma \vdash A$



一阶逻辑的完备性

- 经典一阶逻辑就是指谓词逻辑
- 我们只需要大家了解结论，不需要大家掌握证明。



一阶逻辑的完备性

- 一阶完备性定理1： 设 $\Sigma \subseteq \mathbf{Form}(L)$, $A \in \mathbf{Form}(L)$, 则有
 - 如果 Σ 是协调的, 则 Σ 是可满足的;
 - 如果 A 是协调的, 则 A 是可满足的;



一阶逻辑的完备性

- 一阶完备性定理2： 设 $\Sigma \subseteq \text{Form}(L)$, $A \in \text{Form}(L)$, 则有
 - 若 $\Sigma \models A$, 则 $\Sigma \vdash A$
 - 若 $\Phi \models A$, 则 $\Phi \vdash A$(A 是有效公式的话, 则 A 是形式上可证明的)



homework

- 设 Σ 是封闭于形式可推演的，证明 Σ 是极大协调的，当且仅当对于任何的 A ， Σ 包含且只包含 A 和 $\neg A$ 之一。
- 证明：命题逻辑完备性方法二引理中 $A = B \vee C$ ，和 $B \rightarrow C$ 的情况。



Cardinality of Set

- Set A and Set B have the same cardinality if and only if there is a one-to-one correspondence from A to B
- Finite Set, Infinite Set
 - Infinite set has a subset with same cardinality



Countable Set

- Countable: cardinality is \aleph_0
 - Finite Set
 - Infinite Set with the same cardinality as the set of positive integers
- Examples
 - Set of odd positive integers
 - Set of positive rational numbers
 - Set of Integers ?



Uncountable Set

- Set of real numbers
- Diagonalization Argument
- $(0,1)$ vs \mathbb{R}
 - $\tan \pi (2x-1)/2$
- $(0,1)$ vs $[0,1]$?
- Continuum hypothesis



homework

- Prove $(0,1)$ and $[0,1]$ have the same cardinality
- Show that the Union of two countable set is also countable