UNIVERSITY OF THE CORDILLERAS
College of Information Technology and Computer Science

MODULE in Discrete Structures

COURSE	: CC9
COURSE TITLE	: Discrete Structures
COURSE CREDITS	: 3 units
CONTACT HOURS/WEEK	: 5 hours per week
PREREQUISITE	: CC4 – Data Structures and Algorithms

COURSE DESCRIPTION:
This course covers the mathematical topics most directly related in the study of various Computer Science areas and emphasizes on providing a context for the application of the mathematics within computer science. Topics include logic fundamentals, basic set theory, relations and functions, proof techniques, and mathematical induction.

COURSE OUTCOMES:
At the end of the course, students are expected to:

1. Demonstrate understanding in
    a. mathematical reasoning in order to read, comprehend and construct mathematical arguments;
    b. critical and logical reasoning and thinking.
2. Identify and logically analyze problems and issues, and to propose and evaluate solutions.
3. Apply appropriate mathematics procedures and quantitative methods.
4. Perform and explain basic computational procedures and quantitative analysis.
5. Demonstrate awareness of the inter-disciplinary nature of knowledge, particularly the relevance and practicality of Discrete Mathematics in the field of Computer Science.

UNIT 1

Introduction to Logic

OBJECTIVES

At the end of unit 1, students are expected to:

1. convert English sentences in logical form and vice-versa;
2. use logical inference rules and equivalence laws to simplify logical expressions;
3. justify the validity of an argument by the use of truth table;
4. use inference rules and equivalence laws to prove the validity of arguments; and
5. Use inference rules and equivalence laws to prove the validity of arguments.

MATERIALS/RESOURCES

Browser
Text Editor / Word Processor

LESSON

Introduction to Logic

A. What is Discrete Mathematics?

- Discrete Mathematics is the part of mathematics devoted to the study of discrete objects. The kind of mathematics that arises when an understanding is sought of how computers operate in terms of hardware and software, involves organization and symbol and data manipulation; the modern computer is essentially a finite discrete system.
- Areas in which Discrete Mathematics concepts are present: Formal Languages (computer languages), Compiler Design, Data Structures, Computability, Automata Theory, Algorithm Design, Algorithm Design, Relational Database Theory, Complexity Theory (Counting)

B. Logic

- Logic is the study of reasoning. It is specifically concerned with whether reasoning is correct. The rules of logic give precise meaning to mathematical statements. These rules are used to distinguish between valid and invalid mathematical arguments.
- The uses of logic: it verifies the correctness of algorithms, the rules of logic are used in the design of computer circuits and the development of computer programs, it has practical applications to the design of computing machines, to the specification of systems, to artificial intelligence, to computer programming, to program languages, and other areas of computer science.

Proposition

- A proposition is a declarative sentence that is either true or flase, but not both.
- Example 1: The following are declarative sentences and are propositions.
  1. The only positive integers that divide 7 are 1 and 7 itself.
  2. 21 is a prime number.
  3. $2 + 2 = 5$
  4. $12 \div 4 = 3$

     - Propositions 1 and 4 are true, 2 and 3 are false.

- Example 2: Consider the following sentences.
  1. Read this statement carefully.
  2. What time is it?
  3. $x + 1 = 5$
  4. $x + y = z$

     - Sentences 1 and 2 are not propositions because they are not declarative sentences, sentences 3 and 4 are not propositions because they are neither true nor false.

- Propositions are the basic building bocks of any theory of logic.
- Propositional Calculus/Logic – the area of logic that deals with proposition.
- The bases for propositional logic are the three laws of Aristotelian logic.
  1. Law of Identity. “A thing is itself”.
  2. Law of Excluded Middle. A statement is either true or false but not both.
  3. Law of Non-Contradiction. No statement is both true and false.
- Simple propositions
- Compound propositions results from combining propositions using logical operators.
- Propositions are denoted by letters. The conventional letters used are p, q, r, s, t…..

Logical Operators

- Logical connectives used to form new propositions from existing ones.
- Defintion:
  - **Conjunction**. Let p and q be propositions. The conjunction of p and q denoted by $p \wedge q$ is the proposition p and q.

    Example: p: 2 is an even number.
    q: 5 is a prime number.
    $p \wedge q$: 2 is an even number and 5 is a prime number.

- **Disjunction**. Let p and q be propositions. The disjunction of p and q denoted by $p \vee q$ is the proposition p or q.

  Example: p: It is raining.
  q: It is cold
  $p \vee q$: It is raining or it is cold

- **Negation.** Let p be a proposition. The negation of p, denoted by $\sim p (\neg p)$ is the proposition, not p.

  Example: p: Paris is the capital of England.
  $\sim p$: It is not the case that Paris is the capital of England.
  Or simply, $\sim p$: Paris is not the capital of England.

- **Exclusive OR (xor).** 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.
  - it is used when two cases happen at the same time.

  Example: p; You can have a glass of milk
  q: You can have a glass of orange juice for breakfast.
  $p \oplus q$: Either you can have a glass of milk or a glass of orange juice for breakfast.
  In this example, you cannot have both milk and orange juice for breakfast.

- **Implication:** If p and q are propositions, then the compound proposition, "if p, then q" is called implication/conditional proposition and is denoted by $p \rightarrow q$.
  p is the hypothesis/premise/antecedent
  q is the conclusion/consequence.

  Other ways to express conditional statement:

" if p, then q"	"p implies q"
" if p, q"	" p only if q"
" p is sufficient for q"	"a sufficient condition for q is p"
"q if p"	"q whenever p"
"q when p"	"q is necessary for p"
"q follows from p"	" a necessary condtion for p is q"

  Examples:
  1. If all men are mortal, then Mario is mortal. (Logical implication)
  2. If this polygon is a quadrilateral, then it has four sides. (Definitional implication)

3. If today is Tuesday, then tomorrow is Wednesday. (Sequential implication)

4. If sugar is placed in water, then it will dissolve. (Causal implication)

Inverse, Converse, Contra-positive

Def: given an implication $p \rightarrow q$;
a. the inverse is $\sim p \rightarrow \sim q$
b. the converse is $q \rightarrow p$
c. the contra-positive is $\sim q \rightarrow \sim p$.

Ex: Find the inverse, converse, and contra-positive of the implication:

p: The United States of America goes to war.
q: The price of crude oil goes up.
$p \rightarrow q$: If the United States of America goes to war, then the price of crude oil goes up.

Ans:
$\sim p \rightarrow \sim q$: *If the United States of America does not go to war, then the price of crude oil does not go up.*

$q \rightarrow p$: *If the price of crude oil goes up, then the United States of America goes to war.*

$\sim q \rightarrow \sim p$: *If the price of crude oil does not go up, then the United States of America does not go to war.*

Which among the inverse, converse and contra-postive is of exactly the same meaning as the original implication?

- **Bi-conditional Proposition.** Def: Let p and q be propositions. The bi-conditional proposition of p and q denoted by $p \leftrightarrow q$, read as " p if and only if q", is the proposition that is true when p and q have the same truth values, and is false otherwise.

  p iff q
  p implies q and q implies p

Example: p: The curve is a circle.
q: The curve is equidistant to a point.

$p \leftrightarrow q$: The curve is a circle if and only if the curve is equidistant to a point.

PRACTICE EXERCISE

A. Determine whether the sentence is a proposition. If the sentence is a proposition, write its negation).
1. 2 + 5 = 19
2. Waiter, will you please serve the nuts?
3. Today is Monday.
4. The difference of two primes.
5. Peel me a grape.

B. Formulate the symbolic expression in words using
p: Today is Monday.
q: It is raining.
r: It is cold.
1. $p \vee q$
2. $\sim p \wedge (q \vee r)$
3. $p \rightarrow (q \vee r)$
4. $q \leftrightarrow r$

C. Represent the propositions symbolically if:
p: 1 + 1 = 0
q: Roses are red.
r: Violets are blue.
s: The instructor is pretty.
1. Either 1 + 1 = 0 or roses are red.
2. If roses are red, then violets are blue.
3. It is not the case that the instructor is pretty.
4. If 1 + 1 = 0, then the instructor is pretty or violets are blue.
5. 1 + 1 = 0 if and only if violets are blue.

GRADED ACTIVITY

Activities will be given in In-person classes

REFERENCES

1. Rosen, Kenneth D (2007). Discrete mathematics and its applications. 6th ed. Boston : McGraw Hill Higher Education.
2. Introduction to Logic. Taken on June 15, 2020 from http://intrologic.stanford.edu/notes/notes.html
3. https://www.math.fsu.edu/~pkirby/mad2104/SlideShow/s2_1.pdf

ANSWERS TO THE PRACTICE EXERCISE

A. Determine whether the sentence is a proposition. If the sentence is a proposition, write its negation).

1. $2 + 5 = 19$
   Ans: Proposition: It is not the case that 2 + 5 = 19.
2. Waiter, will you please serve the nuts?
   Ans: Not a proposition
3. Today is Monday.
   Ans: Proposition: Today is not Monday.
4. The difference of two primes.
   Ans: Not a proposition.
5. Peel me a grape.
   Ans: Not a proposition.

B. Formulate the symbolic expression in words using
p: Today is Monday.
q: It is raining.
r: It is cold.

1. $p \vee q$
   ans: Today is Monday or it is raining.
2. $\sim p \wedge (q \vee r)$
   ans: Today is not Monday and it is raining or it is cold.
3. $p \rightarrow (q \vee r)$
   ans: If today is Monday, then it is raining or it is cold.
4. $q \leftrightarrow r$
   ans: It is raining if and only if it is cold.

C. Represent the propositions symbolically if:
p: $1 + 1 = 0$
q: Roses are red.

r: Violets are blue.
s: The instructor is pretty.

1. Either 1 + 1 = 0 or roses are red.
   Ans: $p \vee q$
2. If roses are red, then violets are blue.
   Ans: $q \rightarrow r$
3. It is not the case that the instructor is pretty.
   Ans: $\sim s$
4. If 1 + 1 = 0, then the instructor is pretty or violets are blue.
   Ans: $p \rightarrow (s \vee r)$
5. 1 + 1 = 0 if and only if violets are blue.
   Ans: $p \leftrightarrow r$

Prepared by:

Melinda A. Beninsig
Instructor

CHAPTER 2

Logic

1. Logic Definitions

1.1. Propositions.

Definition 1.1.1. *A* **proposition** *is a declarative sentence that is either*

true (denoted either T or 1) or

false (denoted either F or 0).

Notation: Variables are used to represent propositions. The most common variables used are p, q, and r.

Discussion

Logic has been studied since the classical Greek period (600-300BC). The Greeks, most notably Thales, were the first to formally analyze the reasoning process. Aristotle (384-322BC), the "father of logic", and many other Greeks searched for universal truths that were irrefutable. A second great period for logic came with the use of symbols to simplify complicated logical arguments. Gottfried Leibniz (1646-1716) began this work at age 14, but failed to provide a workable foundation for symbolic logic. George Boole (1815-1864) is considered the "father of symbolic logic". He developed logic as an abstract mathematical system consisting of defined terms (propositions), operations (conjunction, disjunction, and negation), and rules for using the operations. It is this system that we will study in the first section.

Boole's basic idea was that if simple propositions could be represented by precise symbols, the relation between the propositions could be read as precisely as an algebraic equation. Boole developed an "algebra of logic" in which certain types of reasoning were reduced to manipulations of symbols.

1.2. Examples.

Example 1.2.1. *"Drilling for oil caused dinosaurs to become extinct." is a proposition.*

Example 1.2.2. *"Look out!" is not a proposition.*

Example 1.2.3. *"How far is it to the next town?" is not a proposition.*

Example 1.2.4. *"$x + 2 = 2x$" is not a proposition.*

Example 1.2.5. *"$x + 2 = 2x$ when $x = -2$" is a proposition.*

Recall a *proposition* is a declarative sentence that is either true or false. Here are some further examples of propositions:

Example 1.2.6. *All cows are brown.*

Example 1.2.7. *The Earth is further from the sun than Venus.*

Example 1.2.8. *There is life on Mars.*

Example 1.2.9. $2 \times 2 = 5$.

Here are some sentences that are not propositions.

Example 1.2.10. *"Do you want to go to the movies?" Since a question is not a declarative sentence, it fails to be a proposition.*

Example 1.2.11. *"Clean up your room." Likewise, an imperative is not a declarative sentence; hence, fails to be a proposition.*

Example 1.2.12. *"$2x = 2 + x$." This is a declarative sentence, but unless x is assigned a value or is otherwise prescribed, the sentence neither true nor false, hence, not a proposition.*

Example 1.2.13. *"This sentence is false." What happens if you assume this statement is true? false? This example is called a paradox and is not a proposition, because it is neither true nor false.*

Each proposition can be assigned one of two *truth values*. We use T or 1 for true and use F or 0 for false.

1.3. Logical Operators.

Definition 1.3.1. *Unary Operator* **negation:** *"not p", $\neg p$.*

Definitions 1.3.1. *Binary Operators*

(a) **conjunction:** *"p and q", $p \wedge q$.*
(b) **disjunction:** *"p or q", $p \vee q$.*
(c) **exclusive or:** *"exactly one of p or q", "p xor q", $p \oplus q$.*
(d) **implication:** *"if p then q", $p \to q$.*
(e) **biconditional:** *"p if and only if q", $p \leftrightarrow q$.*

Discussion

A sentence like "I can jump and skip" can be thought of as a combination of the two sentences "I can jump" and "I can skip." When we analyze arguments or logical expression it is very helpful to break a sentence down to some composition of simpler statements.

We can create *compound propositions* using propositional variables, such as $p, q, r, s, \ldots$, and *connectives* or *logical operators*. A logical operator is either a *unary* operator, meaning it is applied to only a single proposition; or a *binary* operator, meaning it is applied to two propositions. *Truth tables* are used to exhibit the relationship between the truth values of a compound proposition and the truth values of its component propositions.

1.4. Negation. Negation Operator, "not", has symbol $\neg$.

EXAMPLE 1.4.1. *p: This book is interesting.*

$\neg p$ can be read as:

(i.) This book is not interesting.
(ii.) This book is uninteresting.
(iii.) It is not the case that this book is interesting.

Truth Table:

p	$\neg p$
T	F
F	T

Discussion

The *negation* operator is a unary operator which, when applied to a proposition p, changes the truth value of p. That is, the negation of a proposition p, denoted by $\neg p$, is the proposition that is false when p is true and true when p is false. For example, if p is the statement "I understand this", then its negation would be "I do not understand this" or "It is not the case that I understand this." Another notation commonly used for the negation of p is $\sim p$.

Generally, an appropriately inserted "not" or removed "not" is sufficient to negate a simple statement. Negating a compound statement may be a bit more complicated as we will see later on.

1.5. Conjunction. **Conjunction Operator**, "and", has symbol $\wedge$.

EXAMPLE 1.5.1. *p: This book is interesting. q: I am staying at home.*

$p \wedge q$: *This book is interesting, and I am staying at home.*

Truth Table:

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

Discussion

The *conjunction* operator is the binary operator which, when applied to two propositions p and q, yields the proposition "p and q", denoted $p \wedge q$. The conjunction $p \wedge q$ of p and q is the proposition that is true when both p and q are true and false otherwise.

1.6. Disjunction. **Disjunction Operator**, inclusive "or", has symbol $\vee$.

EXAMPLE 1.6.1. *p: This book is interesting. q: I am staying at home.*

$p \vee q$: *This book is interesting, or I am staying at home.*

Truth Table:

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

Discussion

The *disjunction* operator is the binary operator which, when applied to two propositions p and q, yields the proposition "p or q", denoted $p \vee q$. The disjunction $p \vee q$ of p and q is the proposition that is true when either p is true, q is true, or *both* are true, and is false otherwise. Thus, the "or" intended here is the *inclusive or*. In fact, the symbol $\vee$ is the abbreviation of the Latin word *vel* for the inclusive "or".

1.7. Exclusive Or. Exclusive Or Operator, "xor", has symbol $\oplus$.

EXAMPLE 1.7.1. *p: This book is interesting. q: I am staying at home.*

$p \oplus q$: *Either this book is interesting, or I am staying at home, but not both.*

Truth Table:

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

Discussion

The *exclusive or* is the binary operator which, when applied to two propositions p and q yields the proposition "p xor q", denoted $p \oplus q$, which is true if exactly one of p or q is true, but not both. It is false if both are true or if both are false.

Many times in our every day language we use "or" in the exclusive sense. In logic, however, we always mean the inclusive or when we simply use "or" as a connective in a proposition. If we mean the exclusive or it must be specified. For example, in a restaurant a menu may say there is a choice of soup or salad with a meal. In logic this would mean that a customer may choose both a soup and salad with their meal. The logical implication of this statement, however, is probably not what is intended. To create a sentence that logically states the intent the menu could say that there is a choice of *either* soup or salad (but not both). The phrase "either . . . or . . . " is normally indicates the exclusive or.

1.8. Implications. Implication Operator, "if...then...", has symbol $\to$.

EXAMPLE 1.8.1. *p: This book is interesting. q: I am staying at home.*

$p \to q$: *If this book is interesting, then I am staying at home.*

Truth Table:

p	q	$p \to q$
T	T	T
T	F	F
F	T	T
F	F	T

Equivalent Forms of "If p then q":

- p implies q
- If p, q
- p only if q
- p is a sufficient condition for q
- q if p
- q whenever p
- q is a necessary condition for p

Discussion

The *implication* $p \to q$ is the proposition that is often read "if p then q." "If p then q" is false precisely when p is true but q is false. There are many ways to say this connective in English. You should study the various forms as shown above.

One way to think of the meaning of $p \to q$ is to consider it a contract that says if the first condition is satisfied, then the second will also be satisfied. If the first condition, p, is not satisfied, then the condition of the contract is null and void. In this case, it does not matter if the second condition is satisfied or not, the contract is still upheld.

For example, suppose your friend tells you that if you meet her for lunch, she will give you a book she wants you to read. According to this statement, you would expect her to give you a book if you do go to meet her for lunch. But what if you do not meet her for lunch? She did not say anything about that possible situation, so she would not be breaking any kind of promise if she dropped the book off at your house that night or if she just decided not to give you the book at all. If either of these last two possibilities happens, we would still say the implication stated was true because she did not break her promise.

EXERCISE 1.8.1. *Which of the following statements are equivalent to "If x is even, then y is odd"? There may be more than one or none.*

(1) y is odd only if x is even.
(2) x is even is sufficient for y to be odd.
(3) x is even is necessary for y to be odd.
(4) If x is odd, then y is even.
(5) x is even and y is even.
(6) x is odd or y is odd.

1.9. Terminology. For the compound statement $p \to q$

- p is called the **premise, hypothesis,** or the **antecedent**.
- q is called the **conclusion** or **consequent**.
- $q \to p$ is the **converse** of $p \to q$.

- $\neg p \to \neg q$ is the **inverse** of $p \to q$.
- $\neg q \to \neg p$ is the **contrapositive** of $p \to q$.

Discussion

We will see later that the converse and the inverse are not equivalent to the original implication, but the contrapositive $\neg q \to \neg p$ is. In other words, $p \to q$ and its contrapositive have the exact same truth values.

1.10. Example.

EXAMPLE 1.10.1. *Implication: If this book is interesting, then I am staying at home.*

- **Converse:** *If I am staying at home, then this book is interesting.*
- **Inverse:** *If this book is not interesting, then I am not staying at home.*
- **Contrapositive:** *If I am not staying at home, then this book is not interesting.*

Discussion

The converse of your friend's promise given above would be "if she gives you a book she wants you to read, then you will meet her for lunch," and the inverse would be "If you do not meet her for lunch, then she will not give you the book." We can see from the discussion about this statement that neither of these are the same as the original promise. The contrapositive of the statement is "if she does not give you the book, then you do not meet her for lunch." This is, in fact, equivalent to the original promise. Think about when would this promise be broken. It should be the exact same situation where the original promise is broken.

EXERCISE 1.10.1. *p is the statement "I will prove this by cases", q is the statement "There are more than 500 cases," and r is the statement "I can find another way."*

1. *State $(\neg r \vee \neg q) \to p$ in simple English.*
2. *State the converse of the statement in part 1 in simple English.*
3. *State the inverse of the statement in part 1 in simple English.*
4. *State the contrapositive of the statement in part 1 in simple English.*

1.11. Biconditional. Biconditional Operator, "if and only if", has symbol $\leftrightarrow$.

EXAMPLE 1.11.1. *p: This book is interesting. q: I am staying at home.*

$p \leftrightarrow q$: This book is interesting if and only if I am staying at home.

Truth Table:

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

Discussion

The biconditional statement is equivalent to $(p \to q) \wedge (q \to p)$. In other words, for $p \leftrightarrow q$ to be true we must have both p and q true or both false. The difference between the implication and biconditional operators can often be confusing, because in our every day language we sometimes say an "if...then" statement, $p \to q$, when we actually mean the *biconditional* statement $p \leftrightarrow q$. Consider the statement you may have heard from your mother (or may have said to your children): "If you eat your broccoli, then you may have some ice cream." Following the strict logical meaning of the first statement, the child still may or may not have ice cream even if the broccoli isn't eaten. The "if...then" construction does not indicate what would happen in the case when the hypothesis is not true. The intent of this statement, however, is most likely that the child *must* eat the broccoli in order to get the ice cream.

When we set out to prove a biconditional statement, we often break the proof down into two parts. First we prove the implication $p \to q$, and then we prove the converse $q \to p$.

Another type of "if...then" statement you may have already encountered is the one used in computer languages. In this "if...then" statement, the premise is a condition to be tested, and if it is true then the conclusion is a procedure that will be performed. If the premise is not true, then the procedure will not be performed. Notice this is different from "if...then" in logic. It is actually closer to the biconditional in logic. However, it is not actually a logical statement at all since the "conclusion" is really a list of commands, not a proposition.

1.12. NAND and NOR Operators.

Definition 1.12.1. *The* **NAND Operator***, which has symbol* $|$ *("Sheffer Stroke"), is defined by the truth table*

p	q	$p\|q$
T	T	F
T	F	T
F	T	T
F	F	T

DEFINITION 1.12.2. *The* **NOR Operator***, which has symbol* $\downarrow$ *("Peirce Arrow"), is defined by the truth table*

p	q	$p \downarrow q$
T	T	F
T	F	F
F	T	F
F	F	T

Discussion

These two additional operators are very useful as logical gates in a combinatorial circuit, a topic we will discuss later.

1.13. Example.

EXAMPLE 1.13.1. *Write the following statement symbolically, and then make a truth table for the statement. "If I go to the mall or go to the movies, then I will not go to the gym."*

Solution. *Suppose we set*

- $p =$ *I go to the mall*
- $q =$ *I go to the movies*
- $r =$ *I will go to the gym*

The proposition can then be expressed as "If p *or* q*, then not* r*," or* $(p \vee q) \to \neg r$*.*

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

Discussion

When building a truth table for a compound proposition, you need a row for every possible combination of T's and F's for the component propositions. Notice if there is only one proposition involved, there are 2 rows. If there are two propositions, there are 4 rows, if there are 3 propositions there are 8 rows.

EXERCISE 1.13.1. *How many rows should a truth table have for a statement involving n different propositions?*

It is not always so clear cut how many columns one needs. If we have only three propositions p, q, and r, you would, in theory, only need four columns: one for each of p, q, and r, and one for the compound proposition under discussion, which is $(p \vee q) \to \neg r$ in this example. In practice, however, you will probably want to have a column for each of the successive intermediate propositions used to build the final one. In this example it is convenient to have a column for $p \vee q$ and a column for $\neg r$, so that the truth value in each row in the column for $(p \vee q) \to \neg r$ is easily supplied from the truth values for $p \vee q$ and $\neg r$ in that row.

Another reason why you should show the intermediate columns in your truth table is for grading purposes. If you make an error in a truth table and do not give this extra information, it will be difficult to evaluate your error and give you partial credit.

EXAMPLE 1.13.2. *Suppose p is the proposition "the apple is delicious" and q is the proposition "I ate the apple." Notice the difference between the two statements below.*

(a) $\neg p \wedge q$ = The apple is not delicious, and I ate the apple.
(b) $\neg(p \wedge q)$ = It is not the case that: the apple is delicious and I ate the apple.

EXERCISE 1.13.2. *Find another way to express Example 1.13.2 Part b without using the phrase "It is not the case."*

EXAMPLE 1.13.3. *Express the proposition "If you work hard and do not get distracted, then you can finish the job" symbolically as a compound proposition in terms of simple propositions and logical operators.*

Set

- p = you work hard
- q = you get distracted
- r = you can finish the job

In terms of p, q, and r, the given proposition can be written

$$(p \wedge \neg q) \to r.$$

The comma in Example 1.13.3 is not necessary to distinguish the order of the operators, but consider the sentence "If the fish is cooked then dinner is ready and I am hungry." Should this sentence be interpreted as $f \to (r \wedge h)$ or $(f \to r) \wedge h$, where f, r, and h are the natural choices for the simple propositions? A comma needs to be inserted in this sentence to make the meaning clear or rearranging the sentence could make the meaning clear.

EXERCISE 1.13.3. *Insert a comma into the sentence "If the fish is cooked then dinner is ready and I am hungry." to make the sentence mean*

(a) $f \to (r \wedge h)$
(b) $(f \to r) \wedge h$

EXAMPLE 1.13.4. *Here we build a truth table for* $p \to (q \to r)$ *and* $(p \wedge q) \to r$. *When creating a table for more than one proposition, we may simply add the necessary columns to a single truth table.*

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

EXERCISE 1.13.4. *Build one truth table for* $f \to (r \wedge h)$ *and* $(f \to r) \wedge h$.

1.14. Bit Strings.

DEFINITION 1.14.1. *A* **bit** *is a 0 or a 1 and a* **bit string** *is a list or string of bits.*

The logical operators can be turned into **bit operators** by thinking of 0 as false and 1 as true. The obvious substitutions then give the table

$\overline{0} = 1$	$\overline{1} = 0$	
$0 \vee 0 = 0$	$0 \wedge 0 = 0$	$0 \oplus 0 = 0$
$0 \vee 1 = 1$	$0 \wedge 1 = 0$	$0 \oplus 1 = 1$
$1 \vee 0 = 1$	$1 \wedge 0 = 0$	$1 \oplus 0 = 1$
$1 \vee 1 = 1$	$1 \wedge 1 = 1$	$1 \oplus 1 = 0$

Discussion

We can define the *bitwise NEGATION* of a string and *bitwise OR, bitwise AND,* and *bitwise XOR* of two bit strings of the same length by applying the logical operators to the corresponding bits in the natural way.

EXAMPLE 1.14.1.

(a) $\overline{11010} = 00101$

(b) $11010 \vee 10001 = 11011$

(c) $11010 \wedge 10001 = 10000$

(d) $11010 \oplus 10001 = 01011$

UNIVERSITY OF THE CORDILLERAS
College of Information Technology and Computer Science

MODULE in Discrete Structures

COURSE	: CC9
COURSE TITLE	: Discrete Structures
COURSE CREDITS	: 3 units
CONTACT HOURS/WEEK	: 5 hours per week
PREREQUISITE	: CC4 – Data Structures and Algorithms

COURSE DESCRIPTION:
This course covers the mathematical topics most directly related in the study of various Computer Science areas and emphasizes on providing a context for the application of the mathematics within computer science. Topics include logic fundamentals, basic set theory, relations and functions, proof techniques, and mathematical induction.

COURSE OUTCOMES:
At the end of the course, students are expected to:

1. Demonstrate understanding in
    a. mathematical reasoning in order to read, comprehend and construct mathematical arguments;
    b. critical and logical reasoning and thinking.
2. Identify and logically analyze problems and issues, and to propose and evaluate solutions.
3. Apply appropriate mathematics procedures and quantitative methods.
4. Perform and explain basic computational procedures and quantitative analysis.
5. Demonstrate awareness of the inter-disciplinary nature of knowledge, particularly the relevance and practicality of Discrete Mathematics in the field of Computer Science.

UNIT 1

Introduction to Logic
Truth Tables
Precedence of Logical Operators
Logic and Bit Operations

OBJECTIVES

At the end of unit 1, students are expected to:
1. represent the given proposition symbolically; and determine the truth value of each sentence.
2. determine the truth value of each sentence;
3. construct the truth table for each compound proposition; and
4. find the bitwise OR, bitwise AND and bitwise XOR of each pair of bit strings.

MATERIALS/RESOURCES

Browser
Text Editor / Word Processor

LESSON

Introduction to Logic

A. The Truth Table

- Truth table –The truth table of a proposition p made up of the individual propositions $p_1 \ldots p_n$ lists all possible combinations of truth values for $p_1 \ldots p_n$. T denoting true and F denoting false, and for each combination lists the truth value of p.
  - table that lists all possible truth values of a sentence based on its components.

Conjunction ($p \wedge q$)

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

- A conjunction is true if all components is true otherwise it is false.

This table has a row for each of the four possible combinations of the truth values of p and q. The four rows correspond to the pairs of truth values TT, TF, FT and FF, where the first truth value in the pair is the truth value of p and the second truth value is the truth value of q.

Disjunction ($p \vee q$)

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

- A disjunction is true if atleast one component is true.

Negation (~ p)

p	~ p
T	F
F	T

This table has a row for each of the two possible truth values of a proposition p. Each row shows the truth value of ~ 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. This operator is a *unary operator.*

Exclusive OR ($p \oplus q$)

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

- It is used when two cases happens at the same time.

When the *exclusive or* is used to connect the propositions p and q, the proposition "p or q" (but not both) is obtained. This proposition is true when p is true and q is false, and when p is false and q is true. It is false when both p and q are false and when both are true.

Implication ($p \rightarrow q$)

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

- False when the premise is true but the consequence is false.
- A conditional proposition that is true because the hypothesis is false is said to be true by default or vacuously true.

Bi-condtional Prposition ($p \leftrightarrow q$)

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

- True whenever the variable have the same truth values.

The statement $p \leftrightarrow q$ is true when both the conditional statements $p \rightarrow q$ and $q \rightarrow p$ are true (or both are false) and is false otherwise. That is why we use the words 'if and only if" to express the logical connective and why it is symbolically written by combining the symbols $\rightarrow$ and $\leftarrow$.

B. Precedence of Logical Operators

Operator precedence is an ordering of logical operator designed to allow the reduction of the number of parenthesis in logical expressions. The table gives the precedence of logical operator.

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

- The $\sim$ operator has higher precedence than $\wedge$; $\wedge$ has higher precedence than $\vee$; and $\vee$ has higher precedence than $\rightarrow$ and $\leftrightarrow$.

C. Logic and Bit Operations

Computers represent information using bits. A bit is a symbol with two possible values, namely, 0 (zero) and 1 (one). This meaning of the word bit comes from binary digit, because zeros and ones are the digits used in binary representations of numbers. A bit can be used to represent a truth value, because there are two values, namely, true and false. 1 represents T (true), 0 represents F (false).

Computer bit operations use logical connectives namely $\wedge$, $\vee$, $\oplus$. Information is often represented using bit strings, which are lists of zeros and ones. When this is done, operations on the bit strings can be used to manipulate this information.

A bit string is a sequence of zero or more bits. The length of this string is the number of bits in the string.

Truth table for bit operators OR, AND , and XOR

x	y	$x \vee y$	$x \wedge y$	$x \oplus y$
1	1	1	1	0
1	0	1	0	1
0	1	1	0	1
0	0	0	0	0

Examples:

Represent the given propositions symbolically by letting:

$p: 5 < 9$
$q: 9 < 7$
$r: 5 < 7$

then determine whether each proposition is true or false.

1. $5 < 9$ and $9 < 7$.
2. It is not the case that ($5 < 9$ and $9 < 7$).
3. If $5 < 9$, then $9 < 7$.
4. $5 < 9$ if and only if $9 < 7$ or $5 < 7$.

Solution: $p: 5 < 9$ is T
$q: 9 < 7$ is F
$r: 5 < 7$ is T

1. $p \wedge q$
   $T \wedge F$
   $\boxed{F}$

2. $\sim (p \wedge q)$
   $\sim (T \wedge F)$
   $\sim F$
   $\boxed{T}$

3. $p \rightarrow q$
   $T \rightarrow F$
   $\boxed{F}$

4. $p \leftrightarrow q \vee r$
   $T \leftrightarrow F \vee T$
   $T \leftrightarrow T$
   $\boxed{T}$

Find the truth value of the ff. if
p and s are true, q and r are false

1. $\sim p$
2. $p \rightarrow r$
3. $r \rightarrow (q \wedge \sim s)$
4. $(q \oplus r) \leftrightarrow (p \oplus s)$

Solution:

1. $\sim p$
   $\sim T$
   $\boxed{F}$
2. $p \rightarrow r$
   $T \rightarrow F$
   $\boxed{F}$

3. $r \rightarrow (q \wedge \sim s)$

$F \rightarrow (F \wedge \sim T)$

$F \rightarrow (F \wedge F)$

$F \rightarrow F$

$\boxed{T}$

4. $(q \oplus r) \leftrightarrow (p \oplus s)$

   $(F \oplus F) \leftrightarrow (T \oplus T)$

   $F \leftrightarrow F$

   $\boxed{T}$

Write the truth table of the following propositions.

1. $p \wedge \sim q$

p	**q**	**~q**	**p ∧ ~q**
T	T	F	F
T	F	T	T
F	T	F	F
F	F	T	F

2. $p \oplus (p \vee q)$

p	**q**	**p ∨ q**	**p ⊕ (p ∨ q)**
T	T	T	F
T	F	T	F
F	T	T	T
F	F	F	F

3. $\sim (p \wedge q) \rightarrow (r \wedge \sim p)$

p	**q**	**r**	**p ∧ q**	**~ (p ∧q)**	**~ p**	**r ∧ ~ p**	**~ (p ∧ q) → (r ∧ ~ p)**
T	T	T	T	F	F	F	T
T	T	F	T	F	F	F	T
T	F	T	F	T	F	F	F
T	F	F	F	T	F	F	F
F	T	T	F	T	T	T	T
F	T	F	F	T	T	F	F
F	F	T	F	T	T	T	T
F	F	F	F	T	T	F	F

Represent the given proposition symbolically and generate its truth table.

If John doesn't pass then he will lose his scholarship and drop out of school.
Solution: Let p: John pass.
$\sim p$: John doesn't pass.
q: He will lose his scholarship.
r: He will drop out of school.
In symbols: $\sim p \rightarrow (q \wedge r)$
Truth Table:

p	**q**	**r**	**$\sim p$**	**$q \wedge r$**	**$\sim p \rightarrow (q \wedge r)$**
T	T	T	F	T	T
T	T	F	F	F	T
T	F	T	F	F	T
T	F	F	F	F	T
F	T	T	T	T	T
F	T	F	T	F	F
F	F	T	T	F	F
F	F	F	T	F	F

Find the bitwise OR, bitwise AND, and bitwise XOR of the bit strings 01 1011 0110 and 11 0001 1101. (In here, bit strings will be split into blocks of four bits to make them easier to read)

Solution:
01 1011 0110
11 0001 1101
11 1011 1111 bitwise OR
01 0001 0100 bitwise AND
10 1010 1011 bitwise XOR

PRACTICE EXERCISE

A. Represent the given propositions symbolically by letting:
p: $5 < 9$
q: $9 < 7$
r: $5 < 7$
then determine whether each proposition is true or false.
1. $9 < 7$ or $5 < 7$
2. If $5 < 9$, then ($9 < 7$ and $5 < 7$)
3. $9 < 7$ if and only if ($5 < 9$ or $9 < 7$).
4. $5 < 9$ or it is not the case that ($9 < 7$ and $5 < 7$)

B. Find the truth value of the ff. if
p and s are true, q and r are false
1. $p \vee \sim q$
2. $p \rightarrow r$
3. $r \rightarrow (q \wedge \sim s)$

4. $\sim (p \wedge q) \rightarrow (r \vee s)$

C. Write the truth table of the following propositions.

1. $p \vee \sim q$
2. $(p \wedge q) \vee (\sim p \vee q)$
3. $(p \wedge q) \vee r$
4. $(\sim p \leftrightarrow q) \rightarrow (\sim q \leftrightarrow r)$

D. Represent the given proposition symbolically and generate its truth table.

1. The sun is hot but it is not humid.
2. If your car won't start or you don't wake up on time then you will miss your interview and you will not get the new job.

E. Find the bitwise OR, bitwise AND and bitwise XOR of the following strings.

1. 101 1110 0000
   010 0011 1001

GRADED ACTIVITY

A separate activity will be given (posted).

REFERENCES

1. Rosen, Kenneth D (2007). Discrete mathematics and its applications. 6th ed. Boston : McGraw Hill Higher Education.
2. Introduction to Logic. Taken on June 15, 2020 from http://intrologic.stanford.edu/notes/notes.html
3. https://www.math.fsu.edu/~pkirby/mad2104/SlideShow/s2_1.pdf
4. https://www.youtube.com/watch?v=UiGu57JzLkE&t=61s
5. https://www.youtube.com/watch?v=wRMC-ttjhwM
6. https://www.youtube.com/watch?v=8octtUkdv4Y

ANSWERS TO SELECTED PRACTICE EXERCISE

A. Represent the given propositions symbolically by letting:

p: 5 < 9
q: 9 < 7
r: 5 < 7

then determine whether each proposition is true or false.

1. 9 < 7 or 5 < 7

Solution: p: 5 < 9 is T; q: 9 < 7 is F; r: 5 < 7 is T

1. $q \vee r$

$F \vee T$

$\boxed{T}$

B. Find the truth value of the ff. if

p and s are true, q and r are false

1. $p \vee \sim q$

Solution:

1. $p \vee \sim q$

   $T \vee \sim F$

   $T \vee T$

   $\boxed{T}$

C. Write the truth table of the following propositions.

2. $(p \wedge q) \vee (\sim p \vee q)$

p	**q**	**$p \wedge q$**	**$\sim p$**	**$\sim p \vee q$**	**$(p \wedge q) \vee (\sim p \vee q)$**
T	T	T	F	T	T
T	F	F	F	F	F
F	T	F	T	T	T
F	F	F	T	T	T

D. Represent the given proposition symbolically and generate its truth table.

1. The sun is hot but it is not humid.

Solution: Let p: The sun is hot.

q: It is humid.

$\sim$ q: It is not humid.

Symbols: $p \wedge \sim q$

Truth Table:

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

E. Find the bitwise OR, bitwise AND and bitwise XOR of the following strings.

1. 101 1110 0000
   010 0011 1001
   111 1111 1001 bitwise OR
   000 0010 0000 bitwise AND
   111 1101 1001 bitwise XOR

Prepared by:

Melinda A. Beninsig
Intructor

UNIVERSITY OF THE CORDILLERAS
College of Information Technology and Computer Science

MODULE in Discrete Structures

COURSE	: CC9
COURSE TITLE	: Discrete Structures
COURSE CREDITS	: 3 units
CONTACT HOURS/WEEK	: 5 hours per week
PREREQUISITE	: CC4 – Data Structures and Algorithms

COURSE DESCRIPTION:
This course covers the mathematical topics most directly related in the study of various Computer Science areas and emphasizes on providing a context for the application of the mathematics within computer science. Topics include logic fundamentals, basic set theory, relations and functions, proof techniques, and mathematical induction.

COURSE OUTCOMES:
At the end of the course, students are expected to:
1. Demonstrate understanding in
    a. mathematical reasoning in order to read, comprehend and construct mathematical arguments;
    b. critical and logical reasoning and thinking.
2. Identify and logically analyze problems and issues, and to propose and evaluate solutions.
3. Apply appropriate mathematics procedures and quantitative methods.
4. Perform and explain basic computational procedures and quantitative analysis.
5. Demonstrate awareness of the inter-disciplinary nature of knowledge, particularly the relevance and practicality of Discrete Mathematics in the field of Computer Science.

UNIT 1

Introduction to Logic

Propositional Equivalences

Logical Equivalences

Logical Equivalence Laws

OBJECTIVES

At the end of unit 1, students are expected to:

1. differentiate tautology, contradiction and contingency;
2. show that a compound proposition is a tautology, contradiction and contingency using truth table;
3. use the truth table to show compound propositions are logically equivalent;
4. explain the different laws of equivalence; and
5. show that a statement is logically equivalent using the laws of equivalences.

MATERIALS/RESOURCES

Browser

Text Editor / Word Processor

LESSON

Propositional Equivalence

A. Types of Compound Propositions

1. Tautology (Valid) – It is a compound proposition that is always true no matter what the truth values that occur in it.
   - It is true for all interpretation

   Example: $p \vee \sim p$ is a tautology

2. Contradiction (Absurdity) – it a compound proposition that is always false.
   - It is false for all interpretation.

   Example: $p \wedge \sim p$ is a contradiction

3. Contingency (Satisfiable) – It is a compound proposition that is neither a tautology nor a contradiction.

   Remarks: *We can show tautology, contradiction and contingency using truth table.*

Examples: Use truth tables to determine if the proposition is a tautology, contingency or contradiction.

1. $[(p \to q) \wedge \sim q] \leftrightarrow \sim (p \vee q)$

p	**q**	**$p \to q$**	**$\sim q$**	**$(p \to q) \wedge \sim q$**	**$p \vee q$**	**$\sim (p \vee q)$**	**$[(p \to q) \wedge \sim q] \leftrightarrow \sim (p \vee q)$**
T	T	T	F	F	T	F	T
T	F	F	T	F	T	F	T
F	T	T	F	F	T	F	T
F	F	T	T	T	F	T	T

$\therefore$ Tautology

2. $(p \vee q) \wedge \sim (\sim p \to q)$

p	**q**	**$p \vee q$**	**$\sim p$**	**$\sim p \to q$**	**$\sim(\sim p \to q)$**	**$(p \vee q) \wedge \sim (\sim p \to q)$**
T	T	T	F	T	F	F
T	F	T	F	T	F	F
F	T	T	T	T	F	F
F	F	F	T	F	T	F

$\therefore$ Contradiction

B. Logical Equivalence

Compound propositions that have the same truth values in all possible cases are called logically equivalent.

Definition: The proposition p and q are called logically equivalent if $p \leftrightarrow q$ is a tautology. The notation $p \equiv q$ denotes p and q are logically equivalent.

Remarks: 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 sometime used instead of $\equiv$ to denote logical equivalence.

Examples: Use truth table to show that the ff are logically equivalent.

1. $p \to q \equiv \sim q \to \sim p$

p	**q**	**$p \to q$**	**$\sim q$**	**$\sim p$**	**$\sim q \to \sim p$**
T	T	T	F	F	T
T	F	F	T	F	F
F	T	T	F	T	T
F	F	T	T	T	T

$\therefore$ Logically equivalent since $p \to q$ have exactly the same truth values with $\sim q \to \sim p$

2. $p \leftrightarrow q \equiv (p \wedge q) \vee (\sim p \wedge \sim q)$

p	**q**	**$p \leftrightarrow q$**	**$p \wedge q$**	**$\sim p$**	**$\sim q$**	**$\sim p \wedge \sim q$**	**$(p \wedge q) \vee (\sim p \wedge \sim q)$**
T	T	T	T	F	F	F	T
T	F	F	F	F	T	F	F
F	T	F	F	T	F	F	F
F	F	T	F	T	T	T	T

∴ Logically equivalent

3. $(p \rightarrow r) \vee (q \rightarrow r) \equiv (p \wedge q) \rightarrow r$

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

∴Logically equivalent

C. Logical Equivalence Laws

The table shows the important equivalences. In these equivalences, **T** denotes the compound proposition that is always true and **F** denotes the compound proposition that is always false.

Name of Equivalences	**Equivalences**
Reflexive Law	$p \equiv p$
Identity Law	$p \wedge \mathbf{T} \equiv p$ $p \vee \mathbf{F} \equiv p$
Domination Law	$p \vee \mathbf{T} \equiv \mathbf{T}$ $p \wedge \mathbf{F} \equiv \mathbf{F}$
Idempotent Law	$p \vee p \equiv p$ $p \wedge p \equiv p$
Double Negation	$\sim(\sim p) \equiv p$
Commutative Law	$p \vee q \equiv q \vee p$ $p \wedge q \equiv q \wedge p$
Associative Law	$(p \vee q) \vee r \equiv p \vee (q \vee r)$ $(p \wedge q) \wedge r \equiv p \wedge (q \wedge r)$
Distributive Law	$p \vee (q \wedge r) \equiv (p \vee q) \wedge (p \vee r)$

	$p \wedge (q \vee r) \equiv (p \wedge q) \vee (p \wedge r)$
De Morgan's Law	$\sim (p \wedge q) \equiv \sim p \vee \sim q$ $\sim (p \vee q) \equiv \sim p \wedge \sim q$
Absorption Law	$p \vee (p \wedge q) \equiv p$ $p \wedge (p \vee q) \equiv p$
Negation Law	$p \vee \sim p \equiv T$ $p \wedge \sim p \equiv F$
Contra-positive	$p \rightarrow q \equiv \sim q \rightarrow \sim p$
Implication	$p \rightarrow q \equiv \sim p \vee q$
Equivalence	$p \leftrightarrow q \equiv (p \rightarrow q) \wedge (q \rightarrow p)$
Exportation	$(p \wedge q) \rightarrow r \equiv p \rightarrow (q \rightarrow r)$

Examples: Show that the ff are equivalent. Use equivalence laws .

1. **$(p \vee q) \wedge \sim (\sim p \wedge q) \equiv p$**

   Solution: Simplify the left side of the equivalence until it will be transformed to the right side.

$(p \vee q) \wedge \sim (\sim p \wedge q) \equiv p$
$(p \vee q) \wedge (\sim \sim p \vee \sim q) \equiv p$ De Morgan's Law
$(p \vee q) \wedge (p \vee \sim q) \equiv p$ Double Negation Law
$p \vee (q \wedge \sim q) \equiv p$ Distributive Law
$p \vee F \equiv p$ Negation Law
$p \equiv p$ Identity Law
$\therefore$Logically equivalent

2. **$p \rightarrow (q \wedge r) \equiv (p \rightarrow q) \wedge (p \rightarrow r)$**

   Solution: Simplify the left side

$p \rightarrow (q \wedge r) \equiv (p \rightarrow q) \wedge (p \rightarrow r)$
$\sim p \sim (q \wedge r) \equiv (p \rightarrow q) \wedge (p \rightarrow r)$ Implication Law
$(\sim p \vee q) \wedge (\sim p \vee r) \equiv (p \rightarrow q) \wedge (p \rightarrow r)$ Distributive Law (p
$\rightarrow q) \wedge (p \rightarrow r) \equiv (p \rightarrow q) \wedge (p \rightarrow r)$ Implication Law
$\therefore$Logically equivalent

or simplify the right side
$p \rightarrow (q \wedge r) \equiv (p \rightarrow q) \wedge (p \rightarrow r)$
$p \rightarrow (q \wedge r) \equiv (\sim p \vee q) \wedge (\sim p \vee r)$ Implication Law
$p \rightarrow (q \wedge r) \equiv \sim p \vee (q \wedge r)$ Distributive Law
$p \rightarrow (q \wedge r) \equiv p \rightarrow (q \wedge r)$ Implication Law
$\therefore$Logically equivalent

3. $\mathbf{(p \wedge q) \rightarrow p \equiv T}$

Solution: Simplify the left side

$(p \wedge q) \rightarrow p \equiv T$	
$\sim (p \wedge q) \vee p \equiv T$	Implication Law
$(\sim p \vee \sim q) \vee p \equiv T$	De Morgan's Law
$p \vee (\sim p \vee \sim q) \equiv T$	Commutative Law
$(p \vee \sim p) \vee \sim q \equiv T$	Associative Law
$T \vee \sim q \equiv T$	Negation Law
$T \equiv T$	Domination Law

$\therefore$ Logically equivalent

PRACTICE EXERCISE

A. Use truth tables to determine if the proposition is a tautology, contingency or contradiction:

1. $\{[(\sim p \vee q) \rightarrow r]\} \rightarrow \sim (p \rightarrow q)$
2. $[(p \rightarrow q) \wedge (q \rightarrow r)] \rightarrow (p \rightarrow r)$
3. $[\sim p \wedge (p \rightarrow q)] \rightarrow \sim q$

B. Use truth table to show that the ff are logically equivalent.

1. $(p \vee q) \wedge \sim (\sim p \wedge q) \equiv p$
2. $p \wedge (q \vee r) \equiv (p \wedge q) \vee (p \wedge r)$
3. $(\sim p \wedge q) \wedge (q \rightarrow p) \equiv F$

C. Show that the ff are equivalent. Use equivalence laws .

1. $\sim [p \vee (\sim p \wedge q)] \equiv \sim p \wedge \sim q$
2. $(p \vee q) \rightarrow r \equiv (p \rightarrow r) \wedge (q \rightarrow r)$
3. $p \rightarrow (q \vee r) \equiv \sim r \rightarrow (p \rightarrow q)$
4. $\sim [p \vee \sim(p \wedge q)] \equiv F$
5. $(p \wedge q) \rightarrow (p \vee q) \equiv T$

D. 1. Use truth tables to verify the associative laws:

a. $(p \vee q) \vee r \equiv p \vee (q \vee r)$

b. $(p \wedge q) \wedge r \equiv p \wedge (q \wedge r)$

2. Use a truth table to verify the distributive law

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

3. Use a truth table to verify the first De Morgan Law $\sim(p \wedge q) \equiv \sim p \vee \sim q$

GRADED ACTIVITY

A separate activity will be given (posted).

REFERENCES

1. Rosen, Kenneth D (2007). Discrete mathematics and its applications. 6th ed. Boston : McGraw Hill Higher Education.
2. Introduction to Logic. Taken on June 15, 2020 from http://intrologic.stanford.edu/notes/notes.html
3. https://math.la.asu.edu/~boerner/mat243/1.3%20Propositional%20Equivalence%20(expanded).pdf
4. https://www.cs.sfu.ca/~ggbaker/zju/math/equiv.html
5. https://www.youtube.com/watch?v=tDH67yRNXzI
6. https://www.youtube.com/watch?v=x0ELYvaf21E

ANSWERS TO SELECTED PRACTICE EXERCISE

A.Use truth tables to determine if the proposition is a tautology, contingency or contradiction:

3. . $[\sim p \wedge (p \rightarrow q)] \rightarrow \sim q$

p	**q**	**$\sim p$**	**$p \rightarrow q$**	**$[\sim p \wedge (p \rightarrow q)]$**	**$\sim q$**	**$[\sim p \wedge (p \rightarrow q)] \rightarrow \sim q$**
T	T	F	T	F	F	T
T	F	F	F	F	T	T
F	T	T	T	T	F	F
F	F	T	T	T	T	T

$\therefore$Contingency

B. Use truth table to show that the ff are logically equivalent.

1. $(p \vee q) \wedge \sim(\sim p \wedge q) \equiv p$

p	**q**	**$p \vee q$**	**$\sim p$**	**$\sim p \wedge q$**	**$\sim(\sim p \wedge q)$**	**$(p \vee q) \wedge \sim(\sim p \wedge q)$**
T	T	T	F	F	T	T
T	F	T	F	F	T	T
F	T	T	T	T	F	F
F	F	F	T	F	T	F

$\therefore$ Logically equivalent

C. Show that the ff are equivalent. Use equivalence laws .

1. $\sim [p \vee (\sim p \wedge q)] \equiv \sim p \wedge \sim q$

   Solution: Simplify the left side

$\sim [p \vee (\sim p \wedge q)] \equiv \sim p \wedge \sim q$	
$\sim p \wedge \sim (\sim p \wedge q) \equiv \sim p \wedge \sim q$	De Morgan's Law
$\sim p \wedge (\sim \sim p \vee \sim q) \equiv \sim p \wedge \sim q$	De Morgan's Law
$\sim p \wedge (p \vee \sim q) \equiv \sim p \wedge \sim q$	Double Negation
$(\sim p \wedge p) \vee (\sim p \wedge \sim q) \equiv \sim p \wedge \sim q$	Distributive Law
$F \vee (\sim p \wedge \sim q) \equiv \sim p \wedge \sim q$	Negation Law
$\sim p \wedge \sim q \equiv \sim p \wedge \sim q$	Identity Law

$\therefore$ Logically equivalent

D. 3. Use a truth table to verify the first De Morgan Law

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

p	**q**	**$p \wedge q$**	**$\sim (p \wedge q)$**	**$\sim p$**	**$\sim q$**	**$\sim p \vee \sim q$**
T	T	T	**F**	F	F	**F**
T	F	F	**T**	F	T	**T**
F	T	F	**T**	T	F	**T**
F	F	F	**T**	T	T	**T**

$\therefore$ Verified

UNIVERSITY of the CORDILLERAS
BAGUIO CITY

COURSE : CC9

COURSE TITLE : Discrete Structures

COURSE CREDITS : 3 units

CONTACT HOURS/WEEK : 5 hours per week

PREREQUISITE : CC4 – Data Structures and Algorithms

COURSE DESCRIPTION:
This course covers the mathematical topics most directly related in the study of various Computer Science areas and emphasizes on providing a context for the application of the mathematics within computer science. Topics include logic fundamentals, basic set theory, relations and functions, proof techniques, and mathematical induction.

COURSE OUTCOMES:
At the end of the course, students are expected to:

1. Demonstrate understanding in
   a. mathematical reasoning in order to read, comprehend and construct mathematical arguments;
   b. critical and logical reasoning and thinking.
2. Identify and logically analyze problems and issues, and to propose and evaluate solutions.
3. Apply appropriate mathematics procedures and quantitative methods.
4. Perform and explain basic computational procedures and quantitative analysis.
5. Demonstrate awareness of the inter-disciplinary nature of knowledge, particularly the relevance and practicality of Discrete Mathematics in the field of Computer Science.

Prepared by:
Myleen A. Beninsig

UNIT 1

Introduction to Logic
Rules of Inference

OBJECTIVES

At the end of unit 1, students are expected to:

1. define argument, validity, and conclusion;
2. enumerate the different rules of inference;
3. translate the arguments symbolically; and
4. use the rules of inference to construct a proof using the given hypotheses.

MATERIALS/RESOURCES

Browser
Text Editor / Word Processor

LESSON

Introduction to Logic

A. Introduction

Mathematical logic is often supported by proofs. Proofs in mathematics are valid arguments that establish the truth of mathematical statements. An rrgument is a sequence of statements, premises, that ends with a conclusion. The symbol " ∴ " (read as, therefore) is placed before the conclusion. A deductive argument is said to be valid if and only if it takes a form that makes it impossible for the premises to be true and the conclusion to be false. A fallacy is an incorrect reasoning which leads to invalid arguments.

Example:

"If you have the current password,
then you can log onto the network."

"You have a current password."

Argument

Therefore, "You can log onto the network."

Conclusion

Prepared by:
Myleen A. Beninsig

Structure of an Argument:

First Premise
Second Premise
Third Premise
.
.
nth Premise

$\therefore$ Conclusion

An argument is valid if whenever all the hypotheses are true, the conclusion is also true.

Defintion: 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 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.

B. The Rules of Inference

- Rules of inference are rules that provide the justification of the steps used to show that a conclusion follows logically from a set of hypothesis.
- Pattern establishing that if we know that a set of *hypotheses* are all true, then a certain related *conclusion* statement is true.

1. **Addition:**

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

Example: Let p: It is below freezing
q: It is raining now.

Answer: <u>It is below freezing.</u>
$\therefore$ It is below freezing or it is raining now.

2. **Simplification**

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

Example: Let p: Covid-19 is an infectious disease.
q: Fever is one of its symptoms.

Answer: Covid-19 is an infectious disease and fever is one of its symptoms.
∴ Covid-19 is an infectious disease.

3. Conjunction

$$\begin{array}{l} p \\ \underline{q} \\ \therefore p \wedge q \end{array}$$

Example: Let p: Discrete Structures is easy.
Let q: I study hard.

Answer: Discrete Structures is easy.
I study hard.
∴ Discrete Structure is easy and I study hard.

4. Modus Ponens (Law of Detachment)

$$\begin{array}{l} p \rightarrow q \\ \underline{p} \\ \therefore q \end{array}$$

Example: Let p: Discrete Structures is easy.
q: I study hard.

Answer: If Discrete Structures is easy, then I study hard.
Discrete Structures is easy.
∴ I study hard.

5. Modus Tollens

$$\begin{array}{l} p \rightarrow q \\ \underline{\sim q} \\ \therefore \sim p \end{array}$$

Example: Let p: Discrete Stuctures is easy
q: I study hard

Answer: If Discrete Structures is easy, then I study hard.
Discrete Structures is not easy.
∴ I do not study hard.

6. Hypothetical Syllogism

$$\begin{array}{l} p \rightarrow q \\ \underline{q \rightarrow r} \\ \therefore p \rightarrow r \end{array}$$

Example: Let p: Covid-19 is an infectious disease.
q: Fever is one of its symptoms.
r: You have to stay at home.

Answer: If Covid-19 is an infectious disease, then fever is one of its symptoms.
If fever is one of its symptoms, you have to stay at home.
∴ If Covid-19 is an infectious disease, then you have to stay at home.

7. **Disjunctive Syllogism**

$$\begin{array}{l} p \vee q \\ \underline{\sim p} \\ \therefore q \end{array}$$

Example: Let p: It is sunny outside.
q: I carry my umbrella.

Answer: It is sunny outside or I carry my umbrella.
It is not sunny outside.
∴I do not carry my umbrella.

8. **Resolution**

$$\begin{array}{l} p \vee q \\ \underline{\sim p \vee r} \\ \therefore q \vee r \end{array}$$

Example: Let p: It is sunny outside.
q: I carry my umbrella.
r: I walk to work.
Answer: It is sunny outside or I carry my umbrella.
It is not sunny outside or I walk to work.
∴ I carry my umbrella or I walk to work.

9. **Constructive Dilemma**

$$\begin{array}{l} p \rightarrow q \\ r \rightarrow s \\ \underline{p \vee r} \\ \therefore q \vee s \end{array}$$

Example: Let p: Discrete Structures is a required course.
q: Tolkien wrote the Lord of the Rings.
r: Covid-19 is treatable.
s: Classes is online this third trimester.

Answer: If Discrete Structures is a required course, then Tolkien wrote the Lord of the Rings.
If covid-19 is treatable, then classes is online this third trimester.
Discrete Structures is a required course or covid-19 is treatable.
∴ Tolkien wrote the Lord of the Rings or classes is online this third trimester

10. Destructive Dilemma

$$\begin{array}{l} p \rightarrow q \\ r \rightarrow s \\ \underline{\sim q \vee \sim s} \\ \therefore \sim p \vee \sim r \end{array}$$

Example: Let p: Discrete Structures is a required course.
q: Tolkien wrote the Lord of the Rings.
r: Covid-19 is treatable.
s: Classes is online this third trimester.

Answer: If Discrete Structures is a required course, then Tolkien wrote the Lord of the Rings.
If covid-19 is treatable, then classes is online this third trimester.
Tolkien did not write the Lord of the Rings or classes is not online this third trimester.

∴ Discrete Structures is not a required course or covid-19 is not treatable.

C. Using Rules of Inference to Build Arguments

When there are many premises, several rules of inference are often needed to show that an argument is valid. The following are examples where we use the rules of inference to establish the validity of the argument.

Examples:

1. Randy works hard. If Randy works hard, then he is a dull boy. If Randy is a dull boy, then he will not get a job. Therefore, he will not get a job.

   Solution: Let p: Randy works hard.
   q: Randy is a dull boy.
   ~ r: He will not get a job.

   Symbols:

$$\left.\begin{array}{l} p \\ p \rightarrow q \\ \underline{q \rightarrow \sim r} \end{array}\right\} \text{Premises/Hypothesis}$$

$$\therefore \sim r \longrightarrow \text{Conclusion}$$

   In this argument, we need to establish the validity of the conclusion. Since there is no direct rule of inference that will support the conclusion the following steps with reasons will be explicitly stated.

   Proof:

Step	**Reason**
1. p	Premise
2. $p \rightarrow q$	Premise
3. q	Modus Ponens 1, 2

4. $q \to \sim r$	Premise
5. $\sim r$	Modus Ponens 3, 4

.: ***Valid***

In proving the validity of the conclusion, you always start with the premises. Start with any two (2) of the premises, check if there is a conclusion to be drawn from the two (2) premises, until you reach the conclusion.

Use all premises from the argument, however, do not use/write three (3) premises consecutively, unless the conclusion to be drawn is a constructive dilemma or destructive dilemma.

Another Proof:

Step	**Reason**
1. $p \to q$	Premise
2. $q \to \sim r$	Premise
3. $p \to \sim r$	Hypothetical Syllogism 1, 2
4. p	Premise
5. $\sim r$	Modus Ponens 3, 4

.: ***Valid***

2. It is not sunny this afternoon and it is colder than yesterday. If we will go swimming, then it is sunny this afternoon. If we do not go swimming, then we will take a canoe trip. If we take a canoe trip, then we will be home by sunset. Therefore, we will be home by sunset.

Solution: Let p: It is sunny this afternoon.
$\sim p$: It is not sunny this afternoon.
q: It is colder than yesterday.
r: We will go swimming.
$\sim r$: We do not go swimming.
s: We will take a canoe trip.
t: We will be home by sunset.

Symbol: $\sim p \wedge q$

$r \rightarrow p$

$\sim r \rightarrow s$

$\underline{s \rightarrow t}$

$\therefore t$

Proof:

Step	**Reason**
1. $\sim p \wedge q$	Premise
2. $\sim p$	Simplification 1
3. $r \rightarrow p$	Premise
4. $\sim r$	Modus Tollens 2, 3
5. $\sim r \rightarrow s$	Premise
6. s	Modus Ponens 4, 5
7. $s \rightarrow t$	Premise
8. t	Modus Ponens 6, 7

$\therefore$ ***Valid***

Another Proof

Step	**Reason**
1. $\sim r \rightarrow s$	Premise
2. $s \rightarrow t$	Premise
3. $\sim r \rightarrow t$	Hypothetical syllogism 1, 2
4. $\sim p \wedge q$	Premise
5. $\sim p$	Simplification 4
6. $r \rightarrow p$	Premise
7. $\sim r$	Modus Tollens 5, 6
8. t	Modus Ponens 3, 7

$\therefore$ ***Valid***

PRACTICE EXERCISE

A. Translate the arguments symbolically. What rule of inference is used in each of these arguments?

1. I study hard. Therefore, I study hard or I get passing grade.
2. I study hard and I get passing grade. Therefore, I study hard.
3. Math is easy. I passed. Therefore, math is easy and I passed.
4. If I will die, then the students are happy. The students are not happy. Therefore, I will not die.
5. If Math is easy, then I will get passing grade. If I get passing grade, then I will be happy. Therefore, if Math is easy, then I will be happy.

B. Formulate the arguments symbolically and determine whether each is valid. Let

p: I study hard.
q: I get A's.
r: I get rich.

1. If I study hard, then I get A's.
I study hard.
∴ Therefore, I get A's.

2. If I study hard, then I get A's.
if I don't get rich, then I don't get A's.
∴ therefore, I get rich.

3. If I study hard or I get rich, then I get A's.
I get A's.
∴ if I don't study hard, then I get rich.

4. If I study hard, then I get A's.
I don't get A's.
∴ therefore, I don't study hard.

5. If I study hard, then I get A's.
if I get A's, then I get rich.
∴ therefore, If I study hard, then I get rich.

C. Use rules of inference to establish the validity of the argument.

1. If you send me an email message, then I will finish writing the program. If you do not send me an email message, then I will go to sleep early. If I go to sleep early, then I will wake up feeling refreshed. Therefore, if I do not finish writing the program, then I will wake up feeling refreshed.

2. If Jill can sing or Jack can play, then I'll buy the compact disc. Jill can sing. I'll buy the compact disc player. Therefore, I'll buy the compact disc and the compact disc player.

Prepared by:
Myleen A. Beninsig

3. If Joel is enrolled in Java, then Marlene is enrolled in COBOL. Joel is enrolled in Java and Francis is enrolled in C++. Therefore, Marlene is enrolled in COBOL.

GRADED ACTIVITY

A separate activity will be given (posted).

REFERENCES

1. Rosen, Kenneth D (2007). Discrete mathematics and its applications. 6th ed. Boston : McGraw Hill Higher Education.
2. Introduction to Logic. Taken on June 15, 2020 from http://intrologic.stanford.edu/notes/notes.html
3. https://www.tutorialspoint.com/discrete_mathematics/rules_of_inference.htm
4. https://www.geeksforgeeks.org/mathematical-logic-rules-inference/
5. https://www.youtube.com/watch?v=8DW0K3mnc-0
6. https://www.youtube.com/watch?v=HcS4lqXxrV4
7. https://www.youtube.com/watch?v=28lebQ60TCc

ANSWERS TO SELECTED PRACTICE EXERCISE

A. A. Translate the arguments symbolically. What rule of inference is used in each of these arguments?

1. I study hard. Therefore, I study hard or I get passing grade.

Solution: Let p: I study hard.
q: I get passing grades.

Answer: $\underline{\quad p \quad}$
$\therefore p \vee q$
Addition Rule

3. Math is easy. I passed. Therefore, math is easy and I passed.

Solution: Let p: Math is easy.
q: I passed.

Answer: p
$\underline{\quad q \quad}$
$\therefore p \wedge q$
Conjunction Rule

B. Formulate the arguments symbolically and determine whether each is valid. Let

p: I study hard.
q: I get A's.
r: I get rich.

1. If I study hard, then I get A's.
   I study hard.
   ∴ Therefore, I get A's.

Solution: $p \rightarrow q$

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

Valid: Modus Ponens

4. If I study hard, then I get A's.
   I don't get A's.
   ∴ Therefore, I don't study hard.

Solution: $p \rightarrow q$

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

Valid: Modus Tollens

C. Use rules of inference to establish the validity of the argument.

1. If you send me an email message, then I will finish writing the program. If you do not send me an email message, then I will go to sleep early. If I go to sleep early, then I will wake up feeling refreshed. Therefore, if I do not finish writing the program, then I will wake up feeling refreshed.

Solution: Let p: You send me an email message.
$\sim p$: You do not send me an email message.
q: I will finish writing the program.
$\sim q$: I do not finish writing the program.
r: I will go to sleep early.
s: I will wake up feeling refreshed.

Symbols: $p \rightarrow q$
$\sim p \rightarrow r$

$$\frac{r \rightarrow s}{\therefore \sim q \rightarrow s}$$

Proof:

Step	**Proof**
1. $p \rightarrow q$	Premise
2. $\sim q \rightarrow \sim p$	Contrapositive 1
3. $\sim p \rightarrow r$	Premise
4. $\sim q \rightarrow r$	Hypothetical Syllogism 2, 3

5. $r \rightarrow s$	Premise
6. $\sim q \rightarrow s$	Hypothetical Syllogism 4, 5
$\therefore$ ***Valid***	

Prepared by:
Myleen A. Beninsig

CC9 – Discrete Structures
Unit 2 – Methods of Proof

Lesson

INTRODUCTION

The methods of proof discussed in this unit are important not only because they are used to prove mathematical theorems, but also for their many applications to computer science. These applicatons include verifying that computer programs are correct, establishing that operating systems are secure, making inferences in artificial intelligence, showing that system specifications are consistent, and so on. Consequently, understanding the techniques used in proofs is essential both in mathematics and in computer science.

MATHEMATICAL SYSTEMS

This section presents an overview of what a mathematical system is and how logic plays an important role in one. The system of propositions and logical operators developed will serve as a model for our discussions.

Mathematical system consist of:

1. ***Undefined terms*** – these are basic building blocks of a mathematical system. These are words that are accepted as starting concepts of a mathematucal system.
   Example: in Euclidean geometry we have undefined terms such as points and lines.

2. ***Defintions*** – are propositions constructed from undefined terms and previously accepted concepts in order to create a new concept.
   Examples: Defintion of parallel lines,
   Two acute angles are complementary if their sum is 90°.
   If a triangle is isosceles, then it has two congruent sides.

3. ***Axioms(Postulates, Conjectures)*** – are propositions accepted as true without proof within the mathematical system. Axioms are assumed to be true.
   Examples: In Eulidean geometry, the following are axioms:
   - Given two distinct points, there is exactly one line that contains them.
   - Two distinct parallel lines never meet.
   - Given a line and a point not on the line, there is exactly one line through the point which is parallel to the line.

4. ***Theorems*** – A *theorem* is a proposition of the form

   $$p \rightarrow q$$

   which must be shown to be true by a sequence of logical steps that assume that p is true, anduse definitions, axioms and previously proven theorems.

   - *Theorem* is a proposition that has been proved to be true.

In relation to logic, propositional calculus is a formal name for logical system. The universe consists of *propositions*. The *axioms* are the truth tables for the logical operators and the key *definitions* are those of equivalence and implication. We use propositions to describe any other mathematical system; this is the minimum amount of structure that a mathematical system can have.

PROOF

Definition: A *proof* is a logical argument that consists of a series of steps using propositions in such a way that the truth of the theorem is established.

- A proof of a theorem is a finit sequence of logically valid steps that demonstrate that the premises of a theorem imply the conclusion.

In mathematics, ***a proof*:**

- Must be ***correct*** (well-reasoned, logically valid) and ***complete*** (clear, detailed) that rigorously & undeniably establishes the truth of a mathematical statement.

Why must the argument be correct & complete?

- *Correctness* prevents us from fooling ourselves.
- *Completeness* allows anyone to verify the result.

PROOFS IN PROPOSITIONAL CALCULUS

Theoretically, we can prove anything in propositional calculus with truth table. In fact, the laws of logic (equivalence) are all theorems. Propositional calculus is one of the few mathematical systems for which any valid sentence can be determined true or false bymechanical means. A program to write truth tables is not too difficult to write; however, what can be done theoretically is not always practical.

DIRECT PROOFS (CHAIN OF REASONING)

A *direct proof* is a proof in which the truth of the premises of a theorem are shown to *directly imply the truth of the theorem's conclusion*. It is a conditional statement $p \to q$ 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. In direct proof, we assume that p is true and use axioms, definitions, and previously proven theorems, together with the rules of inference and equivalence laws, to show that q must also be true.

Example: Theorem:

$$\begin{array}{l} p \to r \\ q \to s \\ p \vee q \\ \hline \therefore s \vee r \end{array}$$

Proof:

Steps	**Reason**
1. $p \vee q$	Premise
2. $\sim p \to q$	Implication 1
3. $q \to s$	Premise
4. $\sim p \to s$	Hypothetical Syllogism 2, 3
5. $\sim s \to p$	Contrapositive 4
6. $p \to r$	Premise
7. $\sim s \to r$	Hypothetical Syllogism 5, 6
8. $s \vee r$	Implication 7

$\therefore$ Valid

Rules for Formal Proofs:

1. A proof must end in a finite number of steps.
2. Each step must be either a premise or a proposition that is implied from prevoius steps using any valid rule of inference or equivalence laws.
3. For a direct proof, the last step must be the conclusion of the theorem. For an indirect proof, the last step must be a contradiction.

INDIRECT PROOFS/PROOF BY CONTRADICTION

A *proof by contradiction* assumes that the hypothesis *p* is true and that the conclusion *q* is false. A contradiction is a proposition of the form $r \wedge \sim r$.

Consider a theorem $P \Rightarrow Q$, (*read as P therefore Q*) where P represents p_1, p_2, …, and p_n, the premises. The method of indirect proof is based on the equivalence $P \to Q \Leftrightarrow \sim (P \wedge \sim Q)$.

In other words, this logical law states that if $P \rightarrow Q$, then $P \wedge \sim Q$ is always false, that is, $P \wedge \sim Q$ is a contradiction. This means that a valid method of proof is to negate the *conclusion of a theorem and add this negation* to the premises. If a contradiction can be implied from this set of propositions, the proof is complete. Indirect proofs are often more convenient than direct proofs in certain situations.

Example: Theorem:

$$\begin{array}{l} \sim p \vee q \\ s \vee p \\ \sim q \\ \hline \therefore s \end{array}$$

Proof:

Steps	**Reason**
1. $\sim s$	Negated Conclusion
2. $s \vee p$	Premise
3. p	Disjunctive Syllogism 1, 2
4. $\sim p \vee q$	Premise
5. q	Disjunctive Syllogism 3, 4
6. $\sim q$	Premise
7. $q \wedge \sim q$	Contradiction 5, 6

$\therefore$ Valid

PROOF BY RESOLUTION

Computer programs have been developed to automate the task of reasoning and proving theorems. Many of these programs make use of a rule of inference known as resolution. This is a proof technique that depends on a single rule:

$$\left.\begin{array}{l} p \vee q \\ \sim p \vee r \end{array}\right\} \rightarrow \text{clause separated by or's}$$
$$\begin{array}{l} \hline \therefore q \vee r \quad \rightarrow \text{resolvent} \end{array}$$

Examples: Prove the following theorems using resolution:

1. $$\begin{array}{l} a \vee b \\ \sim a \vee c \\ \sim c \vee d \\ \hline \therefore b \vee d \end{array}$$

Proof:

Steps	**Reason**
1. $a \vee b$	Premise
2. $\sim a \vee c$	Premise
3. $b \vee c$	Resolution 1, 2
4. $\sim c \vee d$	Premise
5. $b \vee d$	Resolution 3, 4

$\therefore$ Valid

2. $\mathbf{p \rightarrow q}$

$$\begin{array}{l} q \rightarrow r \\ \underline{\sim r} \\ \therefore \sim p \end{array}$$

Solution: Convert the theorem to clausal form using equivalence laws

$$\begin{array}{l} \sim p \vee q \\ \sim q \vee r \\ \underline{\sim r} \\ \therefore \sim p \end{array}$$

Proof:

Steps	**Reason**
1. $\sim p \vee q$	Premise
2. $\sim q \vee r$	Premise
3. $\sim p \vee r$	Resoution 1, 2
4. $\sim r$	Premise
5. $\sim p$	Resolution 3, 4

Note: *Disjunctive syllogism is a special case of resolution*.
$\therefore$ Valid

3. Randy works hard. If Randy works hard, then he is a dull boy. If Randy is a dull boy, then he will not get a job. Therefore, he will not get a job.

Solution:Let p: Randy works hard.
q: He is a dull boy.
~r: He will not get a job.

$$\begin{array}{l} p \\ p \rightarrow q \\ \underline{q \rightarrow \sim r} \\ \therefore \sim r \end{array}$$

Convert the theorem to clausal form using equivalence laws

p
$\sim p \vee q$
$\sim q \vee \sim r$
$\therefore \sim r$

Proof:

Steps	**Reason**
1. p	Premise
2. $\sim p \vee q$	Premise
3. q	Resolution 1, 2
4. $\sim q \vee \sim r$	Premise
5. $\sim r$	Resolution 3, 4

$\therefore$ Valid

Video Lessons:
1. Methods of Proof (54 minutes)
2. Four Basic Proof Techniques used in Mathematics (22 minutes)

Supplementary Reading:
Methods of Proof

References

1. Rosen, Kenneth D (2007). Discrete mathematics and its applications. 6th ed. Boston : McGraw Hill Higher Education.
2. Doerr, Alan & Levasseur, Kenneth. Applied Discrete Structures. USA.
3. Propositional Proofs. http://intrologic.stanford.edu/notes/chapter_04.html
4. Methods of Proof. https://www.math.fsu.edu/~pkirby/mad2104/SlideShow/s3_2.pdf