

Chapter 1

The Foundations: Logic and Proofs

Xu Zi Kang

更新: June 29, 2019

1 Propositional Logic

The material on logical connectives is straightforward. Most difficulties with this material involve confusion between common English usage and precise mathematical definitions.

1.1 Gist:

Proposition: A *declarative* sentence that is *either true or false*, but not both.

Statements that can not be decided is not a proposition

Propositional/Statement variable: **p, q, r, s** ; Truth value: **T, F**

Propositional Calculus/Logic: The area of logic that deals with propositions

Compound proposition: Proposition formed from existing propositions using operators.

Operators:

- Not: $\neg p$. e.g. It is not the case that...
- Conjunction/And: $p \wedge q$. e.g. ... and/but ...
- Disjunction/or: $p \vee q$. e.g. ... or ...
- Exclusive or: $p \oplus q$.
- Conditional Statement/Implication: $p \rightarrow q$.
 - if p, then q
 - p only if q
 - p implies q
 - q is a sufficient condition for p
 - q is a necessary condition for p
 - q follows from/if/when/whenever p
 - q unless $\neg p$
False only when p is true, but q is false; when p is false, $p \rightarrow q$ is defined to be true.
- **Converse:** $p \rightarrow q$ to $q \rightarrow p$

- **Inverse:** $p \rightarrow q$ to $\neg p \rightarrow \neg q$
- **Contrapositive:** Converse and inverse. $p \rightarrow q$ to $\neg q \rightarrow \neg p$
- Biconditional statement: $p \leftrightarrow q$.
 - p if and only if q
 - p is necessary and sufficient for q
 - if p then q, and conversely
 - p iff q

$p \leftrightarrow q == (q \rightarrow p)$, only true when p and q have the same value

2 Applications of Propositional Logic

Introduces some important applications of propositional logic, including many important applications in computer science.

2.1 Advice:

See Example 7, which introduces one of Raymond Smullyan's knights and knaves puzzles.

2.2 Gist:

Consistent system specification:

- no conflicting requirements;
- a set of values to satisfy all the statements translated;
- does not necessarily have a real usage;

3 Propositional equivalences

Shows how propositional equivalences are established and to introduce the most important such equivalences

3.1 Advice:

Introduce the notion of propositional equivalences by establishing De Morgan's laws—see Example 2, Table 6 presents basic propositional equivalences. We will see similar tables for set identities in Section 2.2 and for Boolean algebra in Section 12.1. Mention that these properties hold in a wide variety of settings that all fit into one abstract form.

3.2 Gist:

Classification of compound proposition

- Tautology Always true
- Contradiction Always false
- Contingency Neither a tautology nor a Contradiction

Logical Equivalence: Compound propositions that have the same truth values in all possible cases $p \leftrightarrow q$ is a tautology $p \equiv q, p \Leftrightarrow q$

3.3 Logical Equivalences:

Equivalence	Name
$p \wedge T \equiv p$ $p \vee F \equiv p$	Identity laws
$p \vee T \equiv T$ $p \wedge F \equiv F$	Domination laws
$p \vee p \equiv p$ $p \wedge p \equiv p$	Idempotent laws
$\neg(\neg p) \equiv p$	Double negation law
$p \vee q \equiv q \vee p$ $p \wedge q \equiv q \wedge p$	Commutative laws
$(p \vee q) \vee r \equiv p \vee (q \vee r)$ $(p \wedge q) \wedge r \equiv p \wedge (q \wedge r)$	Distributive laws
$\neg(p \wedge q) \equiv \neg p \vee \neg q$ $\neg(p \vee q) \equiv \neg p \wedge \neg q$	De Morgan's laws
$p \vee (p \wedge q) \equiv p$ $p \wedge (p \vee q) \equiv p$	Absorption laws
$p \vee \neg p \equiv T$ $p \wedge \neg p \equiv F$	Neagtion laws
$p \rightarrow q \equiv \neg p \vee q$ $p \leftrightarrow q \equiv (p \wedge q) \vee (\neg p \wedge \neg q)$	Implication laws

Mine: $\neg(p \rightarrow) \equiv p \wedge \neg q$

Mine: $p \vee (\neg p \wedge q) \equiv p \vee q$ See Section 3.1 No.30 **Prove Logical Equivalence:** Truth table or developing a series of logical equivalences. **Prove Not Logically Equivalent:** (Simplify first) and find a counterexample. **Propositional Satisfiability:** An assignment of truth values that makes it true. Truth table, or whether its negation is a tautology. (or say whether itself is a Contradiction?)

4 Predicates and Quantifiers

Introduced predicate logic, especially existential and universal quantification.

4.1 Advice:

This section is important. It's easy to have trouble proving statements that involve quantification, including the inductive step in mathematical induction.

A quantification is not well-defined unless the domain is specified and that changing the domain can change the truth value of the quantification. Example 23 and 24 showed how to translate a particular English sentence into a logical statement; Example 25 illustrates how to use Quantifiers in system specifications; Example 26 and 27, taken from Lewis Carroll, illustrate the subtleties of translating English sentences into correct statements involving predicate and propositional logic.

4.2 Gist:

Predicate: Statements involving variables, the variables are called subjects, the other is called a predicate.

The statement of $P(x)$ is also called the value of Propositional function P at X .

Precondition: Conditions for valid input.

Postcondition: Conditions for correct output.

Qualification: Express the extent to which a predicate is true over a range of elements.

Domain / universe of discourse / domain: A predicate is true for a variable in a particular domain.

Predicate calculus: The area of logic that deals with predicates and quantifiers.

Universal qualification: For every element $\forall xP(x)$: For all every x $P(x)$

Existential qualification: For one or more element $\exists xP(x)$: For one or more element.

Abbreviated qualifier notation: Use condition for domain. Or use $\exists xP(x) \rightarrow Q(x)$ for using $P(x)$ as condition for domain.

Quantifiers (\exists and \forall) have higher precedence than all logical operators from propositional calculus. i.e. They absorb less.

Occurrence of variable is bound: Quantifier is used on the variable.

All the variables that occur in a propositional function must be bound or set equal to a particular value to turn it into a proposition.

Scope of quantifier: The part of a logical expression to which a quantifier is applied.

The same letter is often used to represent variables bound by different quantifiers with scopes that do not overlap. Statements involving predicates and quantifiers are Logically Equivalent: If and only if they have the same truth value *no matter* which **predicates** are substituted into these statements and which **domain** of discourse is used for the variables in these propositional functions.

$$S \equiv T$$

$$\forall xP(x) \wedge Q(x) \equiv \forall xP(x) \wedge \forall xQ(x)$$

$$\exists xP(x) \vee Q(x) \equiv \exists xP(x) \vee \exists xQ(x)$$

De Morgan's laws for quantifiers:

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

$$\neg \exists x Q(x) \equiv \forall x \neg Q(x)$$

Translating from English into Logical Expression . . .

Using Quantifiers in System Specifications: . . .

5 Nested Quantifiers

Explained how to work with nested quantifiers and makes clear that the order of quantification matters.

5.1 Gist:

Nested quantifiers : One quantifier is within the scope of another.

Understanding Statements involving Nested Quantifiers: . . .

The order of the quantifiers is important, unless all the quantifiers are universal quantifiers or all are existential quantifiers.

6 Rules of Inference

Introduced the notion of a valid argument and rules of inference for propositional logic. Explained how to use rules of inference to build correct arguments in propositional calculus. Moreover, introduced rules of inference for predicate logic and how to use these rules of inference to build correct arguments in predicate logic. Showed how rules of inference for propositional calculus and predicate calculus can be combined.

7 Introduction to Proofs

Introduced the notion of proof and basic methods of proof, including direct proof, proof by contraposition, and proof by contradiction.

7.1 Gist:

Theorem / Fact / Result(定理): A statement that can be shown to be true.

Propositions: Less important theorems.

Axiom / Postulate (公理): Statements we assume to be true.

Lemma (引理): A less important theorem that is helpful in the proof of other results. (plural lemmas or lemmata)

Corollary (推论): Theorem that can be established directly from a theorem that has been proved.

Conjecture (猜想): Statement that is being proposed to be a true statement.

Direct Proof: Direct proof of a conditional statement $p \rightarrow q$ is constructed when the first step is the assumption that p is true;

subsequent steps are constructed using rules of inference, with the final step showing that q must also be true.

Indirect Proofs: Proofs that do not start with the premises and end with the conclusion.

Proof by Contraposition(证明逆否命题): We take $\neg q$ as a premise, and using axioms, definitions, and previously proven theorems, together with rules of inference, we show that $\neg p$ must follow.

Vacuous Proofs: If we can show that p is false, then we have a proof of the conditional statement $p \rightarrow q$.

Trivial Proof: By showing that q is true, it follows that $p \rightarrow q$ must also be true.

Proofs by Contradiction (反证): We can prove that p is true if we can show that $\neg p \rightarrow (r \wedge \neg r)$ is true for some proposition r . i.e. r is premise and $\neg r$ is proved if $\neg p$.

To rewrite a proof by contraposition of $p \rightarrow q$ as a proof by contradiction, we suppose that both p and $\neg q$ are true. Then, we use the step from the proof of $\neg q \rightarrow \neg p$ to show that $\neg p$ is true.

Proofs of Equivalence: $(p \leftrightarrow q) \leftrightarrow (p \rightarrow q) \wedge (q \rightarrow p)$. (if and only if) $(p_1 \leftrightarrow p_2 \leftrightarrow \dots \leftrightarrow p_n) \leftrightarrow (p_1 \leftrightarrow p_2) \wedge (p_2 \leftrightarrow p_3) \wedge \dots \wedge (p_n \leftrightarrow p_1)$

Counterexamples: To show that a statement of the form $\forall x P(x)$ is false. we only need to find a counterexample.

Mistakes in proofs: Division by zero, Affirming the conclusion, Denying the hypothesis, Begging the question.