Bachelor of Science in Computer Science
University of Colombo School of Computing

SCS 1302 – Discrete Mathematics 1

Prof. D D Karunarata
(Lecture Hall - S104)

Discrete Mathematics

Main Topics :

1. Propositional Logic
2. Predicate Logic
3. Arguments
4. Introduction to Proofs
5. Sets
6. Functions
7. Relation
8. Number Theory

Textbook

Discrete Mathematics and Its Applications

By Kenneth H. Rosen

Published by McGraw-Hill

(7^{th} Edition)

Chapters 1, 2, 4, and 9.

Lesson 1: Propositional Logic

- **Topics to be covered.**
  - **Propositions**
  - **Primitive propositions, logical operators, Compound Propositions and Truth Tables**
  - **Equivalences**
  - **Precedence of Logical Operators**
  - **Tautology and Contradiction**
  - **Normal Forms**
  - **Applications of Propositional Logic**
  - **Satisfiability**

Propositions

A **proposition** is a **declarative** sentence (assertion) that is either **true (1)** or **false(0)**, but not both. The **true** or **false** value that can be assigned to a proposition is called its **truth value**.

A proposition can be viewed as a sentence that can be assigned with a truth value.

The area of logic that deals with propositions is called the **propositional calculus or propositional logic.**

Examples

Examples of propositions:

- Colombo is the biggest city of Sri Lanka.
- $2 + 3 = 4$.
- The Moon is made of cheese.

The following sentences are not propositions:

- What time is it?
- Please be silent!
- $X + 5 = 3$.

The first two are not declarative sentences while the truth value of the third one depends on the value of X. Without knowing it, this statement cannot be assigned with a truth value.

Atomic proposition (Primitive proposition).

A statement that cannot be broken down into smaller statements, is known as an **atomic** proposition (or **primitive** proposition).

Atomic proposition (Primitive proposition).

Examples

1. 2 is a prime number.
2. $6 > 3$.
3. $3 > 6$.
4. Kamal is left-handed.

- Sentence 1 and sentence 2 are true. Sentence 3 is false. We can't say for certain whether sentence 4 is true or false without knowing who Kamal is. However, it is either true or false since typically a person cannot be both a left-handed and a right-handed.
- The four sentences above are atomic statements (propositions).

Propositional Variables

- Letters such as p,q,r and s are used to denote atomic propositions. These letters are known as **propositional variables** since they can be assigned with different propositions at different times.
- Generally, a truth value of T (or 1) for true or F (or 0) for false is assigned to each propositional variable.

Compound Propositions

- Statements that combine more than one primitive proposition by using **logical operators (connectors)** are called **compound propositions**.

Logical Operator (Connector)	Symbol
Negation	$\neg$
Conjunction (and)	$\wedge$
Disjunction (or)	$\vee$
Exclusive Disjunction	$\oplus$
Implication (Conditional)	$\rightarrow$
implication (Bi – conditional)	$\leftrightarrow$

Conjunction – and, product
Disjunction – or , sum

Compound Propositions using propositional variables

The following statements are **compound propositions**?

1. 5 is a prime number and 1 = 0.

   p q

   $p \wedge q$

1. Nimal is holding a pencil or water is a liquid.

   r s

   $r \vee s$

1. If Jack has a dog, then cat have 5 lungs.

   p q

   p -> q

Compound Propositions - Examples

If compound propositions are also propositions, they should have truth values.

How to determine the truth values of compound positions?

Truth Tables

- A truth table is a table that displays all possible truth values for the primitive propositions of a compound proposition and the corresponding truth value of the compound proposition.

Negation

Let *p be a proposition. The* ***negation of p****, denoted by* $\neg$ ***p****,* is the statement "It is not the case that *p.*"

Negation is a unary operator

Example: Let p be "*Nimal's phone has at least 32GB of memory* ".

The negation of p is

$\neg$ *(Nimal's phone has at least 32GB of memory)*

"*It is not the case that Nimal's phone has at least 32GB of memory*." Or

"*Nimal's phone does not have at least 32GB of memory*" Or Simply

"*Nimal's phone has less than 32GB of memory*."

Truth Table for negation:

p	$\neg p$
F	T
T	F

Conjunction & Disjunction

Let *p and q be propositions. The **conjunction(product) of p and q,** denoted by* ***p ∧ q****, is the proposition "p and q." The conjunction p ∧ q is true when both p and q are true and is false otherwise.*

Let *p and q be propositions. The **disjunction(sum) of p and q,** denoted by* ***p ∨ q****, is the proposition "p or q." The disjunction p ∨ q is false when both p and q are false and is true otherwise.*

Truth Table for Conjunction

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

Truth Table for Disjunction

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

Examples

Let p and q represent the propositions:
p : It is below freezing. q : It is snowing.

Write these propositions using p and q and logical connectives (including negations).

- It is below freezing and snowing.
  $p \wedge q$
- It is below freezing and not snowing.
  $p \wedge \neg q$
- It is not below freezing, and it is not snowing.
  $\neg p \wedge \neg q$
- It is either snowing or below freezing (**or both**).
  $p \vee q$

Exclusive Disjunction (exclusive or)

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

Example: “ Either soup or salad comes with the meal”

The truth table for the exclusive or

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

Conditional Statement or Implication

Let *p and q be propositions. The* ***conditional statement p → q*** *is the proposition* ***"if p, then q."*** *The conditional statement p → q is false when p is true and q is false, and true otherwise.*

p is called the ***hypothesis*** *(or antecedent or premise)*

and *q is called the* ***conclusion*** *(or consequence).*

Various terminology used:

- *"p is sufficient for q"*
- "a sufficient condition for *q is p"*
- *"p implies q"*
- *"q whenever p"*

Truth Table for p → q

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

Example

- Consider the following statement:

  If it rains, It will be cloudy

  x y

  x -> y

- Case 1: It rained then we say, It should be cloudy. So If x happened then y should happen. x -> y
- Case 2: There are no clouds, So there is no rain. ~ y -> ~ x
- Case 3: It is not raining. Uncertain. As there may be clouds or may not be.
- Case 4: It is cloudy. Uncertain. As it may rain may not rain.

Example

p	q	P -> q
F	F	T
F	T	T
T	F	F
T	T	T

Example

p	q	P -> q
F	F	T
F	T	T
T	F	F
T	T	T

Example

p	q	P -> q
F	F	T
F	T	T
T	F	F
T	T	T

Examples of a conditional statement

Let ***p*** *be "Kamal learns discrete mathematics" and*
q *be "Kamal will* find a good job."
Express the statement ***p* → *q*** *as a statement in English.*

Solution: $p \rightarrow q$ *represents the proposition* " If Kamal learns discrete mathematics, then he will find a good job."

$p \rightarrow q$ can also be expressed as:

- "Kamal will find a good job when he learns discrete mathematics."
- "For Kamal to get a good job, it is sufficient for him to learn discrete mathematics."

Equivalent propositions

When two compound propositions always have the same truth values for all possible truth values for its primitive propositions, we call them **equivalent($\equiv$).**

$p \rightarrow q$ and $\sim q \rightarrow \sim p$ are equivalent ($p \rightarrow q \equiv \sim q \rightarrow \sim p$)

Equivalence

p	q	$\neg p$	$\neg q$	$p \to q$	$q \to p$	$\neg q \to \neg p$	$\neg p \to \neg q$
0	0	1	1	1	1	1	1
0	1	1	0	1	0	1	0
1	0	0	1	0	1	0	1
1	1	0	0	1	1	1	1

$$p \to q \equiv \neg q \to \neg p \qquad\qquad \mathbf{q} \to p \equiv \neg p \to \neg q$$

Bi-conditionals or Bi-implications

Let *p and q be propositions. The* ***bi-conditional*** *statement* $\boldsymbol{p \leftrightarrow q}$ *is the proposition "p if* and only if *q." The bi-conditional statement* $p \leftrightarrow q$ *is true when p and q have the same truth* values, and is false otherwise.

Truth Table for $\mathbf{p \leftrightarrow q}$

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

Other ways to express $p \leftrightarrow q$:

- *"p is necessary and sufficient for q"*
- "if *p then q, and if q then p"*
- *"p iff q."*

Example: "You can have dessert if and only if you finish your meal."

Precedence of Logical Operators

$\vee$ and $\rightarrow$ are binary operators

Does $p \vee q \rightarrow r$ *mean*

1) $(p \vee q) \rightarrow r$ *or*

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

Precedence of Logical Operators

Precedence of Logical Operators	
Operator	**Precedence (Highest to lowest)**
$\neg$	1
$\wedge$	2
$\vee$	3
$\rightarrow$	4
$\leftrightarrow$	5

$p \vee q \rightarrow r$ is the same as $(p \vee q) \rightarrow r$

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

Precedence of Logical Operators

- All operators are right-associative

$$p \vee q \vee r$$

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

The two binary operators – and + have the same precedence

5 – 3 + 2

(5 – 3) + 2 = 4 left associative

5 – (3+2) = 0 right associative

Construction of truth tables for compound propositions

Let a compound proposition C consist of the atomic propositions p1;p2; : : : ;pn.

- To compute the truth table do the following:

1. List its atomic propositions p1;p2; : : : ;pn and count their number, n.
2. Form a table with $m = 2^n$ rows.
3. List all possible combinations of the truth values for p1;p2; : : : ;pn.
4. Prioritize the logical operations in C.
5. Compute the truth values in order of the operators.

Example

- Let the compound proposition C be $(p \wedge q) \rightarrow p$.
  Count the number of atomic propositions, n = 2.
  Form a table with $m = 2^2 = 4$ rows.
  List all possible combinations of the truth values for p and q.
  Prioritize the logical operations in the proposition $(p \overset{1}{\wedge} q) \overset{2}{\rightarrow} p.$
- Compute first 1 and then 2

Example

p	**q**	p ^q	(p ^q) → p.
T	T		
T	F		
F	T		
F	F		

Example

p	**q**	p ^q	(p ^q) → p.
T	T	T	
T	F	F	
F	T	F	
F	F	F	

Example

p	q	p ^q	(p ^q) → p.
T	T	T	T
T	F	F	T
F	T	F	T
F	F	F	T

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

Truth Tables of Compound Propositions

Construct the truth table of the compound proposition

$$(p \vee \neg q) \longrightarrow (p \wedge q).$$

p	q	$\neg q$	$p \vee \neg q$	$p \wedge q$	$(p \vee \neg q) \longrightarrow (p \wedge q)$
0	0	1	1	0	0
0	1	0	0	0	1
1	0	1	1	0	0
1	1	0	1	1	1

Construction of truth tables

p	q	r
T		
T		
T		
T		
F		
F		
F		
F		

Construction of truth tables

p	q	r
T	T	
T	T	
T	F	
T	F	
F	T	
F	T	
F	F	
F	F	

Construction of truth tables

p	q	r
T	T	T
T	T	F
T	F	T
T	F	F
F	T	T
F	T	F
F	F	T
F	F	F

Tautology and Contradiction

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***.

Tautology

p	$\neg p$	$p \vee \neg p$
T	F	T
F	T	T

Contradiction

p	$\neg p$	$p \wedge \neg p$
T	F	F
F	T	F

$p \rightarrow q \equiv \neg p \vee q$

p	q	$p \rightarrow q$	$\neg p$	$\neg p \vee q$	$(p \longrightarrow q) \leftrightarrow (\neg p \vee q)$
0	0	1	1	1	1
0	1	1	1	1	1
1	0	0	0	0	1
1	1	1	0	1	1

Logical Equivalences

Identity laws:

$$p \wedge T \equiv p$$
$$p \vee F \equiv p$$

Domination laws:

$$p \vee T \equiv T$$
$$p \wedge F \equiv F$$

Idempotent laws:

$$p \vee p \equiv p$$
$$p \wedge p \equiv p$$

Comutative laws:

$$p \vee q \equiv q \vee p$$
$$p \wedge q \equiv q \wedge p$$

Associative laws:

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

Distributive laws:

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

De Morgan's laws:

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

Absorption laws:

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

Negation laws:

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

Double Negation law:

$$\neg(\neg p) \equiv p.$$

Logical Equivalences

$$P \oplus q \equiv (p \land \neg q) \lor (\neg p \land q)$$
$$P \oplus q \equiv \neg(p \land q) \land (p \lor q)$$

p	q	¬p	¬q	p ∧ ¬q	(p ∧ ¬q)	(¬p ∧ q)	(p ∧ ¬q) ∨ (¬p ∧ q)	p ⊕ q
T	T	F	F	F	F	F	F	F
T	F	F	T	T	T	F	T	T
F	T	T	F	F	F	T	T	T
F	F	T	T	F	F	F	F	F

Principle of Duality

- The principle of duality in logic states that if a Boolean expression or statement is true, its dual is also true. The dual is formed by interchanging "AND" and "OR" operations, and 0 and 1 (representing false and true, respectively).

Examples

Identity Law: $p \land \boldsymbol{T} \equiv p$

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

De Morgan's Law: $\neg(p \land q) \equiv \neg p \lor \neg q$

p	q	$\neg p$	$\neg q$	$(p \land q)$	$\neg(p \land q)$	$\neg p \lor \neg q$
0	0	1	1	0	1	1
0	1	1	0	0	1	1
1	0	0	1	0	1	1
1	1	0	0	1	0	0

Distributive Law:
$p \vee (q \wedge r) \equiv (p \vee q) \wedge (p \vee r)$

p	q	r	$(q \wedge r)$	$p \vee (q \wedge r)$	$(p \vee q)$	$(p \vee r)$	$(p \vee q) \wedge (p \vee r)$
0	0	0	0	0	0	0	0
0	0	1	0	0	0	1	0
0	1	0	0	0	1	0	0
0	1	1	1	1	1	1	1
1	0	0	0	1	1	1	1
1	0	1	0	1	1	1	1
1	1	0	0	1	1	1	1
1	1	1	1	1	1	1	1

More Logical Equivalences

Logical Equivalences Involving Conditional Statements
$p \rightarrow q \equiv \neg p \vee q$
$p \rightarrow q \equiv \neg q \rightarrow \neg p$
$p \vee q \equiv \neg p \rightarrow q$
$p \wedge q \equiv \neg(p \rightarrow \neg q)$
$\neg(p \rightarrow q) \equiv p \wedge \neg q$
$(p \rightarrow q) \wedge (p \rightarrow r) \equiv p \rightarrow (q \wedge r)$
$(p \rightarrow r) \wedge (q \rightarrow r) \equiv (p \vee q) \rightarrow r$
$(p \rightarrow q) \vee (p \rightarrow r) \equiv p \rightarrow (q \vee r)$

Logical Equivalences Involving Bi-conditional Statements
$p \leftrightarrow q \equiv (p \rightarrow q) \wedge (q \rightarrow p)$
$p \leftrightarrow q \equiv \neg p \leftrightarrow \neg q$
$p \leftrightarrow q \equiv (p \wedge q) \vee (\neg p \wedge \neg q)$
$\neg(p \leftrightarrow q) \equiv p \leftrightarrow \neg q$

Examples

Prove the following logical identities by developing a series of logical equivalences:

1. $\neg\big(p \vee (\neg p \wedge q)\big) \equiv \neg p \wedge \neg \mathrm{q}$
2. $(p \wedge q) \rightarrow (p \vee q) \equiv T$
3. $(p \rightarrow q) \vee (p \rightarrow r) \equiv \neg p \vee q \vee r.$

$$
\begin{aligned}
\neg\big(p \vee (\neg p \wedge q)\big) &\equiv \neg p \wedge \neg(\neg p \wedge q) && (\because \text{De Morgan's Law}) \\
&\equiv \neg p \wedge [(\neg\neg p) \vee \neg q)] && (\because \text{De Morgan's Law}) \\
&\equiv \neg p \wedge [p \vee \neg q] && (\because \text{Double negation}) \\
&\equiv (\neg p \wedge p) \vee (\neg p \wedge \neg q) && (\because \text{Distributive law}) \\
&\equiv F \vee (\neg p \wedge \neg q) && (\because \text{Negation law}) \\
&\equiv (\neg p \wedge \neg q) && (\because \text{Identity law})
\end{aligned}
$$

Do we need all logical operators?

$\vee$ - Disjunction

$\wedge$ - Conjunction $p \land q \equiv \neg(\neg p \vee \neg q)$.

$\neg$ - Negation

$\rightarrow$ - Implication $p \rightarrow q \equiv \neg p \vee q$

$\oplus$ - Exclusive or $p \oplus q \equiv (p \land \neg q) \vee (\neg p \land q)$

$\leftrightarrow$ - Bi-conditional $p \leftrightarrow q \equiv (p \rightarrow q) \land (q \rightarrow p)$

$$\equiv (\neg p \vee q) \land (\neg q \vee p).$$

Functionally Complete

A **set of logical operators** is called **functionally complete** if every compound proposition is logically equivalent to a compound proposition involving only that set of logical operators.

$\vee$ and $\neg$ form a functionally complete set of operators.

Proposition vs Formula

- A proposition is a declarative statement that can be either true or false, but not both. A formula, also known as a well-formed formula (WFF), is a formal expression, built using symbols and connectives. Essentially, a formula is the symbolic representation of a proposition.

Normal Forms

If a proposition comprises of only product (conjunction) of the variables and their negations that proposition then it is said to be in a **elementary product form**. E.g: $p \wedge q \wedge \neg r$

If a proposition comprise of only the sum (disjunction) of the variables and their negations it Is said to be in **elementary sum form**.

Example: $p \vee \neg q \vee r$

Disjunctive Normal Form: A proposition which is equivalent to a proposition which consists of a sum(∨) of elementary products (∧) it is said to be in a disjunctive normal form of the given formula.

Example: $(p \wedge q) \vee (p \wedge \neg q) \vee (\neg p \wedge q)$

Conjunctive Normal Form: A proposition which is equivalent to a given proposition, and which consists of a product of elementary sums is called a conjunctive normal form of the given formula.

Example : $(p \vee q \vee r) \wedge (\neg p \vee q \vee r)$.

Normal Forms

Disjunctive Normal Form: A proposition which is equivalent to a given proposition, and which consists of a sum of elementary products is called a disjunctive normal form of the given formula.
Example: $(p \wedge q) \vee (p \wedge \neg q) \vee (\neg p \wedge q)$

Normal forms offer a standardized way to represent logical propositions.

Normal Forms

Conjunctive Normal Form: A proposition which is equivalent to a given proposition, and which consists of a product of elementary sums is called a conjunctive normal form of the given proposition.
Example : $(p \vee q \vee r) \wedge (\neg p \vee q \vee r)$.

Example -1 (method 1)

Find the disjunctive normal form of the proposition
$(\neg p \lor \neg q) \to (p \leftrightarrow \neg q)$.

$$(\neg p \lor \neg q) \to (p \leftrightarrow \neg q) \equiv \neg(\neg p \lor \neg q) \lor (p \leftrightarrow \neg q)$$

$$\equiv (p \land q) \lor \left[(p \land \neg q) \lor \left(\neg p \land (\neg(\neg q))\right)\right]$$

$$\equiv (p \land q) \lor [(p \land \neg q) \lor (\neg p \land q)]$$

$$\equiv (p \land q) \lor (p \land \neg q) \lor (\neg p \land q).$$

Example -1 (method 2)

Find the disjunctive normal form of the formula
$x = (\neg p \vee \neg q) \rightarrow (p \leftrightarrow \neg q)$.

p	q	$\neg p$	$\neg q$	$(\neg p \vee \neg q)$	$(p \leftrightarrow \neg q)$	x
0	0	1	1	1	0	0
0	1	1	0	1	1	1
1	0	0	1	1	1	1
1	1	0	0	0	0	1

$$x = (\neg p \wedge q) \vee (p \wedge \neg q) \vee (p \wedge q).$$

Example -2 (method -1)

Find the conjunctive normal form of the formula
$x = (\neg p \vee \neg q) \rightarrow (p \leftrightarrow \neg q)$.

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

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

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

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

$$\equiv \left[p \vee \left((\neg p \vee \neg q) \wedge (q \vee p)\right)\right] \wedge \left[q \vee \left((\neg p \vee \neg q) \wedge (q \vee p)\right)\right]$$

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

$$\equiv [T \wedge (p \vee q)] \wedge [T \wedge (p \vee q)]$$

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

Example (method 2)

Find the conjunctive normal form of the formula
$x = (\neg p \vee \neg q) \rightarrow (p \leftrightarrow \neg q)$.

p	q	$\neg p$	$\neg q$	$(\neg p \vee \neg q)$	$(p \leftrightarrow \neg q)$	x	$\neg x$
0	0	1	1	1	0	0	1
0	1	1	0	1	1	1	0
1	0	0	1	1	1	1	0
1	1	0	0	0	0	1	0

$\neg x = (\neg p \wedge \neg q)$

$x = \neg(\neg x) = \neg(\neg p \wedge \neg q) = (p \vee q)$.

Applications of Propositional Logic

- Translating English sentences into expressions involving propositional variables and logical connectives.
- Translating Logical Expressions into English Sentences
- System specifications
- Logic circuits
- Logic puzzles

Translating English Sentences

Translate the following English sentences 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:

Let a:= "You can access the Internet from campus"

c:= "You are a computer science major", and

f := "You are a freshman".

Given sentence can be represented as $a \rightarrow (c \vee \neg f)$.

Note: *The conditional statement p → q can also be expressed as " p only if q".*

Exercise

Translate the following English sentences into a logical expression:

- “School is closed if more than 2 feet of snow falls or if the wind chill is below -100.”
- “To take discrete mathematics, you must have taken calculus or a course in computer science.”

Translating Logical Expressions into English Sentences

Let P be “ It is snowing”,

q be “ I will go to town” , and

r be “ I have time”.

Write a sentence in English corresponding to each of the following propositions:

1. $r \wedge q$
2. $q \leftrightarrow (r \wedge \neg p)$
3. $(q \longrightarrow r) \wedge (r \longrightarrow q)$
4. $\neg(q \vee r)$.

Classification of Propositions – Satisfiable and Unsatisfiable

In Logic a proposition C is called SATISFIABLE if there exists at least one output value T (true) in the truth table for C.

Examples

p, $\neg p$, $p \land q$

In Logic a PROPOSITION C is called UNSATISFIABLE if there are no output values T (true) in the truth table for C.

Examples

- $p \land \neg p$

Classification of Propositions – Valid

In Logic a proposition C is called VALID if its output values in the truth table are all T (true) i.e. it is a *tautology.*

Example;

$$p \vee \neg p$$

Logical Consequence

A proposition B is a logical consequence of a collection of propositions A1;A2;A3; :::An if the proposition (A1 ∧ A2 ∧ A3 ∧ :::An -> B: is valid

Example:

From p and p -> q can we conclude q?

Example

p	q	p->q	p ∧ (p->q)	(p ∧ (p->q))->q
T	T	T	T	T
T	F	F	F	T
F	T	T	F	T
F	F	T	F	T

Since (p ∧ (p->q))->q is a valid proposition q can be deduced from the propositions p and p -> q

Consistency of propositions

- A set of propositions are consistent if and only if there exists an assignment of truth values for $all\ its\ propositional\ variables$ that makes all propositions are true at the same time.

System Specifications - Example

Determine whether these system specifications are consistent:

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

Let p denote "The diagnostic message is stored in the buffer", and

let q denote "The diagnostic message is retransmitted."

Specifications as logical expressions are $p \vee q$, $\neg p$, and $p \rightarrow q$.

These system specifications are consistent if and only if there exists an assignment of truth values for p and q that makes all three specifications true.

Example Continued

p	q	$\neg p$	$p \vee q$	$p \to q$	$\neg p \wedge (p \vee q) \wedge (p \to q)$
0	0	1	0	1	0
0	1	1	1	1	1
1	0	0	1	0	0
1	1	0	1	1	0

When p is false and q is true all the specifications are true. System is consistent.

Method 2: Suppose that $\neg p$ and $p \vee q$ are true. Then

p is false ($\because$ $\neg p$ is true)

$\Longrightarrow q$ is true ($\because$ $\boldsymbol{p} \vee \boldsymbol{q}$ is true)

$\Longrightarrow p \to q$ is true

$\Longrightarrow$ When p is false and q is true all the specifications are true. System of specifications are consistent.

System Specifications

- System specifications should be consistent, that is, they should not contain conflicting requirements that could be used to derive a contradiction.

- When specifications are not consistent, there would be no way to develop a system that satisfies all specifications.

Example

Determine whether the following system specifications are consistent:

The system is in multiuser state if and only if it is operating normally. If the system is operating normally, then the kernel is functioning. The kernel is not functioning or the system is in interrupt mode. If the system is not in multiuser state, then it is in interrupt mode. The system is not in interrupt mode.

Let p denote “The system is in multiuser state”;

q denote “The system is operating normally”;

r denote “The kernel is functioning” ; and

s denote “The system is in interrupt mode.”

Specifications as logical expressions are $p \leftrightarrow q, q \rightarrow r$, $\neg r \vee s - p \rightarrow s$ and $\neg s$.

Last Example of Topic -1

Specifications as logical expressions are $p \leftrightarrow q, q \rightarrow r, \neg r \vee s, \neg p \rightarrow s$ and $\neg s$.

Suppose that $q \rightarrow r, \neg r \vee s, \neg p \rightarrow s$ and $\neg s$ are true. Then
s is false ($\because$ $\neg s$ is true)
Also $\neg p$ is false ($\because \neg p \rightarrow s$ is true) and hence p is true.
Since $\neg r \vee s$ is true and s is false, $\neg r$ must be true and hence r is false.
Since $q \rightarrow r$ is true and r is false, q must be false.

Now p is true and q is false. Therefore, $p \leftrightarrow q$ is false.

That is, whenever $q \rightarrow r, \neg r \vee s, \neg p \rightarrow s$ and $\neg s$ are all true, $p \leftrightarrow q$ is **not** true.

Therefore, this system of specifications is **not consistent**.