

CSC3001

Discrete Mathematics

Tutorial 02



01 Overview

Objectives of This Course

- To be familiar with formal **mathematical reasoning**, e.g. logic, proofs
- To improve **problem solving skills**, e.g. induction, recursion
- To learn basic **mathematical concepts**, e.g. sets, functions, graphs
- To see the **connections** between discrete mathematics and computer science

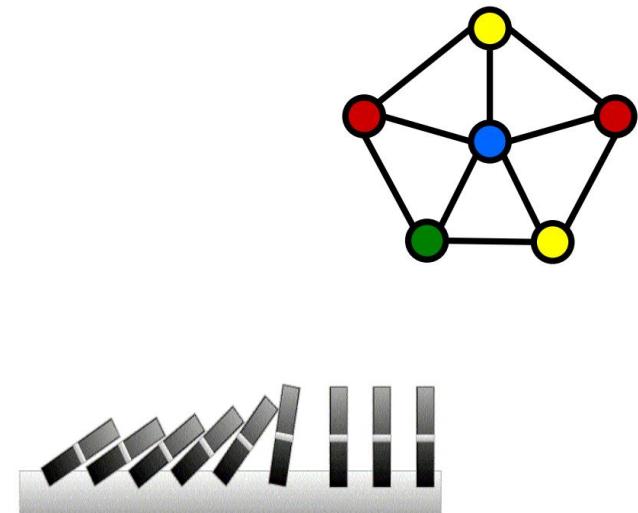
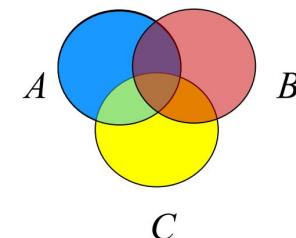
Knowledge will be used in future courses, e.g.,
CSC 3100 Data structures,
CSC 4120 Design and analysis of algorithms



01 Overview

Course schedule

Week	Tentative topics
1	introduction, propositional logic
2	sets, first order logic
3	methods of proofs
4	mathematical induction
5	recursion
6	greatest common divisor
7	modular arithmetic, Chinese remainder theorem
8	topics in number theory
9	graphs
10	graph matching
11	graph coloring
12	combinatorial proofs and its principles
13	counting by mapping
14	cardinality



01 Overview

Assignments and exams

- **5 Assignments**

- Pay attention to Blackboard announcements!
- The first assignment will be released on Sep 23 and due on Oct 14
- The subsequent assignments will be released biweekly starting on Oct 14
 - Each will be due in two weeks
- Submit a single PDF file of your answers to Blackboard

- **Midterm exam**

- **9:00AM - 11:00AM, November 5, Saturday**
- Exam venue will be announced before the exam

- **Final exam**

- To be announced by the registry office



01 Overview



Office hours and locations

Name	Email	Office hour	Location
Baoxiang Wang (Lead Instructor)	bxiangwang@cuhk.edu.cn	Mon 5-6PM	Daoyuan 503
Zitan Chen (Instructor)	chenztan@cuhk.edu.cn	Tue 4-5PM	Research A 417
Zhihan Ning (Lead TA)	zhihanning@link.cuhk.edu.cn	Tue 7-9PM	Daoyuan 318
Dmitry Rybin (TA)	dmitryrybin@link.cuhk.edu.cn	Fri 3-5PM	Research B 309A
Sirong Dai (TA)	sirongdai@link.cuhk.edu.cn	Wed 3-5PM	Zhixin 4th floor
Ziwei Yang (TA)	yangziwei0622@163.com	Fri 6-7PM	Daoyuan 105
TBA (TA)			



01 Overview

My Teaching Schedule > 2022-23 Term 1 > CUHK(SZ)					
Class	Class Title	Enrolled	Days & Times	Room	Class Dates
 CSC 3001-L01 (3174)	Discrete Mathematics 173 (Lecture)		MoWe 15:30 - 16:50	Teaching B Bldg 201	Sep 5, 2022-Dec 16, 2022
 CSC 3001-L02 (3175)	Discrete Mathematics 115 (Lecture)		TuTh 10:30 - 11:50	Teaching B Bldg 201	Sep 5, 2022-Dec 16, 2022
 CSC 3001-T01 (3176)	Discrete Mathematics 58 (Tutorial)		Mo 18:00 - 18:50	Teaching A Bldg 304	Sep 5, 2022-Dec 16, 2022
 CSC 3001-T02 (3177)	Discrete Mathematics 65 (Tutorial)		We 18:00 - 18:50	Teaching A Bldg 304	Sep 5, 2022-Dec 16, 2022
 CSC 3001-T03 (3178)	Discrete Mathematics 48 (Tutorial)		Fr 18:00 - 18:50	Teaching A Bldg 304	Sep 5, 2022-Dec 16, 2022
 CSC 3001-T04 (3179)	Discrete Mathematics 50 (Tutorial)		Tu 19:00 - 19:50	Teaching A Bldg 304	Sep 5, 2022-Dec 16, 2022
 CSC 3001-T05 (3180)	Discrete Mathematics 55 (Tutorial)		We 19:00 - 19:50	Teaching A Bldg 305	Sep 5, 2022-Dec 16, 2022
 CSC 3001-T06 (3181)	Discrete Mathematics 12 (Tutorial)		Fr 19:00 - 19:50	Teaching A Bldg 304	Sep 5, 2022-Dec 16, 2022



01 Overview



**Tutorial
Instructor**

Dmitry Rybin

Zoom Link

[https://cuhk-edu-cn.zoom.us/j/3681470910?
pwd=VGtyR1dvRXFyVGQ3bVBOSTVtK1B2Zz09](https://cuhk-edu-cn.zoom.us/j/3681470910?pwd=VGtyR1dvRXFyVGQ3bVBOSTVtK1B2Zz09)

ID

368 147 0910
(code: 936915)



02 Instruction

Topics:

- **Set Theory**
- **Logic and Proofs**



03 Set Theory

Topics:

1. **Definition**
2. **Defining Sets by Properties**
3. **Size of a Set (finite)**
4. **Subset, Proper Subset, Equality**
5. **Basic Operations on Sets**
6. **Partitions of Sets**
7. **Power Sets**
8. **Cartesian Products**
9. **Set Identities**
10. **Summary**



03 Set Theory

3.1 Definition

A set is an unordered collection of distinct objects.

Well-known Sets

Z: the set of all integers

Z⁺: the set of all positive integers

Z⁻: the set of all negative integers

N: the set of all nonnegative integers

R: the set of all real numbers

Q: the set of all rational numbers

C: the set of all complex numbers

Ø: the empty set, i.e., the set with no elements



03 Set Theory

3.2 Defining Sets by Properties

We use the notation $\{x \in A \mid P(x)\}$

to define the set as the **set of elements**, x ,

in A **such that** x satisfies property P .



03 Set Theory

3.3 Size of a Set (finite)

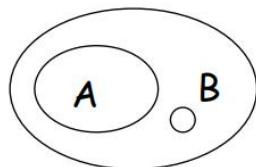
Definition: The **size** of a set S , denoted by $|S|$, is defined as the number of elements contained in S .



03 Set Theory

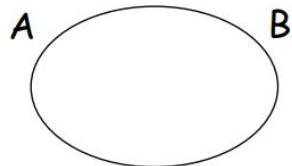
3.4 Subset, Proper Subset, Equality

Definition: Given two sets A and B , we say A is a **proper subset** of B , denoted by $A \subset B$, if every element of A is an element of B , But there is an element in B that is not contained in A .



Fact: If $A \subset B$, then $|A| < |B|$.

Definition: Given two sets A and B , we say $A = B$ if $A \subseteq B$ and $B \subseteq A$.



Fact: If $A = B$, then $|A| = |B|$.



03 Set Theory

3.5 Basic Operations on Sets

- intersection
- union
- difference
- complement



03 Set Theory

3.6 Partitions of Sets

A collection of nonempty sets $\{A_1, A_2, \dots, A_n\}$ is a **partition** of a set A if and only if

$$A = A_1 \cup A_2 \cup \dots \cup A_n$$

A_1, A_2, \dots, A_n are **mutually disjoint** (or **pairwise disjoint**).



03 Set Theory

3.7 Power Sets

power set: $\text{pow}(A) ::= \{S \mid S \subseteq A\}$



03 Set Theory

3.8 Cartesian Products

Definition: Given two sets A and B , the **Cartesian product** $A \times B$ is the set of all ordered pairs (a,b) , where a is in A and b is in B . That is,

$$A \times B = \{(a,b) \mid a \in A, b \in B\}$$



03 Set Theory

3.9 Set Identities

Set Identities

Let A, B, C be subsets of a universal set U .

Commutative Law: (a) $A \cup B = B \cup A$ and (b) $A \cap B = B \cap A$

Associative Law: (a) $(A \cup B) \cup C = A \cup (B \cup C)$

(b) $(A \cap B) \cap C = A \cap (B \cap C)$

Distributive Law: (a) $A \cup (B \cap C) = (A \cup B) \cap (A \cup C)$

(b) $A \cap (B \cup C) = (A \cap B) \cup (A \cap C)$

Identity Law: (a) $A \cup \emptyset = A$ and (b) $A \cap U = A$

Complement Law: (a) $A \cup A^c = U$ and (b) $A \cap A^c = \emptyset$

De Morgan's Law: (a) $(A \cup B)^c = A^c \cap B^c$ and (b) $(A \cap B)^c = A^c \cup B^c$

**Set difference
Law:** $A - B = A \cap B^c$



03 Set Theory

3.10 Summary

Summary

Recall what we have covered so far.

- Basic Definitions (defining sets, membership, subsets, size)
- Operations on Sets (intersection, union, difference, complement, partition, power set, Cartesian product)
- Set Identities (basic laws, e.g., De Morgan's law, checking set identities - proof & disproof, algebraic)



03 Set Theory

3.11 Exercise

1. Compute the power set

(1) {a, b, c}

(2) {1, {2, 3}}

(3) {∅}

(4) {∅, {∅}}



03 Set Theory

3.11 Exercise

1. Compute the power set

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

$$(2) \{1, \{2, 3\}\} \quad P(A) = \{\emptyset, \{1\}, \{\{2, 3\}\}, \{1, \{2, 3\}\}\}$$

$$(3) \{\emptyset\} \quad P(A) = \{\emptyset, \{\emptyset\}\}$$

$$(4) \{\emptyset, \{\emptyset\}\} \quad P(A) = \{\emptyset, \{1\}, \{\{2, 3\}\}, \{1, \{2, 3\}\}\}$$



03 Set Theory

3.11 Exercise

2. True Or False

- (1) $\emptyset \subseteq \emptyset$
- (2) $\emptyset \in \emptyset$
- (3) $\emptyset \subseteq \{\emptyset\}$
- (4) $\emptyset \in \{\emptyset\}$
- (5) $\{a, b\} \subseteq \{a, b, c, \{a, b, c\}\}$
- (6) $\{a, b\} \in \{a, b, c, \{a, b\}\}$
- (7) $\{a, b\} \subseteq \{a, b, \{\{a, b\}\}\}$
- (8) $\{a, b\} \in \{a, b, \{\{a, b\}\}\}$



03 Set Theory

3.11 Exercise

2. True Or False

- (1) $\emptyset \subseteq \emptyset$ 真
- (2) $\emptyset \in \emptyset$ 假
- (3) $\emptyset \subseteq \{\emptyset\}$ 真
- (4) $\emptyset \in \{\emptyset\}$ 真
- (5) $\{a, b\} \subseteq \{a, b, c, \{a, b, c\}\}$ 真
- (6) $\{a, b\} \in \{a, b, c, \{a, b\}\}$ 真
- (7) $\{a, b\} \subseteq \{a, b, \{\{a, b\}\}\}$ 真
- (8) $\{a, b\} \in \{a, b, \{\{a, b\}\}\}$ 假



04 Logic and Proofs

Topics:

- 1. Propositional Logic and its Application**
- 2. Propositional Equivalences**
- 3. Predicates and Quantifiers**
- 4. Rules of Inference**
- 5. Methods of Proof**
- 6. Summary**



04 Logic and Proofs

4.1 Propositional Logic and its Application

- Basic building blocks of logic——**propositions**.
- A proposition is a declarative sentence. (either true or false)



04 Logic and Proofs

4.1 Propositional Logic and its Application

- We use letters to denote **propositional variables** (or **statement variables**), that is, variables that represent propositions. The conventional letters used for propositional variables are p,q,s,...
- The **truth value** of a proposition is true, denoted by T, if it is a true proposition, and the truth value of a proposition is false, denoted by F, if it is a false proposition.
- The area of logic that deals with propositions is called the **propositional calculus** or **propositional logic**.
- Many mathematical statements are constructed by combining one or more propositions. New propositions, called **compound propositions**, are formed from existing propositions using **logical operators**.



04 Logic and Proofs

4.1 Propositional Logic and its Application

DEFINITION 1 Let p be a proposition. The **negation of p** , denoted by $\neg p$ (also denoted by \overline{p}), is the statement

“It is not the case that p .”

The proposition $\neg p$ is read “not p .” The truth value of the negation of p , $\neg p$, is the opposite of the truth value of p .

TABLE 1 The Truth Table for the Negation of a Proposition.

p	$\neg p$
T	F
F	T

Table 1 displays the **truth table** for the negation of a proposition p . This table has a row for each of the two possible truth values of a proposition p . Each row shows the truth value of $\neg p$ corresponding to the truth value of p for this row.

The negation of a proposition can also be considered the result of the operation of the **negation operator** on a proposition. The negation operator constructs a new proposition from a single existing proposition. We will now introduce the logical operators that are used to form new propositions from two or more existing propositions. These logical operators are also called **connectives**.



04 Logic and Proofs

4.1 Propositional Logic and its Application

DEFINITION 2

Let p and q be propositions. The *conjunction* of p and q , denoted by $p \wedge q$, is the proposition “ p and q .” The conjunction $p \wedge q$ is true when both p and q are true and is false otherwise.

DEFINITION 3

Let p and q be propositions. The *disjunction* of p and q , denoted by $p \vee q$, is the proposition “ p or q .” The disjunction $p \vee q$ is false when both p and q are false and is true otherwise.

DEFINITION 4

Let p and q be propositions. The *exclusive or* of p and q , denoted by $p \oplus q$, is the proposition that is true when exactly one of p and q is true and is false otherwise.

TABLE 2 The Truth Table for the Conjunction of Two Propositions.

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

TABLE 3 The Truth Table for the Disjunction of Two Propositions.

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



04 Logic and Proofs

4.1 Propositional Logic and its Application

DEFINITION 5

Let p and q be propositions. The *conditional statement* $p \rightarrow q$ is the proposition “if p , then q .” The conditional statement $p \rightarrow q$ is false when p is true and q is false, and true otherwise. In the conditional statement $p \rightarrow q$, p is called the *hypothesis* (or *antecedent* or *premise*) and q is called the *conclusion* (or *consequence*).

The statement $p \rightarrow q$ is called a conditional statement because $p \rightarrow q$ asserts that q is true on the condition that p holds. A conditional statement is also called an **implication**.

“if p , then q ”

“if p , q ”

“ p is sufficient for q ”

“ q if p ”

“ q when p ”

“a necessary condition for p is q ”

“ q unless $\neg p$ ”

“ p implies q ”

“ p only if q ”

“a sufficient condition for q is p ”

“ q whenever p ”

“ q is necessary for p ”

“ q follows from p ”



04 Logic and Proofs

4.1 Propositional Logic and its Application

CONVERSE, CONTRAPOSITIVE, AND INVERSE We can form some new conditional statements starting with a conditional statement $p \rightarrow q$. In particular, there are three related conditional statements that occur so often that they have special names. The proposition $q \rightarrow p$ is called the **converse** of $p \rightarrow q$. The **contrapositive** of $p \rightarrow q$ is the proposition $\neg q \rightarrow \neg p$.

The proposition $\neg p \rightarrow \neg q$ is called the **inverse** of $p \rightarrow q$. We will see that of these three conditional statements formed from $p \rightarrow q$, only the contrapositive always has the same truth value as $p \rightarrow q$.

We first show that the contrapositive, $\neg q \rightarrow \neg p$, of a conditional statement $p \rightarrow q$ always has the same truth value as $p \rightarrow q$. To see this, note that the contrapositive is false only when $\neg p$ is false and $\neg q$ is true, that is, only when p is true and q is false. We now show that neither the converse, $q \rightarrow p$, nor the inverse, $\neg p \rightarrow \neg q$, has the same truth value as $p \rightarrow q$ for all possible truth values of p and q . Note that when p is true and q is false, the original conditional statement is false, but the converse and the inverse are both true.

When two compound propositions always have the same truth value we call them **equivalent**, so that a conditional statement and its contrapositive are equivalent. The converse and the ~~inversé~~ of a conditional statement are also equivalent, as the reader can verify, but neither is equivalent to the original conditional statement. (We will study equivalent propositions in Section 1.3.) Take note that one of the most common logical errors is to assume that the converse or the inverse of a conditional statement is equivalent to this conditional statement.



04 Logic and Proofs

4.1 Propositional Logic and its Application

DEFINITION 6 Let p and q be propositions. The *biconditional statement* $p \leftrightarrow q$ is the proposition “ p if and only if q .” The biconditional statement $p \leftrightarrow q$ is true when p and q have the same truth values, and is false otherwise. Biconditional statements are also called *bi-implications*.

TABLE 6 The Truth Table for the Biconditional $p \leftrightarrow q$.

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



04 Logic and Proofs

4.1 Propositional Logic and its Application

Truth Tables of Compound Propositions

We have now introduced four important logical connectives—conjunctions, disjunctions, conditional statements, and biconditional statements—as well as negations. We can use these connectives to build up complicated compound propositions involving any number of propositional variables. We can use truth tables to determine the truth values of these compound propositions, as Example 11 illustrates. We use a separate column to find the truth value of each compound expression that occurs in the compound proposition as it is built up. The truth values of the compound proposition for each combination of truth values of the propositional variables in it is found in the final column of the table.

TABLE 7 The Truth Table of $(p \vee \neg q) \rightarrow (p \wedge q)$.

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



04 Logic and Proofs

4.1 Propositional Logic and its Application

Precedence of Logical Operators

TABLE 8
Precedence of
Logical Operators.

Operator	Precedence
\neg	1
\wedge	2
\vee	3
\rightarrow	4
\leftrightarrow	5

We can construct compound propositions using the negation operator and the logical operators defined so far. We will generally use parentheses to specify the order in which logical operators in a compound proposition are to be applied. For instance, $(p \vee q) \wedge (\neg r)$ is the conjunction of $p \vee q$ and $\neg r$. However, to reduce the number of parentheses, we specify that the negation operator is applied before all other logical operators. This means that $\neg p \wedge q$ is the conjunction of $\neg p$ and q , namely, $(\neg p) \wedge q$, not the negation of the conjunction of p and q , namely $\neg(p \wedge q)$.

Another general rule of precedence is that the conjunction operator takes precedence over the disjunction operator, so that $p \wedge q \vee r$ means $(p \wedge q) \vee r$ rather than $p \wedge (q \vee r)$. Because this rule may be difficult to remember, we will continue to use parentheses so that the order of the disjunction and conjunction operators is clear.

Finally, it is an accepted rule that the conditional and biconditional operators \rightarrow and \leftrightarrow have lower precedence than the conjunction and disjunction operators, \wedge and \vee . Consequently, $p \vee q \rightarrow r$ is the same as $(p \vee q) \rightarrow r$. We will use parentheses when the order of the conditional operator and biconditional operator is at issue, although the conditional operator has precedence over the biconditional operator. Table 8 displays the precedence levels of the logical operators, \neg , \wedge , \vee , \rightarrow , and \leftrightarrow .



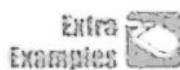
04 Logic and Proofs

4.1 Propositional Logic and its Application

● Translating

EXAMPLE 1 How can this English sentence be translated into a logical expression?

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



Solution: There are many ways to translate this sentence into a logical expression. Although it is possible to represent the sentence by a single propositional variable, such as p , this would not be useful when analyzing its meaning or reasoning with it. Instead, we will use propositional variables to represent each sentence part and determine the appropriate logical connectives between them. In particular, we let a , c , and f represent “You can access the Internet from campus,” “You are a computer science major,” and “You are a freshman,” respectively. Noting that “only if” is one way a conditional statement can be expressed, this sentence can be represented as

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



04 Logic and Proofs

4.1 Propositional Logic and its Application

● Logic Circuits

A **logic circuit** (or **digital circuit**) receives input signals p_1, p_2, \dots, p_n , each a bit [either 0 (off) or 1 (on)], and produces output signals s_1, s_2, \dots, s_n , each a bit. In this section we will restrict our attention to logic circuits with a single output signal; in general, digital circuits may have multiple outputs.

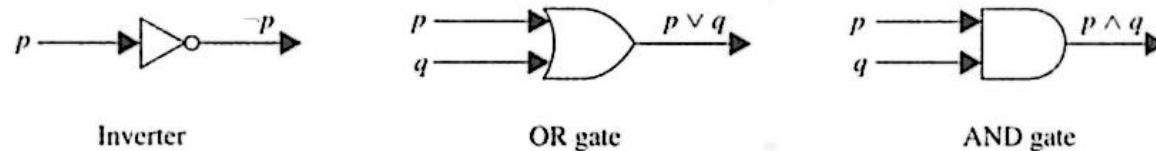


FIGURE 1 Basic logic gates.

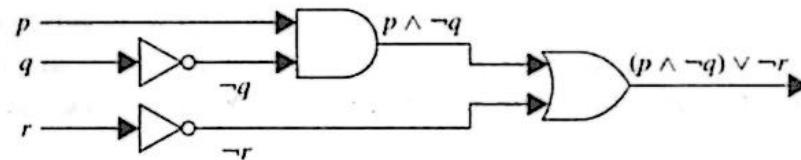


FIGURE 2 A combinatorial circuit.



04 Logic and Proofs

4.2 Propositional Equivalences

DEFINITION 1

A compound proposition that is always true, no matter what the truth values of the propositional variables that occur in it, is called a *tautology*. A compound proposition that is always false is called a *contradiction*. A compound proposition that is neither a tautology nor a contradiction is called a *contingency*.

DEFINITION 2

The compound propositions p and q are called *logically equivalent* if $p \leftrightarrow q$ is a tautology. The notation $p \equiv q$ denotes that p and q are logically equivalent.

Remark: The symbol \equiv is not a logical connective, and $p \equiv q$ is not a compound proposition but rather is the statement that $p \leftrightarrow q$ is a tautology. The symbol \Leftrightarrow is sometimes used instead of \equiv to denote logical equivalence.

TABLE 2 De Morgan's Laws.

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

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



04 Logic and Proofs

4.3 Predicates and Quantifiers

● Predicates

EXAMPLE 2 Let $A(x)$ denote the statement “Computer x is under attack by an intruder.” Suppose that of the computers on campus, only CS2 and MATH1 are currently under attack by intruders. What are truth values of $A(\text{CS1})$, $A(\text{CS2})$, and $A(\text{MATH1})$?

Solution: We obtain the statement $A(\text{CS1})$ by setting $x = \text{CS1}$ in the statement “Computer x is under attack by an intruder.” Because CS1 is not on the list of computers currently under attack, we conclude that $A(\text{CS1})$ is false. Similarly, because CS2 and MATH1 are on the list of computers under attack, we know that $A(\text{CS2})$ and $A(\text{MATH1})$ are true. 

EXAMPLE 3 Let $Q(x, y)$ denote the statement “ $x = y + 3$.” What are the truth values of the propositions $Q(1, 2)$ and $Q(3, 0)$?



Solution: To obtain $Q(1, 2)$, set $x = 1$ and $y = 2$ in the statement $Q(x, y)$. Hence, $Q(1, 2)$ is the statement “ $1 = 2 + 3$,” which is false. The statement $Q(3, 0)$ is the proposition “ $3 = 0 + 3$,” which is true. 

EXAMPLE 5 What are the truth values of the propositions $R(1, 2, 3)$ and $R(0, 0, 1)$?

 *Solution:* The proposition $R(1, 2, 3)$ is obtained by setting $x = 1$, $y = 2$, and $z = 3$ in the statement $R(x, y, z)$. We see that $R(1, 2, 3)$ is the statement “ $1 + 2 = 3$,” which is true. Also note that $R(0, 0, 1)$, which is the statement “ $0 + 0 = 1$,” is false. 



04 Logic and Proofs

4.3 Predicates and Quantifiers

● Quantifiers

DEFINITION 1 The *universal quantification* of $P(x)$ is the statement
“ $P(x)$ for all values of x in the domain.”

The notation $\forall x P(x)$ denotes the universal quantification of $P(x)$. Here \forall is called the **universal quantifier**. We read $\forall x P(x)$ as “for all $x P(x)$ ” or “for every $x P(x)$.” An element for which $P(x)$ is false is called a **counterexample** of $\forall x P(x)$.

DEFINITION 2 The *existential quantification* of $P(x)$ is the proposition
“There exists an element x in the domain such that $P(x)$.”

We use the notation $\exists x P(x)$ for the existential quantification of $P(x)$. Here \exists is called the **existential quantifier**.

TABLE 1 Quantifiers.

<i>Statement</i>	<i>When True?</i>	<i>When False?</i>
$\forall x P(x)$	$P(x)$ is true for every x .	There is an x for which $P(x)$ is false. $P(x)$ is false for every x .
$\exists x P(x)$	There is an x for which $P(x)$ is true.	



04 Logic and Proofs

4.3 Predicates and Quantifiers

● Quantifiers

Precedence of Quantifiers

The quantifiers \forall and \exists have higher precedence than all logical operators from propositional calculus. For example, $\forall x P(x) \vee Q(x)$ is the disjunction of $\forall x P(x)$ and $Q(x)$. In other words, it means $(\forall x P(x)) \vee Q(x)$ rather than $\forall x(P(x) \vee Q(x))$.

Binding Variables

When a quantifier is used on the variable x , we say that this occurrence of the variable is **bound**. An occurrence of a variable that is not bound by a quantifier or set equal to a particular value is said to be **free**. 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. This can be done using a combination of universal quantifiers, existential quantifiers, and value assignments.

The part of a logical expression to which a quantifier is applied is called the **scope** of this quantifier. Consequently, a variable is free if it is outside the scope of all quantifiers in the formula that specify this variable.

EXAMPLE 18 In the statement $\exists x(x + y = 1)$, the variable x is bound by the existential quantification $\exists x$, but the variable y is free because it is not bound by a quantifier and no value is assigned to this variable. This illustrates that in the statement $\exists x(x + y = 1)$, x is bound, but y is free.



04 Logic and Proofs

4.3 Predicates and Quantifiers

● Quantifiers

TABLE 2 De Morgan's Laws for Quantifiers.

<i>Negation</i>	<i>Equivalent Statement</i>	<i>When Is Negation True?</i>	<i>When False?</i>
$\neg\exists x P(x)$	$\forall x \neg P(x)$	For every x , $P(x)$ is false.	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 .



04 Logic and Proofs

4.3 Predicates and Quantifiers

● Nested Quantifiers

In Section 1.4 we defined the existential and universal quantifiers and showed how they can be used to represent mathematical statements. We also explained how they can be used to translate English sentences into logical expressions. However, in Section 1.4 we avoided **nested quantifiers**, where one quantifier is within the scope of another, such as

$$\forall x \exists y (x + y = 0).$$

Note that everything within the scope of a quantifier can be thought of as a propositional function. For example,

$$\forall x \exists y (x + y = 0)$$

is the same thing as $\forall x Q(x)$, where $Q(x)$ is $\exists y P(x, y)$, where $P(x, y)$ is $x + y = 0$.



04 Logic and Proofs

4.3 Predicates and Quantifiers

● The Order of Quantifiers

Many mathematical statements involve multiple quantifications of propositional functions involving more than one variable. It is important to note that the order of the quantifiers is important, unless all the quantifiers are universal quantifiers or all are existential quantifiers.

TABLE 1 Quantifications of Two Variables.

<i>Statement</i>	<i>When True?</i>	<i>When False?</i>
$\forall x \forall y P(x, y)$ $\forall y \forall x P(x, y)$	$P(x, y)$ 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 .



04 Logic and Proofs

4.4 Rules of Inference

Later in this chapter we will study proofs. Proofs in mathematics are valid arguments that establish the truth of mathematical statements. By an **argument**, we mean a sequence of statements that end with a conclusion. By **valid**, we mean that the conclusion, or final statement of the argument, must follow from the truth of the preceding statements, or **premises**, of the argument. That is, an argument is valid if and only if it is impossible for all the premises to be true and the conclusion to be false. To deduce new statements from statements we already have, we use rules of inference which are templates for constructing valid arguments. Rules of inference are our basic tools for establishing the truth of statements.

Before we study mathematical proofs, we will look at arguments that involve only compound propositions. We will define what it means for an argument involving compound propositions to be valid. Then we will introduce a collection of rules of inference in propositional logic. These rules of inference are among the most important ingredients in producing valid arguments. After we illustrate how rules of inference are used to produce valid arguments, we will describe some common forms of incorrect reasoning, called **fallacies**, which lead to invalid arguments.



04 Logic and Proofs

4.4 Rules of Inference

Valid Arguments in Propositional Logic

Consider the following argument involving propositions (which, by definition, is a sequence of propositions):

“If you have a current password, then you can log onto the network.”

“You have a current password.”

Therefore,

“You can log onto the network.”

We would like to determine whether this is a valid argument. That is, we would like to determine whether the conclusion “You can log onto the network” must be true when the premises “If you have a current password, then you can log onto the network” and “You have a current password” are both true.

Before we discuss the validity of this particular argument, we will look at its form. Use p to represent “You have a current password” and q to represent “You can log onto the network.” Then, the argument has the form

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

where \therefore is the symbol that denotes “therefore.”



04 Logic and Proofs

4.4 Rules of Inference

DEFINITION 1

An *argument* in propositional logic is a sequence of propositions. All but the final proposition in the argument are called *premises* and the final proposition is called the *conclusion*. An argument is *valid* if the truth of all its premises implies that the conclusion is true.

An *argument form* in propositional logic is a sequence of compound propositions involving propositional variables. An argument form is *valid* no matter which particular propositions are substituted for the propositional variables in its premises, the conclusion is true if the premises are all true.

The tautology $(p \wedge (p \rightarrow q)) \rightarrow q$ is the basis of the rule of inference called **modus ponens**, or the **law of detachment**. (Modus ponens is Latin for *mode that affirms*.) This tautology leads to the following valid argument form, which we have already seen in our initial discussion about arguments (where, as before, the symbol \therefore denotes “therefore”):

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

Using this notation, the hypotheses are written in a column, followed by a horizontal bar, followed by a line that begins with the therefore symbol and ends with the conclusion. In particular, modus ponens tells us that if a conditional statement and the hypothesis of this conditional statement are both true, then the conclusion must also be true. Example 1 illustrates the use of modus ponens.



04 Logic and Proofs

4.4 Rules of Inference

TABLE 1 Rules of Inference.

Rule of Inference	Tautology	Name
$\begin{array}{c} p \\ p \rightarrow q \\ \hline \therefore q \end{array}$	$(p \wedge (p \rightarrow q)) \rightarrow q$	Modus ponens
$\begin{array}{c} \neg q \\ p \rightarrow q \\ \hline \therefore \neg p \end{array}$	$(\neg q \wedge (p \rightarrow q)) \rightarrow \neg p$	Modus tollens
$\begin{array}{c} p \rightarrow q \\ q \rightarrow r \\ \hline \therefore p \rightarrow r \end{array}$	$((p \rightarrow q) \wedge (q \rightarrow r)) \rightarrow (p \rightarrow r)$	Hypothetical syllogism
$\begin{array}{c} p \vee q \\ \neg p \\ \hline \therefore q \end{array}$	$((p \vee q) \wedge \neg p) \rightarrow q$	Disjunctive syllogism
$\begin{array}{c} p \\ \hline \therefore p \vee q \end{array}$	$p \rightarrow (p \vee q)$	Addition
$\begin{array}{c} p \wedge q \\ \hline \therefore p \end{array}$	$(p \wedge q) \rightarrow p$	Simplification
$\begin{array}{c} p \\ q \\ \hline \therefore p \wedge q \end{array}$	$((p) \wedge (q)) \rightarrow (p \wedge q)$	Conjunction
$\begin{array}{c} p \vee q \\ \neg p \vee r \\ \hline \therefore q \vee r \end{array}$	$((p \vee q) \wedge (\neg p \vee r)) \rightarrow (q \vee r)$	Resolution



04 Logic and Proofs

4.4 Rules of Inference

- Using Rules of Inference to Build Arguments

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



04 Logic and Proofs

4.4 Rules of Inference

● Rules of Inference for Quantified Statements

Universal instantiation is the rule of inference used to conclude that $P(c)$ is true, where c is a particular member of the domain, given the premise $\forall x P(x)$. Universal instantiation is used when we conclude from the statement “All women are wise” that “Lisa is wise,” where Lisa is a member of the domain of all women.

Universal generalization is the rule of inference that states that $\forall x P(x)$ is true, given the premise that $P(c)$ is true for all elements c in the domain. Universal generalization is used when we show that $\forall x P(x)$ is true by taking an arbitrary element c from the domain and showing that $P(c)$ is true. The element c that we select must be an arbitrary, and not a specific, element of the domain. That is, when we assert from $\forall x P(x)$ the existence of an element c in the domain, we have no control over c and cannot make any other assumptions about c other than it comes from the domain. Universal generalization is used implicitly in many proofs in mathematics and is seldom mentioned explicitly. However, the error of adding unwarranted assumptions about the arbitrary element c when universal generalization is used is all too common in incorrect reasoning.

Existential instantiation is the rule that allows us to conclude that there is an element c in the domain for which $P(c)$ is true if we know that $\exists x P(x)$ is true. We cannot select an arbitrary value of c here, but rather it must be a c for which $P(c)$ is true. Usually we have no knowledge

of what c is, only that it exists. Because it exists, we may give it a name (c) and continue our argument.

Existential generalization is the rule of inference that is used to conclude that $\exists x P(x)$ is true when a particular element c with $P(c)$ true is known. That is, if we know one element c in the domain for which $P(c)$ is true, then we know that $\exists x P(x)$ is true.



04 Logic and Proofs

4.4 Rules of Inference

- Rules of Inference for Quantified Statements

TABLE 2 Rules of Inference for Quantified Statements.

<i>Rule of Inference</i>	<i>Name</i>
$\frac{\forall x P(x)}{\therefore P(c)}$	Universal instantiation
$\frac{P(c) \text{ for an arbitrary } c}{\therefore \forall x P(x)}$	Universal generalization
$\frac{\exists x P(x)}{\therefore P(c) \text{ for some element } c}$	Existential instantiation
$\frac{P(c) \text{ for some element } c}{\therefore \exists x P(x)}$	Existential generalization



04 Logic and Proofs

4.5 Methods of Proof

Methods:

1. Direct Proofs
2. Proof by Contraposition
3. Proofs by Contradiction



04 Logic and Proofs

4.5 Methods of Proof

● Direct Proofs

DEFINITION 1

The integer n is *even* if there exists an integer k such that $n = 2k$, and n is *odd* if there exists an integer k such that $n = 2k + 1$. (Note that every integer is either even or odd, and no integer is both even and odd.) Two integers have the *same parity* when both are even or both are odd; they have *opposite parity* when one is even and the other is odd.

EXAMPLE 1 Give a direct proof of the theorem “If n is an odd integer, then n^2 is odd.”

Solution: Note that this theorem states $\forall n P((n) \rightarrow Q(n))$, where $P(n)$ is “ n is an odd integer” and $Q(n)$ is “ n^2 is odd.” As we have said, we will follow the usual convention in mathematical proofs by showing that $P(n)$ implies $Q(n)$, and not explicitly using universal instantiation. To begin a direct proof of this theorem, we assume that the hypothesis of this conditional statement is true, namely, we assume that n is odd. By the definition of an odd integer, it follows that $n = 2k + 1$, where k is some integer. We want to show that n^2 is also odd. We can square both sides of the equation $n = 2k + 1$ to obtain a new equation that expresses n^2 . When we do this, we find that $n^2 = (2k + 1)^2 = 4k^2 + 4k + 1 = 2(2k^2 + 2k) + 1$. By the definition of an odd integer, we can conclude that n^2 is an odd integer (it is one more than twice an integer). Consequently, we have proved that if n is an odd integer, then n^2 is an odd integer. 

Extra Examples



04 Logic and Proofs

4.5 Methods of Proof

- Proof by Contraposition

Direct proofs lead from the premises of a theorem to the conclusion. They begin with the premises, continue with a sequence of deductions, and end with the conclusion. However, we will see that attempts at direct proofs often reach dead ends. We need other methods of proving theorems of the form $\forall x(P(x) \rightarrow Q(x))$. Proofs of theorems of this type that are not direct proofs, that is, that do not start with the premises and end with the conclusion, are called **indirect proofs**.

An extremely useful type of indirect proof is known as **proof by contraposition**. Proofs by contraposition make use of the fact that the conditional statement $p \rightarrow q$ is equivalent to its contrapositive, $\neg q \rightarrow \neg p$. This means that the conditional statement $p \rightarrow q$ can be proved by showing that its contrapositive, $\neg q \rightarrow \neg p$, is true. In a proof by contraposition of $p \rightarrow q$, 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. We will illustrate proof by contraposition with two examples. These examples show that proof by contraposition can succeed when we



04 Logic and Proofs

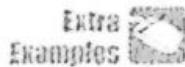
4.5 Methods of Proof

● Proof by Contraposition

cannot easily find a direct proof.

EXAMPLE 3 Prove that if n is an integer and $3n + 2$ is odd, then n is odd.

Solution: We first attempt a direct proof. To construct a direct proof, we first assume that $3n + 2$ is an odd integer. This means that $3n + 2 = 2k + 1$ for some integer k . Can we use this fact to show that n is odd? We see that $3n + 1 = 2k$, but there does not seem to be any direct way to conclude that n is odd. Because our attempt at a direct proof failed, we next try a proof by contraposition.



The first step in a proof by contraposition is to assume that the conclusion of the conditional statement “If $3n + 2$ is odd, then n is odd” is false; namely, assume that n is even. Then, by the definition of an even integer, $n = 2k$ for some integer k . Substituting $2k$ for n , we find that $3n + 2 = 3(2k) + 2 = 6k + 2 = 2(3k + 1)$. This tells us that $3n + 2$ is even (because it is a multiple of 2), and therefore not odd. This is the negation of the premise of the theorem. Because the negation of the conclusion of the conditional statement implies that the hypothesis is false, the original conditional statement is true. Our proof by contraposition succeeded; we have proved the theorem “If $3n + 2$ is odd, then n is odd.”



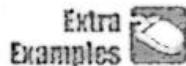
04 Logic and Proofs

4.5 Methods of Proof

● Proof by Contraposition

DEFINITION 2 The real number r is *rational* if there exist integers p and q with $q \neq 0$ such that $r = p/q$. A real number that is not rational is called *irrational*.

EXAMPLE 7 Prove that the sum of two rational numbers is rational. (Note that if we include the implicit quantifiers here, the theorem we want to prove is “For every real number r and every real number s , if r and s are rational numbers, then $r + s$ is rational.”)



Solution: We first attempt a direct proof. To begin, suppose that r and s are rational numbers. From the definition of a rational number, it follows that there are integers p and q , with $q \neq 0$, such that $r = p/q$, and integers t and u , with $u \neq 0$, such that $s = t/u$. Can we use this information to show that $r + s$ is rational? The obvious next step is to add $r = p/q$ and $s = t/u$, to obtain

$$r + s = \frac{p}{q} + \frac{t}{u} = \frac{pu + qt}{qu}.$$

Because $q \neq 0$ and $u \neq 0$, it follows that $qu \neq 0$. Consequently, we have expressed $r + s$ as the ratio of two integers, $pu + qt$ and qu , where $qu \neq 0$. This means that $r + s$ is rational. We have proved that the sum of two rational numbers is rational; our attempt to find a direct proof succeeded.



04 Logic and Proofs

4.5 Methods of Proof

● Proofs by Contradiction

Suppose we want to prove that a statement p is true. Furthermore, suppose that we can find a contradiction q such that $\neg p \rightarrow q$ is true. Because q is false, but $\neg p \rightarrow q$ is true, we can conclude that $\neg p$ is false, which means that p is true. How can we find a contradiction q that might help us prove that p is true in this way?

Because the statement $r \wedge \neg r$ is a contradiction whenever r is a proposition, 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 . Proofs of this type are called **proofs by contradiction**. Because a proof by contradiction does not prove a result directly, it is another type of indirect proof. We provide three examples of proof by contradiction. The first is an example of an application of the pigeonhole principle, a combinatorial technique that we will cover in depth in Section 3.2.



04 Logic and Proofs

4.5 Methods of Proof

● Proofs by Contradiction

EXAMPLE 11 Give a proof by contradiction of the theorem “If $3n + 2$ is odd, then n is odd.”

Solution: Let p be “ $3n + 2$ is odd” and q be “ n is odd.” To construct a proof by contradiction, assume that both p and $\neg q$ are true. That is, assume that $3n + 2$ is odd and that n is not odd. Because n is not odd, we know that it is even. Because n is even, there is an integer k such that $n = 2k$. This implies that $3n + 2 = 3(2k) + 2 = 6k + 2 = 2(3k + 1)$. Because $3n + 2$ is $2t$, where $t = 3k + 1$, $3n + 2$ is even. Note that the statement “ $3n + 2$ is even” is equivalent to the statement $\neg p$, because an integer is even if and only if it is not odd. Because both p and $\neg p$ are true, we have a contradiction. This completes the proof by contradiction, proving that if $3n + 2$ is odd, then n is odd. ◀

Note that we can also prove by contradiction that $p \rightarrow q$ is true by assuming that p and $\neg q$ are true, and showing that q must be also be true. This implies that $\neg q$ and q are both true, a contradiction. This observation tells us that we can turn a direct proof into a proof by contradiction.

PROOFS OF EQUIVALENCE To prove a theorem that is a biconditional statement, that is, a statement of the form $p \leftrightarrow q$, we show that $p \rightarrow q$ and $q \rightarrow p$ are both true. The validity of this approach is based on the tautology

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



04 Logic and Proofs

4.5 Methods of Proof

Strategy:

- 1. Exhaustive Proof and Proof by Cases**
- 2. Existence Proofs**
- 3. Uniqueness Proofs**
- 4. Looking for Counterexample**



04 Logic and Proofs

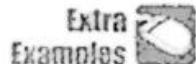
4.5 Methods of Proof

- **Exhaustive Proof and Proof by Cases**

EXHAUSTIVE PROOF Some theorems can be proved by examining a relatively small number of examples. Such proofs are called **exhaustive proofs**, or **proofs by exhaustion** because these proofs proceed by exhausting all possibilities. An exhaustive proof is a special type of proof by cases where each case involves checking a single example. We now provide some illustrations of exhaustive proofs.

EXAMPLE 1 Prove that $(n + 1)^3 \geq 3^n$ if n is a positive integer with $n \leq 4$.

Solution: We use a proof by exhaustion. We only need verify the inequality $(n + 1)^3 \geq 3^n$ when $n = 1, 2, 3$, and 4 . For $n = 1$, we have $(n + 1)^3 = 2^3 = 8$ and $3^n = 3^1 = 3$; for $n = 2$, we have $(n + 1)^3 = 3^3 = 27$ and $3^n = 3^2 = 9$; for $n = 3$, we have $(n + 1)^3 = 4^3 = 64$ and $3^n = 3^3 = 27$; and for $n = 4$, we have $(n + 1)^3 = 5^3 = 125$ and $3^n = 3^4 = 81$. In each of these four cases, we see that $(n + 1)^3 \geq 3^n$. We have used the method of exhaustion to prove that $(n + 1)^3 \geq 3^n$ if n is a positive integer with $n \leq 4$. 



04 Logic and Proofs

4.5 Methods of Proof

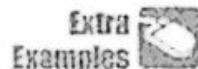
- **Exhaustive Proof and Proof by Cases**

PROOF BY CASES A proof by cases must cover all possible cases that arise in a theorem. We illustrate proof by cases with a couple of examples. In each example, you should check that all possible cases are covered.

EXAMPLE 3 Prove that if n is an integer, then $n^2 \geq n$.

Solution: We can prove that $n^2 \geq n$ for every integer by considering three cases, when $n = 0$, when $n \geq 1$, and when $n \leq -1$. We split the proof into three cases because it is straightforward to prove the result by considering zero, positive integers, and negative integers separately.

Case (i): When $n = 0$, because $0^2 = 0$, we see that $0^2 \geq 0$. It follows that $n^2 \geq n$ is true in this case.



Case (ii): When $n \geq 1$, when we multiply both sides of the inequality $n \geq 1$ by the positive integer n , we obtain $n \cdot n \geq n \cdot 1$. This implies that $n^2 \geq n$ for $n \geq 1$.

Case (iii): In this case $n \leq -1$. However, $n^2 \geq 0$. It follows that $n^2 \geq n$.

Because the inequality $n^2 \geq n$ holds in all three cases, we can conclude that if n is an integer, then $n^2 \geq n$. 



04 Logic and Proofs

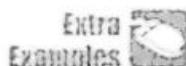
4.5 Methods of Proof

● Existence Proofs

Many theorems are assertions that objects of a particular type exist. A theorem of this type is a proposition of the form $\exists x P(x)$, where P is a predicate. A proof of a proposition of the form $\exists x P(x)$ is called an **existence proof**. There are several ways to prove a theorem of this type. Sometimes an existence proof of $\exists x P(x)$ can be given by finding an element a , called a **witness**, such that $P(a)$ is true. This type of existence proof is called **constructive**. It is also possible to give an existence proof that is **nonconstructive**; that is, we do not find an element a such that $P(a)$ is true, but rather prove that $\exists x P(x)$ is true in some other way. One common method of giving a nonconstructive existence proof is to use proof by contradiction and show that the negation of the existential quantification implies a contradiction. The concept of a constructive existence proof is illustrated by Example 10 and the concept of a nonconstructive existence proof is illustrated by Example 11.

EXAMPLE 10

A Constructive Existence Proof Show that there is a positive integer that can be written as the sum of cubes of positive integers in two different ways.



Solution: After considerable computation (such as a computer search) we find that

$$1729 = 10^3 + 9^3 = 12^3 + 1^3.$$

Because we have displayed a positive integer that can be written as the sum of cubes in two different ways, we are done.



04 Logic and Proofs

4.5 Methods of Proof

● Uniqueness Proofs

Some theorems assert the existence of a unique element with a particular property. In other words, these theorems assert that there is exactly one element with this property. To prove a statement of this type we need to show that an element with this property exists and that no other element has this property. The two parts of a **uniqueness proof** are:

Existence: We show that an element x with the desired property exists.

Uniqueness: We show that if $y \neq x$, then y does not have the desired property.

Equivalently, we can show that if x and y both have the desired property, then $x = y$.

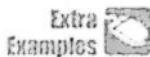
Remark: Showing that there is a unique element x such that $P(x)$ is the same as proving the statement $\exists x(P(x) \wedge \forall y(y \neq x \rightarrow \neg P(y)))$.

We illustrate the elements of a uniqueness proof in Example 13.

EXAMPLE 13 Show that if a and b are real numbers and $a \neq 0$, then there is a unique real number r such that $ar + b = 0$.

Solution: First, note that the real number $r = -b/a$ is a solution of $ar + b = 0$ because $a(-b/a) + b = -b + b = 0$. Consequently, a real number r exists for which $ar + b = 0$. This is the existence part of the proof.

Second, suppose that s is a real number such that $as + b = 0$. Then $ar + b = as + b$, where $r = -b/a$. Subtracting b from both sides, we find that $ar = as$. Dividing both sides of this last equation by a , which is nonzero, we see that $r = s$. This means that if $s \neq r$, then $as + b \neq 0$. This establishes the uniqueness part of the proof. 



04 Logic and Proofs

4.5 Methods of Proof

● Looking for Counterexample

EXAMPLE 17 In Example 14 in Section 1.7 we showed that the statement “Every positive integer is the sum of two squares of integers” is false by finding a counterexample. That is, there are positive integers that cannot be written as the sum of the squares of two integers. Although we cannot write every positive integer as the sum of the squares of two integers, maybe we can write every positive integer as the sum of the squares of three integers. That is, is the statement “Every positive integer is the sum of the squares of three integers” true or false?

Extra Examples 

Solution: Because we know that not every positive integer can be written as the sum of two squares of integers, we might initially be skeptical that every positive integer can be written as the sum of three squares of integers. So, we first look for a counterexample. That is, we can show that the statement “Every positive integer is the sum of three squares of integers” is false if we can find a particular integer that is not the sum of the squares of three integers. To look for a counterexample, we try to write successive positive integers as a sum of three squares. We find that $1 = 0^2 + 0^2 + 1^2$, $2 = 0^2 + 1^2 + 1^2$, $3 = 1^2 + 1^2 + 1^2$, $4 = 0^2 + 0^2 + 2^2$, $5 = 0^2 + 1^2 + 2^2$, $6 = 1^2 + 1^2 + 2^2$, but we cannot find a way to write 7 as the sum of three squares. To show that there are not three squares that add up to 7, we note that the only possible squares we can use are those not exceeding 7, namely, 0, 1, and 4. Because no three terms where each term is 0, 1, or 4 add up to 7, it follows that 7 is a counterexample. We conclude that the statement “Every positive integer is the sum of the squares of three integers” is false.

We have shown that not every positive integer is the sum of the squares of three integers. The next question to ask is whether every positive integer is the sum of the squares of four positive integers. Some experimentation provides evidence that the answer is yes. For example, $7 = 1^2 + 1^2 + 1^2 + 2^2$, $25 = 4^2 + 2^2 + 2^2 + 1^2$, and $87 = 9^2 + 2^2 + 1^2 + 1^2$. It turns out the conjecture “Every positive integer is the sum of the squares of four integers” is true. For a proof, see [Ro10]. 



04 Logic and Proofs

4.6 Summary

Key Terms and Results

TERMS

proposition: a statement that is true or false

propositional variable: a variable that represents a proposition

truth value: true or false

$\neg p$ (**negation of p :**) the proposition with truth value opposite to the truth value of p

logical operators: operators used to combine propositions

compound proposition: a proposition constructed by combining propositions using logical operators

truth table: a table displaying all possible truth values of propositions

$p \vee q$ (**disjunction of p and q :**) the proposition “ p or q ,” which is true if and only if at least one of p and q is true

$p \wedge q$ (**conjunction of p and q :**) the proposition “ p and q ,” which is true if and only if both p and q are true

$p \oplus q$ (**exclusive or of p and q :**) the proposition “ p XOR q ,” which is true when exactly one of p and q is true

$p \rightarrow q$ (**p implies q :**) the proposition “if p , then q ,” which is false if and only if p is true and q is false

converse of $p \rightarrow q$: the conditional statement $q \rightarrow p$

contrapositive of $p \rightarrow q$: the conditional statement $\neg q \rightarrow \neg p$

inverse of $p \rightarrow q$: the conditional statement $\neg p \rightarrow \neg q$

$p \leftrightarrow q$ (**biconditional:**) the proposition “ p if and only if q ,” which is true if and only if p and q have the same truth value

bit: either a 0 or a 1

ables is assigned a value or is bound by a quantifier

domain (or universe) of discourse: the values a variable in a propositional function may take

$\exists x P(x)$ (**existential quantification of $P(x)$:**) the proposition that is true if and only if there exists an x in the domain such that $P(x)$ is true

$\forall x P(x)$ (**universal quantification of $P(x)$:**) the proposition that is true if and only if $P(x)$ is true for every x in the domain

logically equivalent expressions: expressions that have the same truth value no matter which propositional functions and domains are used

free variable: a variable not bound in a propositional function

bound variable: a variable that is quantified

scope of a quantifier: portion of a statement where the quantifier binds its variable

argument: a sequence of statements

argument form: a sequence of compound propositions involving propositional variables

premise: a statement, in an argument, or argument form, other than the final one

conclusion: the final statement in an argument or argument form

valid argument form: a sequence of compound propositions involving propositional variables where the truth of all the premises implies the truth of the conclusion

valid argument: an argument with a valid argument form

rule of inference: a valid argument form that can be used in the demonstration that arguments are valid



04 Logic and Proofs

4.6 Summary

Boolean variable: a variable that has a value of 0 or 1

bit operation: an operation on a bit or bits

bit string: a list of bits

bitwise operations: operations on bit strings that operate on each bit in one string and the corresponding bit in the other string

logic gate: a logic element that performs a logical operation on one or more bits to produce an output bit

logic circuit: a switching circuit made up of logic gates that produces one or more output bits

tautology: a compound proposition that is always true

contradiction: a compound proposition that is always false

contingency: a compound proposition that is sometimes true and sometimes false

consistent compound propositions: compound propositions for which there is an assignment of truth values to the variables that makes all these propositions true

satisfiable compound proposition: a compound proposition for which there is an assignment of truth values to its variables that makes it true

logically equivalent compound propositions: compound propositions that always have the same truth values

predicate: part of a sentence that attributes a property to the subject

propositional function: a statement containing one or more variables that becomes a proposition when each of its vari-

fallacy: an invalid argument form often used incorrectly as a rule of inference (or sometimes, more generally, an incorrect argument)

circular reasoning or begging the question: reasoning where one or more steps are based on the truth of the statement being proved

theorem: a mathematical assertion that can be shown to be true

conjecture: a mathematical assertion proposed to be true, but that has not been proved

proof: a demonstration that a theorem is true

axiom: a statement that is assumed to be true and that can be used as a basis for proving theorems

lemma: a theorem used to prove other theorems

corollary: a proposition that can be proved as a consequence of a theorem that has just been proved

vacuous proof: a proof that $p \rightarrow q$ is true based on the fact that p is false

trivial proof: a proof that $p \rightarrow q$ is true based on the fact that q is true

direct proof: a proof that $p \rightarrow q$ is true that proceeds by showing that q must be true when p is true

proof by contraposition: a proof that $p \rightarrow q$ is true that proceeds by showing that p must be false when q is false

proof by contradiction: a proof that p is true based on the



04 Logic and Proofs

4.6 Summary

truth of the conditional statement $\neg p \rightarrow q$, where q is a contradiction

exhaustive proof: a proof that establishes a result by checking a list of all possible cases

proof by cases: a proof broken into separate cases, where these cases cover all possibilities

without loss of generality: an assumption in a proof that makes it possible to prove a theorem by reducing the number of cases to consider in the proof

counterexample: an element x such that $P(x)$ is false

constructive existence proof: a proof that an element with a specified property exists that explicitly finds such an element

nonconstructive existence proof: a proof that an element with a specified property exists that does not explicitly find such an element

rational number: a number that can be expressed as the ratio of two integers p and q such that $q \neq 0$

uniqueness proof: a proof that there is exactly one element satisfying a specified property

RESULTS

The logical equivalences given in Tables 6, 7, and 8 in Section 1.3.

De Morgan's laws for quantifiers.

Rules of inference for propositional calculus.

Rules of inference for quantified statements.



Thanks !

