Discrete Mathematics

The Foundations: Logic and Proofs

School of Computer Science

Xiao Zhang(zhangxiao@nwpu.edu.cn)

QQ:27728221

Contents

- 1.1 Propositional Logic
- 1.2 Applications of Propositional Logic
- 1.3 Propositional Equivalences
- 1.4 Predicates and Quantifiers
- 1.5 Nested Quantifiers
- 1.6 Rules of Inference
- 1.7 Introduction to Proofs
- 1.8 Proof Methods and Strategy

Logic

Proof

Foundations of Logic

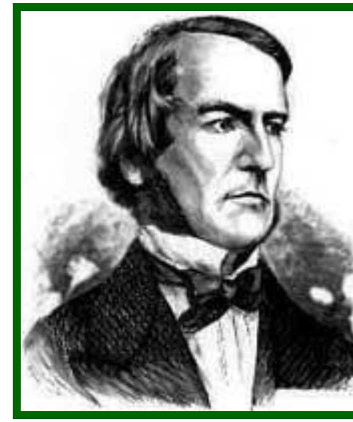
- Rosen 7th ed., §§1.1-1.5
- Propositional logic (§ 1.1-1.3): 命题逻辑
  - Basic definitions. (§ 1.1)
  - Applications (§ 1.2)
  - Equivalence rules (§ 1.3)
- Predicate logic (§ 1.4-1.5) 谓词逻辑
  - Predicates.
  - Quantified predicate expressions.
  - Equivalences

Propositional Logic (§ 1.1)

Propositional Logic is the logic of compound statements built from simpler statements using so-called *Boolean connectives.*

Some applications in computer science:

- Design of digital electronic circuits.
- Expressing conditions in programs.
- Queries to databases & search engines.



George Boole (1815-1864)

Propositions in natural language

Definition: A *proposition* is simply:

- a *statement* (*i.e.*, a declarative sentence)
  - *with some definite meaning*, (not vague or ambiguous)
- having a *truth value* that's either *true* (**T**) or *false* (**F**)
  - it is **never** both, neither, or somewhere "in between!"
    - However, you might not *know* the actual truth value
- Later, we will study *probability theory,* in which we assign *degrees of certainty* ("between" **T** and **F**) to propositions.
  - But for now: think True/False only!

Propositions（命题）

- A **statement** or **proposition** is a declarative sentence（陈述句） that is either *true* or *false*,but not both.
  - true = T (or 1)
  - false = F (or 0) (binary logic)

Example

- ‘The moon is made of green cheese’
- ‘go to town!’
- X - *imperative*
- ‘What time is it?’
- X - *interrogative*

Examples of NL Propositions

- "It is raining." (In a given situation.)
- "Beijing is the capital of China, and 1 + 2 = 2"

But, the following are **NOT** propositions:

- "La la la la la." (no meaning)
- "Who's there?" (interrogative: no truth value)
- "x := x+1" (imperative: no truth value)
- "1 + 2" (term: no truth value)

Example 1

- Which of the following are statements?
  - (a) The earth is round.
  - (b) 2+3=5
  - (c) Do you speak English?
  - (d) 3-*x*=5
  - (e) Take two aspirins.
  - (f) The temperature on the surface of the planet Venus is 800ºF.
  - (g) The sun will come out tomorrow.

Propositional variables

- We use letters to denote **propositional variables** (or statement variables), that is, variables that represent propositions, just as letters are used to denote numerical variables. The conventional letters used for propositional variables are p, q, r, s, . . .
- The area of logic that deals with propositions is called the **propositional calculus** or **propositional logic**.

Propositions in Propositional Logic

- Atoms: *p*, *q*, *r*, …
  (Corresponds with simple English sentences, e.g. \`I had salad for lunch')
- Complex propositions : built up from atoms using operators: $p \wedge q$
  (Corresponds with compound English sentences, e.g., "I had salad for lunch **and** I had steak for dinner.")

propositions

Atoms propositions： can not be divided to smaller propositions

Complex propositions： built up from atoms or complex propositions using operators

Operators / Connectives

An *operator* or *connective* combines *n* *operand* expressions into a larger expression. (*E.g.*, "+" in numeric exprs.)

- *Unary* operators take 1 operand (*e.g.*, −3); *binary* operators take 2 operands (*eg* 3×4).
- *Propositional* or *Boolean* operators operate on propositions instead of on numbers.

Logical Operators(逻辑运算）

- Unary
  - **Negation（否定）**
- Binary
  - **Conjunction（合取）**
  - **Disjunction （析取）**
  - **Exclusive OR （异或）**
  - **Implication （蕴涵）**
  - **Biconditional （等价）**

Some Popular Boolean Operators

Formal Name	Nickname	Arity	Symbol
Negation operator	NOT	Unary	$\neg$
Conjunction operator	AND	Binary	$\wedge$
Disjunction operator	OR	Binary	$\vee$
Exclusive-OR operator	XOR	Binary	$\oplus$
Implication operator	IMPLIES	Binary	$\rightarrow$
Biconditional operator	IFF	Binary	$\leftrightarrow$

The Negation Operator

The unary *negation operator* "¬" (*NOT*) transforms a prop. into its *negation*.

E.g. If p = "I have brown hair."

then $\neg p$ = "I do **not** have brown hair."

The *truth table* 真值表for NOT:

p	$\neg p$
T	F
F	T
Operand column	Result column

T :≡ True; F :≡ False

":≡" means "is defined as"

Some concepts

- Proposition
- atomic proposition 原子命题 VS compositional proposition 复合命题
- propositional constant 命题常量 VS 命题变量

(propositional variable)

- 复合命题compositional proposition VS propositional formulas

真值表TRUTH TABLES

- Write all possible truth value of proposition variables and correspond value of the whole formula. This will help us know every condition of the formula.

p	$\neg p$
T	F
F	T

Operand column | Result column

Operand columns

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

The Conjunction合取 Operator

The binary *conjunction operator* "$\wedge$" (*AND*) combines two propositions to form their logical *conjunction*.

E.g. If *p*="I will have salad for lunch." and *q*="I will have steak for dinner.", then $p \wedge q$="I will have salad for lunch **and** I will have steak for dinner."

Conjunction Truth Table

- Note that a conjunction $p_1 \wedge p_2 \wedge \ldots \wedge p_n$ of n propositions will have 2^n rows in its truth table.

Operand columns

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

- Also: ¬ and ∧ operations together are sufficient to express *any* Boolean truth table!

The Disjunction析取 Operator

The binary *disjunction operator* "$\vee$" (*OR*) combines two propositions to form their logical *disjunction*.

p="My car has a bad engine."

q="My car has a bad door."

$p \vee q$="Either my car has a bad engine, **or** my car has a bad door."

Meaning is like "and/or" in English.

Disjunction Truth Table

- Note that $p \vee q$ means that p is true, or q is true, **or both** are true!
- So, this operation is also called *inclusive or,* because it **includes** the possibility that both p and q are true.
- "$\neg$" and "$\vee$" together are also universal.

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

Note difference from AND

Let's introduce some additional connectives

- A variant of disjunction
- The conditional
- The biconditional

The *Exclusive Or* Operator

The binary *exclusive-or operator* "$\oplus$" (*XOR*) combines two propositions to form their logical "exclusive or".

p = "I will earn an A in this course,"

q = "I will drop this course,"

$p \oplus q$ = "I will either earn an A in this course, or I will drop it (but not both!)"

Exclusive-Or Truth Table

- Note that $p \oplus q$ means that p is true, or q is true, but **not both**!
- This operation is called *exclusive or,* because it **excludes** the possibility that both p and q are true.
- "$\neg$" and "$\oplus$" together are **not** universal.

p	q	$p \oplus q$
F	F	F
F	T	T
T	F	T
T	T	**F**

Note difference from OR.

Natural Language is Ambiguous

Note that <u>English</u> "or" can be <u>ambiguous</u> regarding the "both" case!

p	q	p "or" q
F	F	F
F	T	T
T	F	T
T	T	?

Need context to disambiguate the meaning!

For this class, assume "or" means <u>inclusive</u>.

The *Implication* Operator

The *implication* $\overbrace{p}^{\text{antecedent}} \rightarrow \overbrace{q}^{\text{consequent}}$ states that p implies q.

I.e., If p is true, then q is true; but if p is not true, then q could be either true or false.

E.g., let p = "You study hard."
q = "You will get a good grade."

$p \rightarrow q$ = "If you study hard, then you will get a good grade."

Implication Truth Table

- $p \rightarrow q$ is **false** <u>only</u> when p is true but q is **not** true.
- $p \rightarrow q$ does **not** say that p <u>causes</u> q!
- $p \rightarrow q$ does **not** require that p or q **<u>are true</u>**!
- *E.g.* "(1=0) $\rightarrow$ pigs can fly" is TRUE!

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

The <u>only</u> False case!

Different Ways of Expressing $p \to q$

- if p, then q
- if p, q
- q unless $\neg p$
- q if p
- q whenever p
- q follows from p
- p implies q
- p only if q
- q when p
- q when p
- p is sufficient for q
- q is necessary for p

- a necessary condition for p is q
- a sufficient condition for q is p

English Phrases Meaning $p \to q$

- "p implies q"
- "if p, then q"
- "if p, q"
- "when p, q"
- "whenever p, q"
- "q if p"
- "q when p"
- "q whenever p"
- "p only if q"
- "p is sufficient for q"
- "q is necessary for p"
- "q follows from p"
- "q is implied by p"
- "q unless $\neg p$"

Converse, Contrapositive, and Inverse

- From $p \to q$ we can form new conditional statements .
  - $q \to p$ is the **converse** of $p \to q$ （逆命题)
  - $\neg q \to \neg p$ is the **contrapositive** of $p \to q$（逆反命题)
  - $\neg p \to \neg q$ is the **inverse** of $p \to q$(反命题)

Example: Find the converse, inverse, and contrapositive of “It raining is a sufficient condition for my not going to town.”

Solution:

converse: If I do not go to town, then it is raining.

inverse: If it is not raining, then I will go to town.

contrapositive: If I go to town, then it is not raining.

Contrapositive

Some terminology, for an implication $p \rightarrow q$:

- Its *converse* is: $q \rightarrow p$.
- Its *contrapositive*: $\neg q \rightarrow \neg p$.
- One of these has the *same meaning* (same truth table) as $p \rightarrow q$. Can you figure out which?

原命题为：p——>q

converse 逆命题为：q——>p

inverse 否命题为：非p——>非q

逆否命题为：非q——>非p

Contrapositive

How do we know for sure?

Proving the equivalence of $p \rightarrow q$ and its contrapositive using truth tables:

p	q	$\neg q$	$\neg p$	$p \rightarrow q$	$\neg q \rightarrow \neg p$
F	F	T	T	T	T
F	T	F	T	T	T
T	F	T	F	F	F
T	T	F	F	T	T

The *biconditional* operator

The *biconditional* $p \leftrightarrow q$ states that p is true *if and only if (IFF)* q is true.

p = "Barack Obama won the *2012* presidential *election*."

q = "Barack Obama was president for all of 2013."

$p \leftrightarrow q$ = "If, and only if, Barack Obama won the *2012* presidential *election*, Barack Obama was president for all of 2013."

Biconditional Truth Table

- $p \leftrightarrow q$ means that p and q have the **same** truth value.
- Note this truth table is the exact **opposite** of $\oplus$'s!

  Thus, $p \leftrightarrow q$ means $\neg(p \oplus q)$
- $p \leftrightarrow q$ does **not** imply that p and q are true, or that either of them causes the other.

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

Boolean Operations Summary

- We have seen 1 unary operator (out of the 4 possible) and 5 binary operators:

p	q	$\neg p$	$p \wedge q$	$p \vee q$	$p \oplus q$	$p \rightarrow q$	$p \leftrightarrow q$
F	F	T	F	F	F	T	T
F	T	T	F	T	T	T	F
T	F	F	F	T	T	F	F
T	T	F	T	T	F	T	T

Questions for you to think about

1. Consider a conjunction $p_1 \wedge p_2 \wedge p_3$
   How many rows are there in its truth table?
2. Consider a conjunction
   $p_1 \wedge p_2 \wedge \ldots \wedge p_n$ of n propositions.
   How many rows are there in its truth table?

3. Explain why $\neg$ and $\wedge$ together are sufficient to express *any* Boolean truth table

Questions for you to think about

1. Consider a conjunction $p_1 \wedge p_2 \wedge p_3$
   How many rows are there in its truth table? 8

$p_1 \wedge$	$p_2 \wedge$	p_3
1	1	1
1	1	0
1	0	1
1	0	0
0	1	1
0	1	0
0	0	1
0	0	0

Questions for you to think about

2. Consider $p_1 \wedge p_2 \wedge \ldots \wedge p_n$

   How many rows are there in its truth table?

   2*2*2* … *2 (n factors)

   Hence 2^n (This grows exponentially!)

Questions for you to think about

3. Explain why ¬ and ∧ together are sufficient to express *any*thing that propositional logic may express

- Obviously, if we add new connectives (like ∨) we can write new formulas
- But these formulas would always be equivalent with ones that only use ¬ and ∧ (This what we need to prove)

- Saying this in a different way: if we add new connectives, we can write new formulas, but these formulas will always only express truth functions that can already be expressed by formulas that only use $\neg$ and $\wedge$.

- Example of writing a disjunction in another form:

$$p \vee q \Leftrightarrow \neg(\neg p \wedge \neg q)$$

For you to think about:

1. Can you think of yet another 2-place connective?
   How many possible connectives do there exist?

   - $\vee$
   - $\wedge$
   - $\oplus$
   - ->
   - $\leftrightarrow$
   - ？ ？ ？

For you to think about:

1. How many possible connectives do there exist?

p	**connective**	**q**
T	?	T
T	?	F
F	?	T
F	?	F

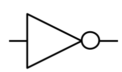
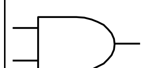
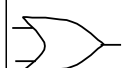
*Each question mark can be T or F, hence 2*2*2*2=16 connectives*

Example of another connective

p	**connective**	**q**	compare: **p and q**
T	F	T	T
T	T	F	F
F	T	T	F
F	T	F	F

Names: NAND, Sheffer stroke

Some Alternative Notations

Name:	not	and	or	xor	implies	iff
Propositional logic:	$\neg$	$\wedge$	$\vee$	$\oplus$	$\rightarrow$	$\leftrightarrow$
Boolean algebra:	$\overline{p}$	pq	$+$	$\oplus$		
C/C++/Java (wordwise):	!	&&	||	!=		==
C/C++/Java (bitwise):	~	&	|	^		
Logic gates:						

The language of propositional logic defined more properly

- Atoms: p1, p2, p3, ..
- Formulas:命题公式
  - All atoms are formulas
  - If α is a formula then $\neg\,\alpha$ is a formula
  - If α and β are formulas then the following are formulas: $(\alpha \wedge \beta)$, $(\alpha \vee \beta)$, $(\alpha \rightarrow \beta)$ *(etc.)*
- *Examples of formulas:*
  $(p1 \vee \neg\, p2)$, $\neg\,\neg\,\neg(p9 \rightarrow p8)$, $(p1 \wedge (p2 \vee p3))$
- *etc.*

The language of propositional logic defined more properly

- *Examples of formulas:*
  $(p1 \vee \neg p2)$, $\neg\neg\neg(p9 \rightarrow p8)$, $(p1 \wedge (p2 \vee p3))$
- *Convention 1:* outermost brackets are omitted,:
  $p1 \vee \neg p2$, $\neg\neg\neg(p9 \rightarrow p8)$, $p1 \wedge (p2 \vee p3)$
- *Convention 2:* associativity allows us to omit even more brackets, e.g.:
  $p1 \wedge p2 \wedge p3$, $p1 \vee p2 \vee p3$

Precedence of logical operators
Nested Propositional Expressions

- Use parentheses to *group sub-expressions*:
  "I just saw my old *f*riend, and either he's *g*rown or I've *s*hrunk." = $f \wedge (g \vee s)$
  - $(f \wedge g) \vee s$ would mean something different
  - $f \wedge g \vee s$ would be ambiguous
- By convention, "¬" takes *precedence* over both "∧" and "∨".
  - $\neg s\, f$ means $(\neg s) \wedge f$, **not** $\neg (s \wedge f)$
- Priority level
  - 1 ¬ 2 ∧ ∨ 3 → ↔

Logic as shorthand for NL

Let p="It rained last night",
q="The sprinklers came on last night,"
r="The lawn was wet this morning."

$\neg p$ = "It didn't rain last night."

$r \wedge \neg p$ = "The lawn was wet this morning, and it didn't rain last night."

$\neg r \vee p \vee q$ = "Either the lawn wasn't wet this morning, or it rained last night, or the sprinklers came on last night."

Truth Tables For Compound Propositions

- Construction of a truth table:
- Rows
  - Need a row for every possible combination of values for the atomic propositions.
- Columns
  - Need a column for the compound proposition (usually at far right)
  - Need a column for the truth value of each expression that occurs in the compound proposition as it is built up.
    - This includes the atomic propositions

Example Truth Table

- Construct a truth table for

$$p \vee q \to \neg r$$

p	q	r	$\neg r$	$p \vee q$	$p \vee q \to \neg r$
T	T	T	F	T	F
T	T	F	T	T	T
T	F	T	F	T	F
T	F	F	T	T	T
F	T	T	F	T	F
F	T	F	T	T	T
F	F	T	F	F	T
F	F	F	T	F	T

Logic and bit operations

- Computer bit operations correspond to the logical connectives.
- Bitwise OR bitwise AND bitwise XOR

Bitwise Operations

- Boolean operations can be extended to operate on bit strings as well as single bits.
- E.g.:
  01 1011 0110
  <u>11 0001 1101</u>
  Bit-wise OR
  Bit-wise AND
  Bit-wise XOR

Exercise(1/3)

- 1.p and q are propositions, IFF __________, value of $P \wedge Q$ is true. IFF _________, value of $P \vee Q$ is false.
- 2.p and q are propositions, IFF __________, value of P->Q is false.
- 3. Which connective don't satisfy commutative law.(交换律)

(A) $\rightarrow$, (B) $\wedge$, (C) $\vee$, (D) $\leftrightarrow$.

Exercise(2/3)

- 4.Which is a proposition with true value?
  - I am lying.
  - If 1+2=3, then snow is black.
  - If 1+2=5, then snow is black.
  - What are you talking about?

Exercise(3/3)

- 5. If value of p,q,r,s is 1,1,0,0, calculate the value of formula:

$$(P \vee (Q \to (R \wedge \neg P))) \leftrightarrow (Q \vee \neg S)$$

- 6.Give the truth table of following formula:

$$(P \wedge Q \to R) \to P$$

answer

- 5: $(P \vee (Q \to (R \wedge \neg P))) \leftrightarrow (Q \vee \neg S)$

  $\Leftrightarrow (1 \vee (1 \to (0 \wedge \neg 1))) \leftrightarrow (1 \vee \neg 1)$

  $\Leftrightarrow 1$

- 6.

P	Q	R	P∧Q	P∧Q->R	（P∧Q->R）->P
0	0	0	0	1	0
0	0	1	0	1	0
0	1	0	0	1	0
0	1	1	0	1	0
1	0	0	0	1	1
1	0	1	0	1	1
1	1	0	1	0	1
1	1	1	1	1	1

-

§ 1.2 Applications of Propositional Logic

Applications of Propositional Logic: Summary

- Translating English to Propositional Logic
- System Specifications 系统规格
- Boolean Searching
- Logic Puzzles
- Logic Circuits

Translating sentences Example(命题翻译)

- 'If I go to school or go home, I will not go shopping.'
  - P: I go to school
  - Q: I go home
  - R: I will go shopping
- If......P......or.....Q.....then....not.....R
  - $(P \vee Q) \rightarrow \neg R$
- You can access the Internet from campus only if you are a computer science major or you are not a freshman.
- $a \rightarrow (c \vee \neg f)$

Translating English Sentences

- Steps to convert an English sentence to a statement in propositional logic
  - Identify atomic propositions and represent using propositional variables.
  - Determine appropriate logical connectives
- "You cannot ride the roller coaster if you are under 4 feet tall unless you are older than 16 years old."
  - *p*: you are under 4 feet tall,
  - q: you are older than 16 years old.
  - *r*: You can ride the roller coaster.

If *p* and ¬*q* then not *r*. $(p \land \neg q) \rightarrow \neg r$

Example

Problem: Translate the following sentence into propositional logic:

"You can access the Internet from campus only if you are a computer science major or you are not a freshman."

One Solution: Let a, c, and f represent respectively "You can access the internet from campus," "You are a computer science major," and "You are a freshman."

$$a \rightarrow (c \vee \neg f)$$

Translate

She is clever and beautiful $P \wedge Q$

Unless you work hard, or you will fail the exam.

$$\neg P \rightarrow Q$$

One and only one of A and B is student.

$$\neg (P \leftrightarrow Q)$$

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

System Specifications

- System and Software engineers take requirements in English and express them in a precise specification language based on logic.

Example: Express in propositional logic:

"The automated reply cannot be sent when the file system is full"

Solution: One possible solution: Let *p* denote "The automated reply can be sent" and *q* denote "The file system is full."

$$q \rightarrow \neg p$$

Consistent System Specifications

Definition: A list of propositions is *consistent* if it is possible to assign truth values to the proposition variables so that each proposition is true.

Exercise: Are these specifications consistent?

- "The diagnostic message is stored in the buffer or it is retransmitted."
- "The diagnostic message is not stored in the buffer."
- "If the diagnostic message is stored in the buffer, then it is retransmitted."

Solution: Let p denote "The diagnostic message is stored in the buffer." Let q denote "The diagnostic message is retransmitted" The specification can be written as: $p \lor q$, $p \to q$, $\neg p$. When p is false and q is true all three statements are true. So the specification is consistent.

- What if "The diagnostic message is not retransmitted is added."

  Solution: Now we are adding $\neg q$ and there is no satisfying assignment. So the specification is not consistent.

Logic Puzzles

- An island has two kinds of inhabitants, *knights*, who always tell the truth, and *knaves*, who always lie.
- You go to the island and meet A and B.
  - A says "B is a knight."
  - B says "The two of us are of opposite types."

Raymond Smullyan (Born 1919)

Example: What are the types of A and B?

Solution: Let p and q be the statements that A is a knight and B is a knight, respectively. So, then $\neg p$ represents the proposition that A is a knave and $\neg q$ that B is a knave.

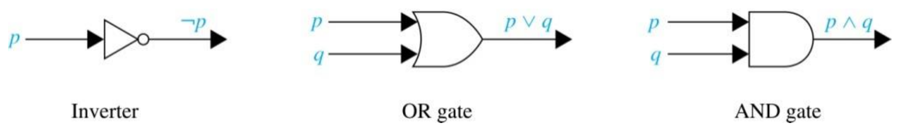
- If A is a knight, then p is true. Since knights tell the truth, q must also be true. Then $(p \land \neg q) \lor (\neg p \land q)$ would have to be true, but it is not. So, A is not a knight and therefore $\neg p$ must be true.
- If A is a knave, then B must not be a knight since knaves always lie. So, then both $\neg p$ and $\neg q$ hold since both are knaves.

Logic Puzzles

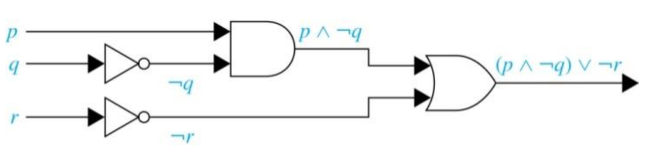
- A father tells his two children, a boy and a girl, to play in their backyard without getting dirty. However, while playing, both children get mud on their foreheads. When the children stop playing, the father says “At least one of you has a muddy forehead,” and then asks the children to answer “Yes” or “No” to the question: “Do you know whether you have a muddy forehead?” The father asks this question twice. What will the children answer each time this question is asked, assuming that a child can see whether his or her sibling has a muddy forehead, but cannot see his or her own forehead? Assume that both children are honest and that the children answer each question simultaneously.

Logic Circuits

- Electronic circuits; each input/output signal can be viewed as a 0 or 1.
  - 0 represents **False**
  - 1 represents **True**
- Complicated circuits are constructed from three basic circuits called gates.



  - The inverter (**NOT gate**)takes an input bit and produces the negation of that bit.
  - The **OR gate** takes two input bits and produces the value equivalent to the disjunction of the two bits.
  - The **AND gate** takes two input bits and produces the value equivalent to the conjunction of the two bits.
- More complicated digital circuits can be constructed by combining these basic circuits to produce the desired output given the input signals by building a circuit for each piece of the output expression and then combining them. For example:



Homework

- § 1.1 2, 14,26,30,38， 43(1,3)
- § 1.2 2,4,32,40

Some important ideas:

- Distinguishing between different kinds of formulas
- Seeing that some formulas that look different may express the same information

- First: different kinds of formulas

1.3 Propositional Equivalences (逻辑等价式)

Section Summary

- Tautologies(永真式), Contradictions(矛盾式), and Contingencies(可能式).
- Logical Equivalence
  - Important Logical Equivalences
  - Showing Logical Equivalence
- Normal Forms
  - Disjunctive Normal Form
  - Conjunctive Normal Form
- Propositional Satisfiability
  - Sudoku Example

Tautologies永真式

A *tautology* is a compound proposition that is **true** *no matter what* the truth values of its atomic propositions are!

Ex. $p \vee \neg p$ [What is its truth table?]

Tautologies

- When every row of the truth table gives T.

- Example: $p \lor \neg p$

-

p	$\neg p$	$p \lor \neg p$
T	F	T
F	T	T

Contradictions永假式

A *contradiction* is a compound proposition that is **false** no matter what! *Ex.* $p \wedge \neg p$ [Truth table?]

Contradictions

- When every row of the truth table gives F

- Example: $p \land \neg p$

p	$\land$	$\neg p$
T	F	F
F	F	T

What's left besides Tautologies and Contradictions

All other props. are *contingencies*:可能式 可满足式

Some rows give T, others give F

Now: formulas that have the same meaning

Tautologies, Contradictions, and Contingencies

- A tautology is a proposition which is always true.
  - Example: $p \vee \neg p$
- A *contradiction* is a proposition which is always false.
  - Example: $p \wedge \neg p$
- A *contingency* is a proposition which is neither a tautology nor a contradiction, such as p

P	$\neg p$	$p \vee \neg p$	$p \wedge \neg p$
T	F	T	F
F	T	T	F

Which of these are tautologies?

1. $p \rightarrow (q \rightarrow p)$
2. $p \rightarrow (\neg p \rightarrow p)$
3. $(q \rightarrow p) \rightarrow (p \rightarrow q)$
4. $(q \rightarrow p) \vee (p \rightarrow q)$
5. $(p \rightarrow (q \rightarrow r)) \rightarrow (q \rightarrow (p \rightarrow r))$

Please prove your claims, using truth tables. (Hint: Ask what assignment of truth values to p,q, and r would *falsify* each formula).

Which of these are tautologies?

1. $p \to (q \to p)$ *Tautologous*
2. $p \to (\neg p \to p)$ *Tautologous*
3. $(q \to p) \to (p \to q)$ *Contingent*
4. $(q \to p) \lor (p \to q)$ *Tautologous*
5. $(p \to (q \to r)) \to (q \to (p \to r))$ *Tautologous*

Logical Equivalence等价式

Compound proposition p is *logically equivalent* to compound proposition q, written $p \Leftrightarrow q$, **IFF** p and q contain the same truth values in all rows of their truth tables

We will also say: they express the same truth function (= the same function from values for atoms to values for the whole formula).

Logically Equivalent

- Two compound propositions p and q are logically equivalent if $p \leftrightarrow q$ is a tautology.
- We write this as $p \Leftrightarrow q$ or as $p \equiv q$ where p and q are compound propositions.
- Two compound propositions p and q are equivalent if and only if the columns in a truth table giving their truth values agree.
- This truth table show $\neg p \vee q$ is equivalent to $p \rightarrow q$.

p	q	$\neg p$	$\neg p \vee q$	$p \rightarrow q$
T	T	F	T	T
T	F	F	F	F
F	T	T	T	T
F	F	T	T	T

Propositional Equivalence

Two *syntactically* (*i.e.,* textually) different compound propositions may be *semantically* identical (*i.e.,* have the same meaning). We call them *logically equivalent*. 逻辑等价

Notation: … $\Leftrightarrow$ …

Proving Equivalence via Truth Tables

Ex. Prove that $p \vee q \Leftrightarrow \neg(\neg p \wedge \neg q)$.

p	q	$p \vee q$	$\neg p$	$\neg q$	$\neg p \wedge \neg q$	$\neg(\neg p \wedge \neg q)$
F	F	F	T	T	T	F
F	T	T	T	F	F	T
T	F	T	F	T	F	T
T	T	T	F	F	F	T

Equivalence Laws

- Similar to arithmetic identities in algebra, but for propositional equivalences instead.
- They provide a pattern or template that can be used to match all or part of a much more complicated proposition and to find an equivalence for it.
- Abbreviation: **T** stands for an arbitrary tautology; **F** an arbitrary contradiction

Logical Equivalences(等价式)

- $P \wedge T \Leftrightarrow P$ Identity（同一律）
- $P \vee F \Leftrightarrow P$
- $P \vee T \Leftrightarrow T$ Domination（零律）
- $P \wedge F \Leftrightarrow F$
- $P \vee P \Leftrightarrow P$ Idempotency（等幂律）
- $P \wedge P \Leftrightarrow P$

Logical Equivalences(等价式)

- $\neg(\neg P)) \Leftrightarrow P$ Double negation（双重否定律）
- $P \vee Q \Leftrightarrow Q \vee P$ Commutativity（交换律）
- $P \wedge Q \Leftrightarrow Q \wedge P$
- $(P \vee Q) \vee R \Leftrightarrow P \vee (Q \vee R)$ Associativity（结合律）
- $(P \wedge Q) \wedge R \Leftrightarrow P \wedge (Q \wedge R)$

Logical Equivalences

- $P \land (Q \lor R) \Leftrightarrow (P \land Q) \lor (P \land R)$ Distributivity（分配律）
- $P \lor (Q \land R) \Leftrightarrow (P \lor Q) \land (P \lor R)$
- $\neg(P \land Q) \Leftrightarrow \neg P \lor \neg Q$ DeMorgan's laws（德·摩根律）
- $\neg(P \lor Q) \Leftrightarrow \neg P \land \neg Q$

Logical Equivalences

- $P \to Q \Leftrightarrow \neg P \vee Q$ Implication（蕴涵等价式）
- $P \vee \neg P \Leftrightarrow T$ Tautology（排中律）
- $P \wedge \neg P \Leftrightarrow F$ Contradiction（矛盾律）
- $(P \to Q) \wedge (Q \to P) \Leftrightarrow (P \leftrightarrow Q)$ Equivalence
- $(P \to Q) \wedge (P \to \neg Q) \Leftrightarrow \neg P$ Absurdity 归谬论

Logical Equivalences

- $(P \rightarrow Q) \Leftrightarrow (\neg Q \rightarrow \neg P)$ Contrapositive（蕴涵逆反式）
- $P \vee (P \wedge Q) \Leftrightarrow P$, $P \wedge (P \vee Q) \Leftrightarrow P$ Absorption（吸收律）
- $(P \wedge Q) \rightarrow R \Leftrightarrow P \rightarrow (Q \rightarrow R)$ Exportation（输出律）

What's the difference between $\leftrightarrow$ and $\Leftrightarrow$?

$A \Leftrightarrow B$ says that **no assignment of truth values** to A and B can make $A \leftrightarrow B$ false

So, $A \Leftrightarrow B$ can only hold between well-chosen compound A and B. For example, $p \lor q \Leftrightarrow \neg(\neg p \land \neg q)$.

$A \leftrightarrow B$ says that A and B happen to have the same truth value. For example,
A=2+2=4. B=I teach at NWPU.
A=2+2=4. B=I teach at NWPU or MIT.

You have learned about:

- Propositional logic operators’
  - Symbolic notations.
  - English equivalents
  - Truth tables.
  - Logical equivalence

- Next:
  - More about logical equivalences.
  - How to prove them.

How to judge Logical Equivalences

- 1. Truth table
- 2、proposition formula calculate(命题公式的演算)
  - Basic Logical Equivalences(基本等值定理)
  - Substitution rule(代入规则)
  - Replacement rule(置换规则)

Substitution rule(代入规则)

- In a tautology, if we replace proposition variable R with another proposition formula, the new formula is still a tautology.

- Ex： show $(p \rightarrow q) \vee \neg (p \rightarrow q)$ is tautology
- $R \vee \neg R \Leftrightarrow T$
- replace R with $(p \rightarrow q)$

Replacement rule(置换规则)

- In a given formula A， the sub proposition are A_1， A_2， …A_n ， if $A_i \Leftrightarrow B_i$, after replace A_i with B_i, get a new formula B， then $A \Leftrightarrow B$。
- Ex： Show
- $(P \rightarrow (Q \rightarrow R)) \Leftrightarrow P \rightarrow (\neg Q \vee R)$ is a tautology
- *because* $(Q \rightarrow R) \Leftrightarrow (\neg Q \vee R)$
- Replace $(Q \rightarrow R)$ with $(\neg Q \vee R)$ in original formula.

De Morgan's Laws

Augustus De Morgan

1806-1871

$$\neg(p \wedge q) \equiv \neg p \vee \neg q$$

$$\neg(p \vee q) \equiv \neg p \wedge \neg q$$

This truth table shows that De Morgan's Second Law holds.

p	q	$\neg p$	$\neg q$	$(p \vee q)$	$\neg(p \vee q)$	$\neg p \wedge \neg q$
T	T	F	F	T	F	F
T	F	F	T	T	F	F
F	T	T	F	T	F	F
F	F	T	T	F	T	T

Key Logical Equivalences

- Identity Laws: $p \wedge T \equiv p$, $p \vee F \equiv p$
- Domination Laws: $p \vee T \equiv T$, $p \wedge F \equiv F$
- Idempotent laws: $p \vee p \equiv p$, $p \wedge p \equiv p$
- Double Negation Law: $\neg(\neg p) \equiv p$
- Negation Laws: $p \vee \neg p \equiv T$, $p \wedge \neg p \equiv F$

Key Logical Equivalences (*cont*)

- Commutative Laws: $p \lor q \equiv q \lor p$, $p \land q \equiv q \land p$
- Associative Laws:

$$(p \land q) \land r \equiv p \land (q \land r)$$
$$(p \lor q) \lor r \equiv p \lor (q \lor r)$$

- Distributive Laws:

$$(p \lor (q \land r) \equiv (p \lor q)) \land (p \lor r)$$
$$(p \land (q \lor r)) \equiv (p \land q) \lor (p \land r)$$

- Absorption Laws:

$$p \lor (p \land q) \equiv p \quad p \land (p \lor q) \equiv p$$

More Logical Equivalences

TABLE 7 Logical Equivalences Involving Conditional Statements.

$$p \to q \equiv \neg p \vee q$$
$$p \to q \equiv \neg q \to \neg p$$
$$p \vee q \equiv \neg p \to q$$
$$p \wedge q \equiv \neg(p \to \neg q)$$
$$\neg(p \to q) \equiv p \wedge \neg q$$
$$(p \to q) \wedge (p \to r) \equiv p \to (q \wedge r)$$
$$(p \to r) \wedge (q \to r) \equiv (p \vee q) \to r$$
$$(p \to q) \vee (p \to r) \equiv p \to (q \vee r)$$
$$(p \to r) \vee (q \to r) \equiv (p \wedge q) \to r$$

TABLE 8 Logical Equivalences Involving Biconditional Statements.

$$p \leftrightarrow q \equiv (p \to q) \wedge (q \to p)$$
$$p \leftrightarrow q \equiv \neg p \leftrightarrow \neg q$$
$$p \leftrightarrow q \equiv (p \wedge q) \vee (\neg p \wedge \neg q)$$
$$\neg(p \leftrightarrow q) \equiv p \leftrightarrow \neg q$$

Tautologies revisited

- We've introduced the notion of a tautology using the example $p \vee \neg p$
- Now, you know more operators, so can formulate many more tautologies, e.g.,
  $(p \vee q) \leftrightarrow \neg(\neg p \wedge \neg q)$
  $(p \rightarrow q) \leftrightarrow (\neg q \rightarrow \neg p)$, *and so on*

Constructing New Logical Equivalences

- We can show that two expressions are logically equivalent by developing a series of logically equivalent statements.
- To prove that $A \equiv B$ we produce a series of equivalences beginning with A and ending with B.

$$\begin{gathered} A \equiv A_1 \\ \vdots \\ A_n \equiv B \end{gathered}$$

- Keep in mind that whenever a proposition (represented by a propositional variable) occurs in the equivalences listed earlier, it may be replaced by an arbitrarily complex compound proposition.

Equivalence Proofs

Example: Show that $\neg(p \vee (\neg p \wedge q))$ is logically equivalent to $\neg p \wedge \neg q$

Solution:

$\neg(p \vee (\neg p \wedge q))$	$\equiv$	$\neg p \wedge \neg(\neg p \wedge q)$	by the second De Morgan law
	$\equiv$	$\neg p \wedge [\neg(\neg p) \vee \neg q]$	by the first De Morgan law
	$\equiv$	$\neg p \wedge (p \vee \neg q)$	by the double negation law
	$\equiv$	$(\neg p \wedge p) \vee (\neg p \wedge \neg q)$	by the second distributive law
	$\equiv$	$F \vee (\neg p \wedge \neg q)$	because $\neg p \wedge p \equiv F$
	$\equiv$	$(\neg p \wedge \neg q) \vee F$	by the commutative law for disjunction
	$\equiv$	$(\neg p \wedge \neg q)$	by the identity law for **F**

Equivalence Proofs

Example: Show that $(p \wedge q) \rightarrow (p \vee q)$ is a tautology.

Solution:

$$
\begin{aligned}
(p \wedge q) \rightarrow (p \vee q) &\equiv \neg(p \wedge q) \vee (p \vee q) && \text{by truth table for } \rightarrow \\
&\equiv (\neg p \vee \neg q) \vee (p \vee q) && \text{by the first De Morgan law} \\
&\equiv (\neg p \vee p) \vee (\neg p \vee \neg q) && \text{by associative and} \\
& && \text{commutative laws} \\
& && \text{laws for disjunction} \\
&\equiv T \vee T && \text{by truth tables} \\
&\equiv T && \text{by the domination law}
\end{aligned}
$$

Logical equivalence application

- 1.Judge if two formula are equivalence
- 判定是否逻辑等价
- 2.Judge tautology , Contradictions
- 判断是否为永真公式或永假公式
- 3. Simplify proposition formula
- 命题公式的化简

EXAMPLE 4

- Show that the propositions $p\vee(q\wedge r)$ and $(p\vee q)\wedge(p\vee r)$ are logically equivalent. This is the distributive law of disjunction over conjunction.

Solution:

We construct the truth table for these propositions in Table 4. Since the truth values of $p\vee(q\wedge r)$ and $(p\vee q)\wedge(p\vee r)$ agree, these propositions are logically equivalent.

Table 4

TABLE 4

A Demonstration That $p \vee (q \vee r)$ and $(p \vee q) \wedge (p \vee r)$ Are Logically Equivalent.

p	q	r	$q \wedge r$	$p \vee (q \wedge r)$	$p \vee q$	$p \vee r$	$(p \vee q) \wedge (p \vee r)$
T	T	T	T	T	T	T	T
T	T	F	F	T	T	T	T
T	F	T	F	T	T	T	T
T	F	F	F	T	T	T	T
F	T	T	T	T	T	T	T
F	T	F	F	F	T	F	F
F	F	T	F	F	F	T	F
F	F	F	F	F	F	F	F

EXAMPLE 5

- Show that $\neg(p \vee (\neg p \wedge q))$ and $\neg p \wedge \neg q$ are logically equivalent.
-

$$
\begin{aligned}
\neg(p \vee (\neg p \wedge q)) &\Leftrightarrow \neg p \wedge \neg(\neg p \wedge q) \\
&\Leftrightarrow \neg p \wedge [\neg(\neg p) \vee \neg q] \\
&\Leftrightarrow \neg p \wedge (p \vee \neg q) \\
&\Leftrightarrow (\neg p \wedge p) \vee (\neg p \wedge \neg q) \\
&\Leftrightarrow F \vee (\neg p \wedge \neg q) \\
&\Leftrightarrow (\neg p \wedge \neg q) \vee F \\
&\Leftrightarrow \neg p \wedge \neg q
\end{aligned}
$$

EXAMPLE 6

- Show that $(p \wedge q) \rightarrow (p \vee q)$ is a tautology.

$$
\begin{aligned}
&(p \wedge q) \rightarrow (p \vee q) \\
&\quad \Leftrightarrow \neg (p \wedge q) \vee (P \vee q) \\
&\quad \Leftrightarrow (\neg p \vee \neg q) \vee (p \vee q) \\
&\quad \Leftrightarrow (\neg p \vee p) \vee (\neg q \vee q) \\
&\quad \Leftrightarrow T \vee T \\
&\quad \Leftrightarrow T
\end{aligned}
$$

An Example Problem

- Check using a symbolic derivation whether $(p \wedge \neg q) \rightarrow (p \oplus r) \Leftrightarrow \neg p \vee q \vee \neg r$.

$(p \wedge \neg q) \rightarrow (p \oplus r)$ [Expand definition of $\rightarrow$]

$\Leftrightarrow \underline{\neg}(p \wedge \neg q)\ \underline{\vee}\ (p \oplus r)$ [Expand defn. of $\oplus$]

$\Leftrightarrow \neg(p \wedge \neg q) \vee (\underline{(p \vee r) \wedge \neg(p \wedge r)})$

[DeMorgan's Law]

$\Leftrightarrow \underline{(\neg p \vee q)} \vee ((p \vee r) \wedge \neg(p \wedge r))$

cont.

Example Continued...

$(\neg p \lor q) \lor ((p \lor r) \land \neg(p \land r)) \Leftrightarrow$ [$\lor$ commutes]

$\Leftrightarrow (\underline{q \lor \neg p}) \lor ((p \lor r) \land \neg(p \land r))$ [$\lor$ is associative]

$\Leftrightarrow q \lor \underline{(}\neg p \lor ((p \lor r) \land \neg(p \land r)))$ [distrib. $\lor$ over $\land$]

$\Leftrightarrow q \lor (((\neg p \lor (p \lor r)) \land (\underline{\neg p \lor} \neg(p \land r)))$

[assoc.] $\Leftrightarrow q \lor (((\neg p \lor p) \lor r) \land (\neg p \lor \neg(p \land r)))$

[trivial taut.] $\Leftrightarrow q \lor ((\mathbf{T} \lor r) \land (\neg p \lor \neg(p \land r)))$

[domination] $\Leftrightarrow q \lor (\mathbf{T} \land (\neg p \lor \neg(p \land r)))$

[identity] $\Leftrightarrow q \lor (\neg p \lor \neg(p \land r)) \Leftrightarrow$ *cont.*

End of Long Example

$q \vee (\neg p \vee \neg(p \wedge r))$

[DeMorgan's] $\Leftrightarrow q \vee (\neg p \vee (\neg p \vee \neg r))$

[Assoc.] $\Leftrightarrow q \vee ((\neg p \vee \neg p) \vee \neg r)$

[Idempotent] $\Leftrightarrow q \vee (\neg p \vee \neg r)$

[Assoc.] $\Leftrightarrow (q \vee \neg p) \vee \neg r$

[Commut.] $\Leftrightarrow \neg p \vee q \vee \neg r$

Q.E.D. (quod erat demonstrandum)

(Which was to be shown.)

Exercise

- Show $Q \vee \neg((\neg P \vee Q) \wedge P)$ is a tautology

Equivalence Laws - Examples

- *Identity*: $p\wedge\mathbf{T} \Leftrightarrow p$ $\quad p\vee\mathbf{F} \Leftrightarrow p$
- *Domination*: $p\vee\mathbf{T} \Leftrightarrow \mathbf{T}$ $\quad p\wedge\mathbf{F} \Leftrightarrow \mathbf{F}$
- *Idempotence*: $p\vee p \Leftrightarrow p$ $\quad p\wedge p \Leftrightarrow p$
- *Double negation:* $\neg\neg p \Leftrightarrow p$
- *Commutativity:* $p\vee q \Leftrightarrow q\vee p$ $\quad p\wedge q \Leftrightarrow q\wedge p$
- *Associativity:* $(p\vee q)\vee r \Leftrightarrow p\vee(q\vee r)$
  $(p\wedge q)\wedge r \Leftrightarrow p\wedge(q\wedge r)$

More Equivalence Laws

- *Distributive*: $p \vee (q \wedge r) \Leftrightarrow (p \vee q) \wedge (p \vee r)$
  $p \wedge (q \vee r) \Leftrightarrow (p \wedge q) \vee (p \wedge r)$
- *De Morgan's*:
  $\neg(p \wedge q) \Leftrightarrow \neg p \vee \neg q$
  $\neg(p \vee q) \Leftrightarrow \neg p \wedge \neg q$
- *Trivial tautology/contradiction*:
  $p \vee \neg p \Leftrightarrow \mathbf{T}$ $\quad$ $p \wedge \neg p \Leftrightarrow \mathbf{F}$

Augustus De Morgan (1806-1871)

Defining Operators via Equivalences

Using equivalences, we can *define* operators in terms of other operators.

- Exclusive or: $p \oplus q \Leftrightarrow (p \vee q) \wedge \neg(p \wedge q)$

  $p \oplus q \Leftrightarrow (p \wedge \neg q) \vee (q \wedge \neg p)$
- Implies: $p \rightarrow q \Leftrightarrow \neg p \vee q$
- Biconditional: $p \leftrightarrow q \Leftrightarrow (p \rightarrow q) \wedge (q \rightarrow p)$

  $p \leftrightarrow q \Leftrightarrow \neg(p \oplus q)$

Normal or *Canonical* Forms(范式）

- Unique representations of a proposition
- Examples:
- Construct a simple proposition of two variables which is true only when
  - P is true and Q is false:

    $P \wedge \neg Q$
  - P is true and Q is true:

    $P \wedge Q$
  - P is true and Q is false or P is true and Q is true:

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

Normal or *Canonical* Forms

- A disjunction of conjunctions where
  - every variable or its negation is represented once in each conjunction （合取式） (a *minterm*)
  - each minterms appears only once
- A conjunction of disjunctions where
  - every variable or its negation is represented once in each conjunction （析取式） (a *maxterm*)
  - each maxterms appears only once

Disjunctive Normal Form

- A propositional formula(析取范式) is in *disjunctive normal form* if it consists of a disjunction of (1, … ,*n*) disjuncts where each disjunct consists of a conjunction of (1, …, *m*) atomic formulas or the negation of an atomic formula.
  - Yes $(p \wedge \neg q) \vee (\neg p \vee q)$
  - No $p \wedge (p \vee q)$

Disjunctive Normal Form

Example: Show that every compound proposition can be put in disjunctive normal form.

Solution: Construct the truth table for the proposition. Then an equivalent proposition is the disjunction with *n* disjuncts (where *n* is the number of rows for which the formula evaluates to **T)**. Each disjunct has m conjuncts where *m* is the number of distinct propositional variables. Each conjunct includes the positive form of the propositional variable if the variable is assigned **T** in that row and the negated form if the variable is assigned **F** in that row. This proposition is in disjunctive normal from.

Disjunctive Normal Form (DNF)（析取范式）

- Important in switching theory, simplification in the design of circuits.
- Method: To find the minterms of the DNF.
  - Use the rows of the truth table where the proposition is 1 or True
  - If a zero appears under a variable, use the negation of the propositional variable in the minterm
  - If a one appears, use the propositional variable.

Disjunctive Normal Form

Example: Find the Disjunctive Normal Form (DNF) of

$$(p \vee q) \to \neg r$$

Solution: This proposition is true when *r* is false or when both *p* and *q* are false.

$$(\neg p \wedge \neg q) \vee \neg r$$

Example

- Find the DNF of $(P \vee Q) \rightarrow \neg R$

P	Q	R	$(P \vee Q) \rightarrow \neg R$
0	0	0	1
0	0	1	1
0	1	0	1
0	1	1	0
1	0	0	1
1	0	1	0
1	1	0	1
1	1	1	0

Example

- There are 5 cases where the proposition is true, hence 5 minterms. Rows 1,2,3, 5 and 7 produce the following disjunction of minterms:

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

$\Leftrightarrow (\neg P \wedge \neg Q \wedge \neg R) \vee (\neg P \wedge \neg Q \wedge R) \vee (\neg P \wedge Q \wedge \neg R) \vee (P \wedge \neg Q \wedge \neg R) \vee (P \wedge Q \wedge \neg R)$

  - Note that you get a *Conjunctive Normal Form* (CNF)（合取范式） if you negate a DNF and use DeMorgan’s Laws.

minterm

- $(\neg P \wedge \neg Q \wedge \neg R) \vee (\neg P \wedge \neg Q \wedge R) \vee (\neg P \wedge Q \wedge \neg R) \vee (P \wedge \neg Q \wedge \neg R) \vee (P \wedge Q \wedge \neg R)$
- $\Leftrightarrow m_0 \vee m_1 \vee m_2 \vee m_4 \vee m_6$

minterm	values	oct	denote
$\neg P \wedge \neg Q \wedge \neg R$	000	0	m_{000}或m_0
$\neg P \wedge \neg Q \wedge R$	001	1	m_{001}或m_1
$\neg P \wedge Q \wedge \neg R$	010	2	m_{010}或m_2
$\neg P \wedge Q \wedge R$	011	3	m_{011}或m_3
$P \wedge \neg Q \wedge \neg R$	100	4	m_{100}或m_4
$P \wedge \neg Q \wedge R$	101	5	m_{101}或m_5
$P \wedge Q \wedge \neg R$	110	6	m_{110}或m_6
$P \wedge Q \wedge R$	111	7	m_{111}或m_7

Conjunctive Normal Form

- A compound proposition is in *Conjunctive Normal Form* (CNF,合取范式) if it is a conjunction of disjunctions.
- Every proposition can be put in an equivalent CNF.
- Conjunctive Normal Form (CNF) can be obtained by eliminating implications, moving negation inwards and using the distributive and associative laws.
- Important in resolution theorem proving used in artificial Intelligence (AI).
- A compound proposition can be put in conjunctive normal form through repeated application of the logical equivalences covered earlier.

Conjunctive Normal Form

Example: Put the following into CNF:

Solution: $\neg(p \to q) \vee (r \to p)$

1. Eliminate implication signs:

$$\neg(\neg p \vee q) \vee (\neg r \vee p)$$

2. Move negation inwards; eliminate double negation:

$$(p \wedge \neg q) \vee (\neg r \vee p)$$

3. Convert to CNF using associative/distributive laws

$$(p \vee \neg r \vee p) \wedge (\neg q \vee \neg r \vee p)$$

Propositional Satisfiability

- A compound proposition is *satisfiable* if there is an assignment of truth values to its variables that make it true. When no such assignments exist, the compound proposition is *unsatisfiable*.
- A compound proposition is unsatisfiable if and only if its negation is a tautology.

Questions on Propositional Satisfiability

Example: Determine the satisfiability of the following compound propositions:

$$(p \vee \neg q) \wedge (q \vee \neg r) \wedge (r \vee \neg p)$$

Solution: Satisfiable. Assign **T** to $p, q,$ and r.

$$(p \vee q \vee r) \wedge (\neg p \vee \neg q \vee \neg r)$$

Solution: Satisfiable. Assign **T** to p and $\boldsymbol{F}$ to q.

$$(p \vee \neg q) \wedge (q \vee \neg r) \wedge (r \vee \neg p) \wedge (p \vee q \vee r) \wedge (\neg p \vee \neg q \vee \neg r)$$

Solution: Not satisfiable. Check each possible assignment of truth values to the propositional variables and none will make the proposition true.

maxterm

- $(\neg P \wedge \neg Q \wedge \neg R) \vee (\neg P \wedge \neg Q \wedge R) \vee (\neg P \wedge Q \wedge \neg R) \vee (P \wedge \neg Q \wedge \neg R) \vee (P \wedge Q \wedge \neg R)$
- $\Leftrightarrow m_0 \vee m_1 \vee m_2 \vee m_4 \vee m_6$
- $\Leftrightarrow M_3 \wedge M_5 \wedge M_7$

maxterm	value	oct	denote
$P \vee Q \vee R$	000	0	M_{000}或M_0
$P \vee Q \vee \neg R$	001	1	M_{001}或M_1
$P \vee \neg Q \vee R$	010	2	M_{010}或M_2
$P \vee \neg Q \vee \neg R$	011	3	M_{011}或M_3
$\neg P \vee Q \vee R$	100	4	M_{100}或M_4
$\neg P \vee Q \vee \neg R$	101	5	M_{101}或M_5
$\neg P \vee \neg Q \vee R$	110	6	M_{110}或M_6
$\neg P \vee \neg Q \vee \neg R$	111	7	M_{111}或M_7

Notation

$$\bigvee_{j=1}^{n} p_j \text{ is used for } p_1 \vee p_2 \vee \ldots \vee p_n$$

$$\bigwedge_{j=1}^{n} p_j \text{ is used for } p_1 \wedge p_2 \wedge \ldots \wedge p_n$$

Needed for the next example.

Sudoku

- A **Sudoku puzzle** is represented by a 9×9 grid made up of nine 3×3 subgrids, known as **blocks**. Some of the 81 cells of the puzzle are assigned one of the numbers 1,2, …, 9.
- The puzzle is solved by assigning numbers to each blank cell so that every row, column and block contains each of the nine possible numbers.
- Example

	2	9				4		
			5			1		
	4							
				4	2			
6							7	
5								
7			3					5
	1			9				
							6	

Encoding as a Satisfiability Problem

- Let $p(i,j,n)$ denote the proposition that is true when the number n is in the cell in the ith row and the jth column.
- There are $9 \times 9 \times 9 = 729$ such propositions.
- In the sample puzzle $p(5,1,6)$ is true, but $p(5,j,6)$ is false for $j = 2,3,\ldots 9$

	2	9				4		
			5			1		
	4							
				4	2			
6							7	
5								
7			3					5
	1			9				
							6	

Encoding (cont)

- For each cell with a given value, assert $p(i,j,n)$, when the cell in row i and column j has the given value.
- Assert that every row contains every number.

$$\bigwedge_{i=1}^{9} \bigwedge_{n=1}^{9} \bigvee_{j=1}^{9} p(i,j,n)$$

- Assert that every column contains every number.

$$\bigwedge_{j=1}^{9} \bigwedge_{n=1}^{9} \bigvee_{i=1}^{9} p(i,j,n)$$

Encoding (cont)

- Assert that each of the 3 x 3 blocks contain every number.

$$\bigwedge_{r=0}^{2}\bigwedge_{s=0}^{2}\bigwedge_{n=1}^{9}\bigwedge_{i=1}^{3}\bigvee_{j=1}^{3} p(3r+i,3s+j,n)$$

(this is tricky - ideas from chapter 4 help)

- Assert that no cell contains more than one number. Take the conjunction over all values of *n*, *n'*, *i*, and j, where each variable ranges from 1 to 9 and $n \neq n'$, of $p(i,j,n) \rightarrow \neg p(i,j,n')$

Solving Satisfiability Problems

- To solve a Sudoku puzzle, we need to find an assignment of truth values to the 729 variables of the form *p(i,j,n)* that makes the conjunction of the assertions true. Those variables that are assigned T yield a solution to the puzzle.
- A truth table can always be used to determine the satisfiability of a compound proposition. But this is too complex even for modern computers for large problems.
- There has been much work on developing efficient methods for solving satisfiability problems as many practical problems can be translated into satisfiability problems.

Homework

- § 1.3 – 8, 28, 30, 36, 40, 48, 54,58
- Write the disjunction normal formula and conjunction normal formula of following:（求下列公式的析取范式和合取范式)
  - $((\neg A \vee \neg B) \rightarrow (A \leftrightarrow C)) \rightarrow B$
- Judge if the formula are logical equivalence by compare their CNF. (通过求主合取范式的方法，判断下列各对公式是否等价)
  - $\neg A \vee (A \wedge B) \vee C$
  - $(\neg A \vee B) \wedge (B \vee C)$
- Get all assigns make following formula true by DNF. (通过求主析取范式，求出所有使下列公式为真的赋值)
  - $(A \vee B) \wedge (A \rightarrow C) \wedge (B \rightarrow C)$

Predicate Logic (§1.4) 谓词逻辑

- Review of proposition logic
- Predicate logic

Propositional logic: review

- **Propositional logic:** a formal language for making logical inferences
- **A proposition is** a statement that is either true or false.
- **A compound proposition** can be created from other propositions using logical connectives
- **The truth of a compound proposition** is defined by truth values of elementary propositions and the meaning of connectives.
- **The truth table for a compound proposition:** table with entries (rows) for all possible combinations of truth values of elementary propositions.

Limitations of the propositional logic

- **Propositional logic:** the world is described in terms of elementary propositions and their logical combinations

Elementary statements:

- **Typically refer to objects**, **their properties and relations.**
  But these are not explicitly represented in the propositional logic
  - **Example:**
  - • "Willy is a NWPU student."

    Willy → a NWPU student

    Object   a property
  - Objects and properties are hidden in the statement, it is not possible to reason about them

Limitations of the propositional logic

(1) Statements that must be repeated for many objects

– In propositional logic these must be exhaustively enumerated

- **Example:**

– If Willy is a CS NWPU graduate then Willy has passed discrete math.

Translation:

– Willy is a CS NWPU graduate –> Willy has passed math.

Similar statements can be written for other graduates:

– David is a CS NWPU graduate –> David has passed math

– Ken is a CS NWPU graduate –> Ken has passed math

– **…**

- **What is a more natural solution to express the above knowledge**

Solution: Make statements with variables

--If x is a CS NWPU graduate then x has passed math

-- x is a CS NWPU graduate –> x has passed math

Limitations of the propositional logic

(2) Statements that define the property of the group of objects

- **Example:**
  - All new cars must be registered.
  - Some of the CS graduates graduate with honors.
- **Solution:** make statements with **quantifiers**
  - **Universal quantifier** –the property is satisfied by all members of the group
  - **Existential quantifier** – at least one member of the group satisfy the property

Predicate logic

Remedies the limitations of the propositional logic

- Explicitly models objects and their properties
- Allows to make statements with variables and quantify them

Basic building blocks of the predicate logic:

- **Constant** –models a specific object

  Examples: “John”, “France”, “7”

- **Variable** – represents object of specific type **(defined by the *universe of discourse*)**

  Examples: x, y (universe of discourse can be people, students, numbers)

- **Predicate - over one, two or many variables or constants. – Represents properties or relations among objects**

  Examples: Red(car23), student(x), married(John,Ann)

Predicates

Predicates represent properties or relations among objects

A predicate P(x) assigns a value **true or false** to each x depending on whether the property holds or not for x.

- The assignment is best viewed as a big table with the variable x substituted for objects from ***the universe of discourse***

Example:

- **Assume Student(x) where the universe of discourse are people**
- **Student(John) …. T (if John is a student)**
- **Student(Ann) …. T (if Ann is a student)**
- **Student(Jane) ….. F (if Jane is not a student)**
- **…**

Predicates

Assume a predicate P(x) that represents the statement:

- **x is a prime number**

What are the truth values of:

- P(2)
- P(3)
- P(4)
- P(5)
- P(6)
- P(7)

All statements P(2), P(3), P(4), P(5), P(6), P(7) are propositions

Is P(x) a proposition? No. Many possible substitutions are possible.

Predicates

Predicates can have **more arguments** which represent the **relations between objects**

Example:

- Older(John, Peter) denotes ‘John is older than Peter’ – this is a proposition because it is either true or false
- Older(x,y) - ‘x is older than y’ – not a proposition, but after the substitution it becomes one

Predicates

Predicates can have **more arguments** which represent the **relations between objects**

Example:

- Let Q(x,y) denote ‘x+5 >y’

– Is Q(x,y) a proposition? **No!**

– Is Q(3,7) a proposition? **Yes.** It is true.

– What is the truth value of:

– Q(3,7) T

– Q(1,6) F

– Q(2,2) T

– Is Q(3,y) a proposition? **No!** We cannot say if it is true or false.

Example

- Let P(x) denote the statement "x > 3." What are the truth values of P(4) and P(2)?
- Let Q(x, y) denote the statement "x = y + 3." What are the truth values of the propositions Q(1, 2) and Q(3, 0)?
- R(x,y,z): x+y=z, What are the truth values of the propositions R(1, 2, 3) and R(0, 0, 1)?

P (4) = .T.	Q(1, 2)= .F.	R(1, 2, 3)=.T.
P (2) = .F.	Q(3, 0)= .T.	R(0, 0, 1)=.F.

Compound statements in predicate logic

Compound statements are obtained via logical connectives

Examples:

Student(Ann) $\wedge$ Student(Jane)

- **Translation:** "Both Ann and Jane are students"
- **Proposition:** yes.

Country(Sienna) $\vee$ River(Sienna)

- **Translation:** "Sienna is a country or a river"
- **Proposition:** yes.

CS-major(x) -> Student(x)

- **Translation:** "if x is a CS-major then x is a student"
- **Proposition:** no.

Predicates

Important:

• statement P(x) is **not a proposition** since there are more objects it can be applied to

This is the same as in propositional logic …

… But the difference is:

• predicate logic allows us to explicitly manipulate and substitute for the objects

• Predicate logic permits quantified sentences where variables are substituted for statements about the group of objects

Quantified statements

Predicate logic lets us to make statements about groups of objects

- To do this we use special quantified expressions

Two types of quantified statements:

- **universal**

Example: ‘ all CS NWPU graduates have to pass discrete math”

– the statement is true for all graduates

- **existential**

Example: ‘Some CS NWPU students graduate with honor.’

– the statement is true for some people

Universal quantifier

Define: The universal quantification of P(x) is the proposition: "*P(x) is true for all values of x in the domain of discourse*." The notation **$\forall$x P(x)** denotes the universal quantification of P(x), and is expressed as **for every x, P(x).**

Example:

- Let P(x) denote x > x - 1.
- What is the truth value of $\forall$x P(x)?
- Assume the universe of discourse of x is all real numbers.
- **Answer:** Since every number x is greater than itself minus 1.

Therefore, **$\forall$x P(x) is true.**

Universal quantifier

Quantification converts a propositional function into **a proposition** by binding a variable to a set of values from the universe of discourse.

Example:

- Let P(x) denote $x > x - 1$.
- Is P(x) a proposition? **No.** Many possible substitutions.
- Is $\forall x$ P(x) a proposition? **Yes.** True if for all x from the universe of discourse P(x) is true.

Universally quantified statements

Predicate logic lets us make statements about groups of objects

Universally quantified statement

- CS-major(x) ->Student(x)

– **Translation:** “if x is a CS-major then x is a student”

– **Proposition: no.**

- $\forall$x CS-major(x) -> Student(x)

– **Translation:** “(For all people it holds that) if a person is a CS-major then she is a student.”

– **Proposition: yes.**

Example (Diff set)

- Express the statement "Every student in this class has studied calculus“ as a universal quantification.

> It can be written as
>
> $\forall xP(x)$ or $\forall x\,(S(x)\rightarrow P(x))$
>
> $P(x)$ = "x has studied calculus."
>
> $S(x)$ = "x is in this class."

- What is the truth value of $\forall x\, P(x)$, where $P(x)$ is the statement "$x^2 < 10$" and the universe of discourse consists of the positive integers not exceeding 4?

> $\forall x\, P(x)=P(1) \wedge P(2) \wedge P(3) \wedge P(4) =$ **.F.**

Existential quantifier

Definition: The **existential quantification** of P(x) is the proposition "*There exists an element in the domain (universe) of discourse such that P(x) is true*." The notation **∃x P(x)** denotes the existential quantification of P(x), and is expressed as **there is an x such that P(x) is true**.

Example 1:

- Let T(x) denote x > 5 and x is from Real numbers.
- What is the truth value of ∃ x T(x)?
- **Answer:**
- Since 10 > 5 is true. Therefore, it is **true that ∃ x T(x).**

Existential quantifier

Definition: The **existential quantification** of P(x) is the proposition "*There exists an element in the domain (universe) of discourse such that P(x) is true*." The notation ∃ **x P(x)** denotes the existential quantification of P(x), and is expressed as **there is an x such that P(x) is true**.

Example 2:

- Let Q(x) denote x = x + 2 where x is real numbers
- What is the truth value of ∃ x Q(x)?
- **Answer:** Since no real number is 2 larger than itself, the truth value of ∃ **x Q(x) is false**.

Existentially quantified statements

Statements about groups of objects

Example:

- CS-NWPU-graduate (x) $\wedge$ Honor-student(x)

– **Translation:** "x is a CS-NWPU-graduate and x is an honor student"

– **Proposition:** ?

- $\exists$ x CS-Upitt-graduate (x) $\wedge$ Honor-student(x)

– **Translation:** "There is a person who is a CS-Upitt-graduate and who is also an honor student."

– **Proposition:** ?

Example

- Let P(x) denote the statement "x > 3." What is the truth value of the quantification ∃ x P(x), where the universe of discourse is the set of real numbers?

Since "x>3" is true, for instance, when x=4 the existential quantification of P(x), which is ∃ xP(x) is true.

- Let Q(x) denote the statement "x = x + 1." What is the truth value of the quantification ∃ xQ(x), where the universe of discourse is the set of real numbers?

. F.

Example

- What is the truth value of $\exists x P(x)$ where $P(x)$ is the statement "$x^2 > 10$" and the universe of discourse consists of the positive integers not exceeding 4?

$$\exists x P(x)$$
$$= P(1) \vee P(2) \vee P(3) \vee P(4)$$
$$= .T.$$

Summary of quantified statements

- When $\forall$**x P(x)** and $\exists$**x P(x)** are true and false?

Statement	When true?	When false?
$\forall x\, P(x)$	P(x) true for all x	There is an x where P(x) is false.
$\exists x\, P(x)$	There is some x for which P(x) is true.	P(x) is false for all x.

- Suppose the elements in the universe of discourse can be enumerated as x1, x2, ..., xN then:
- • $\forall$**x P(x) is true whenever P(x1) $\wedge$ P(x2) $\wedge$ ... $\wedge$ P(xN) is true**
- • $\exists$**x P(x) is true whenever P(x1) $\vee$ P(x2) $\vee$ ... $\vee$ P(xN) is true**.

More to Know About Binding

- $(\forall x\, P(x)) \wedge Q(x)$ - The variable *x* is outside of the *scope* of the $\forall x$ quantifier, and is therefore free. Not a complete proposition!

- $(\forall x\, P(x)) \wedge (\exists x\, Q(x))$ – A complete proposition, and no superfluous quantifiers

Interpret quantified propositions

- Interpret quantified propositions(谓词公式的解释)
  - Set 个体域
  - Quantifier 谓词
  - Function 函数
  - Constant 个体常项

Example 23

- Express the statement “ Every student in this class has studied calculus” using predicates and quantifiers.

- $\forall x\, (S(x) \rightarrow C(x))$
- Difference:
- $\forall x\, (S(x) \lor E(x))$

EXAMPLE 24

- Express the statements "Some student in this class has visited Mexico" and "Every student in this class has visited either Canada or Mexico" using quantifiers.

- $\exists x\,(S(x) \wedge M(x))$
- Difference:
- $\exists x\,(S(x) \rightarrow M(x))$

$$\boxed{\forall\, x(C(x) \vee M(x))}$$

$$\forall x\,(S(x) \rightarrow (C(x) \vee M(x))$$

Example 25

- Use predicates and quantifiers to express the system specifications “Every mail message larger than one megabyte will be compressed” and “If a user is active, at least one network link will be available.”

- Let S(m,y) be “Mail message *m* is larger than y megabytes” and let C(m) denote “Mail message *m* will be compressed.”
- $\forall m\ (S(m,1) \rightarrow C(m))$
- A(u):User *u* is active   S(n,x): network link *n* is in state *x*
- $\exists u A(u) \rightarrow \exists n S(n, \text{available})$

1.5 Nested Quantifiers

- Section Summary
  - Nested Quantifiers
  - Order of Quantifiers
  - Translating from Nested Quantifiers into English
  - Translating Mathematical Statements into Statements involving Nested Quantifiers.
  - Translated English Sentences into Logical Expressions.
  - Negating Nested Quantifiers.

Nested Quantifiers

- Nested quantifiers are often necessary to express the meaning of sentences in English as well as important concepts in computer science and mathematics.
- Example: "Every real number has an inverse" is
- $\forall x \exists y(x + y = 0)$
- where the domains of x and y are the real numbers.
- We can also think of nested propositional functions:
  - $\forall x \exists y(x + y = 0)$ can be viewed as $\forall x\, Q(x)$ where $Q(x)$ is $\exists y\, P(x, y)$ where $P(x, y)$ is $(x + y = 0)$

Thinking of Nested Quantification

- Nested Loops
  - To see if $\forall x \forall y P(x,y)$ is true, loop through the values of x :
    - At each step, loop through the values for y.
    - If for some pair of x andy, P(x,y) is false, then $\forall x \forall y P(x,y)$ is false and both the outer and inner loop terminate.
  - $\forall x \forall y P(x,y)$ is true if the outer loop ends after stepping through each x.
  - To see if $\forall x \exists y P(x,y)$ is true, loop through the values of x:
    - At each step, loop through the values for y.
    - The inner loop ends when a pair x and y is found such that P(x, y) is true.
    - If no y is found such that P(x, y) is true the outer loop terminates as $\forall x \exists y P(x,y)$ has been shown to be false.
  - $\forall x \exists y P(x,y)$ is true if the outer loop ends after stepping through each x.
- If the domains of the variables are infinite, then this process can not actually be carried out.

Order of Quantifiers

- Examples:
- Let P(x,y) be the statement "x + y = y + x." Assume that U is the real numbers. Then $\forall x\, \forall y P(x,y)$ and $\forall y\, \forall x P(x,y)$ have the same truth value.
- Let Q(x,y) be the statement "x + y = 0." Assume that U is the real numbers. Then $\forall x\, \exists y P(x,y)$ is true, but $\exists y\, \forall x P(x,y)$ is false.

Questions on Order of Quantifiers

- Example 1: Let U be the real numbers,
- Define $P(x,y) : x \cdot y = 0$
- What is the truth value of the following:
  - $\forall x \forall y P(x,y)$
    - Answer: False
  - $\forall x \exists y P(x,y)$
    - Answer: True
  - $\exists x \forall y\ P(x,y)$
    - Answer: True
  - $\exists x\ \exists\ y\ P(x,y)$
  - Answer: True

Questions on Order of Quantifiers

- Example 2: Let U be the real numbers,
- Define P(x,y) : x / y = 1
- What is the truth value of the following:
  - $\forall x \forall y P(x,y)$
    - Answer: False
  - $\forall x \exists y P(x,y)$
    - Answer: False(when x is 0)
  - $\exists x \forall y\ P(x,y)$
    - Answer: False
  - $\exists x\ \exists\ y\ P(x,y)$
    - Answer: True

Quantifications of Two Variables

Statement	When True?	When False
$\forall x \forall y P(x, y)$ $\forall y \forall x P(x, y)$	*P*(*x*,*y*) is true for every pair *x*,*y*.	There is a pair *x, y* for which *P*(*x*,*y*) is false.
$\forall x \exists y P(x, y)$	For every *x* there is a *y* for which *P*(*x*,*y*) is true.	There is an x such that *P*(*x*,*y*) is false for every *y*.
$\exists x \forall y P(x, y)$	There is an *x* for which *P*(*x*,*y*) is true for every *y*.	For every *x* there is a y for which *P*(x,y) is false.
$\exists x \exists y P(x, y)$ $\exists y \exists x P(x, y)$	There is a pair *x, y* for which *P*(*x*,*y*) is true.	*P*(x,y) is false for every pair *x*,*y*

Translating Nested Quantifiers into English

- Example 1: Translate the statement
- $\forall x\ (C(x) \vee \exists y\ (C(y) \wedge F(x, y)))$
- where C(x) is “x has a computer,” and F(x,y) is “x and y are friends,” and the domain for both x and y consists of all students in your school.
- Solution: Every student in your school has a computer or has a friend who has a computer.
- Example 1: Translate the statement

- $\exists x \forall y\ \forall z\ ((F(x, y) \wedge F(x,z) \wedge (y \neq z)) \rightarrow \neg F(y,z))$
- Solution: Every student none of whose friends are also friends with each other.

Translating Mathematical Statements into Predicate Logic

- Example : Translate "The sum of two positive integers is always positive" into a logical expression.
- Solution:
  - Rewrite the statement to make the implied quantifiers and domains explicit:
    - "For every two integers, if these integers are both positive, then the sum of these integers is positive."
  - Introduce the variables x and y, and specify the domain, to obtain:
    - "For all positive integers x and y, x + y is positive."
  - The result is:
    - $\forall x \, \forall y \, ((x > 0) \wedge (y > 0) \rightarrow (x + y > 0))$
    - where the domain of both variables consists of all integers

Translating English into Logical Expressions Example

- Example: Use quantifiers to express the statement "There is a woman who has taken a flight on every airline in the world."
- Solution:
  - Let $P(w,f)$ be "w has taken f" and $Q(f,a)$ be "f is a flight on a."
  - The domain of w is all women, the domain of f is all flights, and the domain of a is all airlines.
  - Then the statement can be expressed as:
  - $\exists w\, \forall a\, \exists f\, (P(w,f) \wedge Q(f,a))$

Questions on Translation from English

- Choose the obvious predicates and express in predicate logic.
- Example 1: “Brothers are siblings.”
- Solution: $\forall x \, \forall y \, (B(x,y) \rightarrow S(x,y))$
- Example 2: “Siblinghood is symmetric.”
- Solution: $\forall x \, \forall y \, (S(x,y) \rightarrow S(y,x))$
- Example 3: “Everybody loves somebody.”
- Solution: $\forall x \, \exists y \, L(x,y)$
- Example 4: “There is someone who is loved by everyone.”
- Solution: $\exists y \, \forall x \, L(x,y)$
- Example 5: “There is someone who loves someone.”
- Solution: $\exists x \, \exists y \, L(x,y)$
- Example 6: “Everyone loves himself”
- Solution: $\forall x \, L(x,x)$

NEGATIONS

$$\neg \forall x P(x) \Leftrightarrow \exists x \neg P(x).$$

$$\neg \exists x\, Q(x) \Leftrightarrow \forall x \neg Q(x).$$

TABLE 3 Negating Quantifiers.

Negation	Equivalent Statement	When Is Negation True?	When False?
$\neg \exists x P(x)$	$\forall x \neg P(x)$	$P(x)$ is false for every x.	There is an x for which $P(x)$ is true.
$\neg \forall x P(x)$	$\exists x \neg P(x)$	There is an x for which $P(x)$ is false.	$P(x)$ is true for every x.

Negating Nested Quantifiers

Example 1: Recall the logical expression developed three slides back:

$\exists w\ \forall a\ \exists f\ (P(w,f) \wedge Q(f,a))$

Part 1: Use quantifiers to express the statement that "There does not exist a woman who has taken a flight on every airline in the world."

Solution: $\neg\exists w\ \forall a\ \exists f\ (P(w,f) \wedge Q(f,a))$

Part 2: Now use De Morgan's Laws to move the negation as far inwards as possible.

Solution:

$\neg\exists w\ \forall a\ \exists f\ (P(w,f) \wedge Q(f,a))$

$\forall w\ \neg\ \forall a\ \exists f\ (P(w,f) \wedge Q(f,a))$ by De Morgan's for $\exists$

$\forall w\ \exists\ a\ \neg\ \exists f\ (P(w,f) \wedge Q(f,a))$ by De Morgan's for $\forall$

$\forall w\ \exists\ a\ \forall f\ \neg\ (P(w,f) \wedge Q(f,a))$ by De Morgan's for $\exists$

$\forall w\ \exists\ a\ \forall f\ (\neg P(w,f) \vee \neg Q(f,a))$ by De Morgan's for $\wedge$.

Part 3: Can you translate the result back into English?

Solution:

"For every woman there is an airline such that for all flights, this woman has not taken that flight or that flight is not on this airline"

Different quantification Sequence

- Let Q(x, y, z) be the statement "x + y = z." What are the truth values of the statements
- $\forall x \forall y \exists z Q(x, y, z)$ and $\exists z \forall x \forall y Q(x, y, z)$?

"For all real numbers x and for all real numbers y there is a real number z such that x + y = z,"

the statement is true.

"There is a real number z such that for all real numbers x and for all real numbers y it is true that x + y = z,"

the statement is false.

Table1

TABLE 2 Quantifications of Two Variables.		
Statement	**When True?**	**When False?**
$\forall x \forall y P(x,y)$ $\forall y \forall x P(x,y)$	P(x, y) is true for every pair x, y.	There is a pair x, y for which P(x, y) is false.
$\forall x \exists y P(x,y)$	For every x there is a y for which P(x, y) is true.	There is an x such that P(x, y) is false for every y.
$\exists x \forall y P(x,y)$	There is an x for which P(x, y) is true for every y.	For every x there is a y for which P(x, y) is false.
$\exists x \exists y P(x,y)$ $\exists y \exists x P(x,y)$	There is a pair x, y for which P(x, y) is true.	P(x, y) is false for every pair x, y.

Nesting of Quantifiers

Example: Let the u.d. of x and y be people.

Let $L(x,y)$=“x likes y” (a predicate w. 2 f.v.’s)

Then $\exists y\ L(x,y)$ = “There is someone whom x likes.” (A predicate w. 1 free variable, x)

Then $\forall x\ (\exists y\ L(x,y))$ =
“Everyone has someone whom they like.”

(a real proposition; no free variables left)

Quantifier Equivalence Laws

- Expanding quantifiers: If u.d.=a,b,c,…
  $\forall x\, P(x) \Leftrightarrow P(\mathrm{a}) \wedge P(\mathrm{b}) \wedge P(\mathrm{c}) \wedge \ldots$
  $\exists x\, P(x) \Leftrightarrow P(\mathrm{a}) \vee P(\mathrm{b}) \vee P(\mathrm{c}) \vee \ldots$
- From those, we can prove the laws:
  $\forall x\, P(x) \Leftrightarrow \neg\exists x\, \neg P(x)$
  $\exists x\, P(x) \Leftrightarrow \neg\forall x\, \neg P(x)$
- Which *propositional* equivalence laws can be used to prove this?

DeMorgan's

Ex1： Everyone love her baby

- Let
  - $P(x)$： x refer to people
  - $C(x)$： x refer to baby
  - $I(x, y)$： x is baby of y
  - $L(x, y)$： x loves y
- Proposition:
  - $\forall x \forall y(P(y) \wedge C(x) \wedge I(x, y) \to L(y, x))$

Ex2:

- Every teacher persuade his baby to be a teacher. One let his daughter to be a dancer. Proof: This person is not a teacher.
- Let:
  - S(x): x is a teacher
  - E(x): let his baby to be a teacher.

Example 12

- Translate the statement
- $\forall x\,(C(x) \vee \exists y(C(y) \wedge F(x, y)))$
- into English, where C(x) is "x has a computer," F(x,y) is "x and y are friends," and the universe of discourse for both x and y is the set of all students in NWPU.

The statement says that for every student x in NWPU x has a computer or there is a student y such that y has a computer and x and y are friends. In other words, every student in NWPU has a computer or has a friend who has a computer.

Example 13

- Translate the statement
- $\exists x \forall y \forall z(((F(x,y) \wedge F(x,z) \wedge (y \neq z)) \rightarrow \neg F(y,z)))$
- into English, where F(a,b) means a and b are friends and the universe of discourse for x, y, and z is the set of all students in your school.

This statement says that there is a student x such that for all students y and all students z other than y, if x and y are friends and x and z are friends, then y and z are not friends. In other words, there is a student none of whose friends are also friends with each other.

Proof

- hypothesis：
  - P_1：$\forall x\,(S(x) \rightarrow E(x))$
  - P_2：$\exists x\,(\neg E(x))$
- conclusion：
  - C：$\exists x\,(\neg E(x) \wedge \neg S(x))$
- We need proof：
  - $P_1 \wedge P_2 \rightarrow C$
  - $\forall x\,(S(x) \rightarrow E(x)) \wedge \exists x\,(\neg E(x)) \rightarrow \exists x\,(\neg E(x) \wedge \neg S(x))$
- Is tautology

Ex3:

- Can predicate logic say "there exist exactly two objects with property P"?

Yes:

$$\exists x\, \exists y \,\big(P(x) \wedge P(y) \wedge x \neq y \;\wedge \forall z \,\big(P(z) \rightarrow (z = x \vee z = y)\big)\big)$$

What's wrong with?

$$\exists x\, \exists y\, (P(x) \wedge P(y) \wedge x \neq y) \;\wedge \forall z \,\big(P(z) \rightarrow (z = x \vee z = y)\big)$$

This is a conjunction of two separate propositions. As a result, *x* and *y* are not bound, so this is not even a proposition

Example 16,17

- Express the statement "Everyone has exactly one best friend" as a logical expression.

$$\forall x \exists y \forall z(B(x, y) \land ((z \neq y) \rightarrow \neg B(x,z))).$$

- Express the statement "If somebody is female and is a parent, then this person is someone's mother" as a logical expression.

$$\forall x ((F(x) \land P(x)) \rightarrow \exists y M(x, y)).$$

Remember

- In propositional logic, we can *strictly speaking* only build formulas of finite size.
- E.g., we can write
  $P(\mathrm{a}) \wedge P(\mathrm{b})$
  $P(\mathrm{a}) \wedge P(\mathrm{b}) \wedge P(\mathrm{c})$
  $P(\mathrm{a}) \wedge P(\mathrm{b}) \wedge P(\mathrm{c}) \wedge P(\mathrm{d})$, etc.
- But this way, we could never say that all natural numbers have P

- In predicate logic, you can say this easily: $\forall x P(x)$
- Still, it's sometimes useful to pretend that propositional logic allows infinitely long formulas.

More Equivalence Laws

- $\forall x\, \forall y\, P(x,y) \Leftrightarrow \forall y\, \forall x\, P(x,y)$
  $\exists x\, \exists y\, P(x,y) \Leftrightarrow \exists y\, \exists x\, P(x,y)$
- $\forall x\, (P(x) \wedge Q(x)) \Leftrightarrow (\forall x\, P(x)) \wedge (\forall x\, Q(x))$
- $\exists x\, (P(x) \vee Q(x)) \Leftrightarrow (\exists x\, P(x)) \vee (\exists x\, Q(x))$
- How about this one?
  $\exists x\, (P(x) \wedge Q(x)) \Leftrightarrow (\exists x\, P(x)) \wedge (\exists x\, Q(x))$

More Equivalence Laws

- How about this one?

$$\exists x\, (P(x) \wedge Q(x)) \Leftrightarrow (\exists x\, P(x)) \wedge (\exists x\, Q(x))\ ?$$

- *This equivalence statement is false.*
  Counterexample:
  P(x): x's birthday is on 30 April
  Q(x): x's birthday is on 20 December

End of §1.4-1.5, Predicate Logic

- From these sections you should have learned:
  - Predicate logic notation & conventions
  - Conversions: predicate logic $\leftrightarrow$ clear English
  - Meaning of quantifiers, equivalences
  - Simple reasoning with quantifiers
- Upcoming topics:
  - Introduction to proof-writing.
  - Then: Set theory –
    - a language for talking about collections of objects.

Homework

- § 1.4 – 10, 18, 30, 40,54, 60
- § 1.5 – 10, 28, 34, 46,50

1 The Foundations: Logic and Proofs

-

1.1 Propositional Logic
1.2 Propositional Equivalences
1.3 Predicates and Quantifiers
1.4 Nested Quantifiers
1.5 Rules of Inference
1.6 Introduction to Proofs
1.7 Proof Methods and Strategy