1.1 Propositional Logic

Our discussion begins with an introduction to the basic building blocks of logic-propositions.

Definition: A proposition

A proposition(or a statement) is a declarative sentence that is either true or false, but not both.

Examples of propositions :

1. $5 + 2 = 8$. (F)
2. Beijing is the capital of China (T)
3. $1 + 1 = 2$. (T)
4. 2 is a prime number (T)

But, the following are NOT propositions:

1. It is raining today. (either T or F)
2. How are you? (a question is not a proposition)
3. $x + 5 = 3$ (since x is not specified, neither true nor false)
4. She is very talented. (since she is not specified, neither true nor false)
5. There are other life forms on other planets in the universe. (either T or F)
6. Just do it ! (imperative command)

We use letters to denote **propositional variables** (or **statement variables**), that is, variables that represent propositions, just as letters are used to denote numerical variables.

Definition: Negation

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

Example 1: Find the negation of the proposition
p : "I have brown hair."
Solution: The negation is
$\neg p$: "It is not the case that I have brown hair"

This negation can be more simply expressed as

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

Other examples:

– $5 + 2 \neq 8$.
– 10 is not a prime number.
– It is not the case that buses stop running at 9:00pm.

Example 2 : Negate the following propositions :

1. Today is Sunday.
2. 5 is a prime number.
3. "Ahmad's smartphone has at least 32GB of memory"

Solution:

1. Today is not Sunday.
2. It 5 is not a prime number
3. It is not the case that Ahmad's smartphone has at least 32GB of memory
   or This negation can also be expressed as
   - "Ahmad's smartphone does not have at least 32GB of memory"
   or even more simply as
   - "Ahmad's smartphone has less than 32GB of memory."

• A **truth table** displays **the relationships between truth values** (T or F) of different propositions.

Truth table for NOT:

p	$\neg p$
T	F
F	T

Definition: Conjunction

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

Example 3 : Find the conjunction of the propositions *p* and *q* where *p* is the proposition "Khalid's PC has more than 16 GB free hard disk space" and *q* is the proposition "The processor in Khalid's PC runs faster than 1 GHz."

Solution:

$p \wedge q$: "Khalid's PC has more than 16 GB free hard disk space, and its processor runs faster than 1 GHz."

Other examples:

- Ptuk is located in Ramallah and $5 + 2 = 8$
- It is raining today and 2 is a prime number.
- 2 is a prime number and $5 + 2 \neq 8$.
- 13 is a perfect square and 9 is a prime.

Definition: Disjunction

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

Example 4 : What is the disjunction of the propositions *p* and *q* where *p* and *q* are the same propositions as in Example 3?

Solution:

$p \vee q$: "Khalid's PC has at least 16 GB free hard disk space, or the processor in Khalid's PC runs faster than 1 GHz."

Other examples:

- Ptuk is located in Ramallah or $5 + 2 = 8$
- Today is Sunday or 2 is a prime number.
- 2 is a prime number or $5 + 2 \neq 8$.
- 13 is a perfect square or 9 is a prime.

Truth tables for Conjunction and disjunction

• Four different combinations of values for p and q

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

Note: (the or is used inclusively, i.e., p V q is true when either p or q or both are true).

Definition : Exclusive or

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

Truth table for Exclusive or

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

Definition: Implication (Conditional Statement)

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

Notes :

1. The conditional statement $p \rightarrow q$ is false when *p* is true and *q* is false, and true otherwise.
2. $p \rightarrow q$ is read in a variety of equivalent ways:
   - if p then q
   - p only if q
   - p is sufficient for q

- q whenever p

Truth table for Conditional Statement

p	q	p → q
T	T	T
T	F	F
F	T	T
F	F	T

Example 5 : Let p be the statement “Maria learns discrete mathematics” and q the statement “Maria will find a good job.” Express the statement $p \rightarrow q$ as a statement in English.

Solution:

$p \rightarrow q$ represents the statement

- “If Maria learns discrete mathematics, then she will find a good job.”
  or
- “Maria will find a good job when she learns discrete mathematics.”
  or
- “For Maria to get a good job, it is sufficient for Maria to learn discrete mathematics.”

or

- “Maria will find a good job unless she does not learn discrete mathematics.”

Example 6 : What is the truth value of these conditional statements?

1. if Steelers win the Super Bowl in 2013 then 2 is a prime.
2. if today is Tuesday then 2 * 3 = 8.

Solution:

1. If F then T ? **T**
2. If T then F ? **F**

Remark 1 :

The mathematical concept of a conditional statement is independent of a cause-and effect relationship between hypothesis and conclusion.

For Example let the following two statements

1. "If Juan has a smartphone, then 2 + 3 = 5".
2. "If Juan has a smartphone, then 2 + 3 = 6"

- the first statement is true, because its conclusion is true. (*The truth value of the hypothesis does not matter then*.)
- but the second statement is true if Juan does not have a smartphone, even though 2 + 3 = 6 is false.

Remark 2 :

The if-then construction used in many programming languages is different from that used in logic.

Example 7 : What is the value of the variable x after the statement

if $2 + 2 = 4$ **then** $x := x + 1$

if $x = 0$ before this statement is encountered?

Solution:

Because $2 + 2 = 4$ is true, the assignment statement $x := x + 1$ is executed. Hence, x has the value $0 + 1 = 1$ after this statement is encountered.

Converse , Contrapositive, and Inverse

- The **converse** of $p \rightarrow q$ is $q \rightarrow p$.
- The **contrapositive** of $p \rightarrow q$ is $\neg q \rightarrow \neg p$
- The **inverse** of $p \rightarrow q$ is $\neg p \rightarrow \neg q$

Example 8 : What are the contrapositive, the converse, and the inverse of the conditional statement

"The home team wins whenever it is raining?"

Solution:

p: " It is raining" ***q***: "The home team wins"

$p \rightarrow q$, the original statement can be rewritten as
"If it is raining, then the home team wins."
Consequently, the **contrapositive** of this conditional statement is
"If the home team does not win, then it is not raining."
The **converse** is
"If the home team wins, then it is raining."
The **inverse** is
"If it is not raining, then the home team does not win."

Only the contrapositive is equivalent to the original statement.

Example 8 : What are the contrapositive, the converse, and the inverse of the conditional statement

" If it snows, the traffic moves slowly"

Solution:

• p: "it snows" q: "traffic moves slowly".

– The converse:

If the traffic moves slowly then it snows.

• **$q \rightarrow p$**

– The contrapositive:

• If the traffic does not move slowly then it does not snow.

• $\neg q \rightarrow \neg p$

– The inverse:

• If it does not snow the traffic moves quickly.

• $\neg p \rightarrow \neg q$

Definition: **Implication (Conditional Statement)**

Let p and q be propositions. The *biconditional statement* $p \leftrightarrow q$ is the proposition "p if and only if q". Biconditional statements are also called *bi-implications*.

Note: The biconditional statement $p \leftrightarrow q$ is true when p and q have the same truth values, and is false otherwise.

Truth table for Biconditional Statement

p	q	p ↔ q
T	T	T
T	F	F
F	T	F
F	F	T

• Notes:

1. There are some other common ways to express $p \leftrightarrow q$:
   - "p is necessary and sufficient for q"
   - "if p then q, and conversely"

- "p iff q."

2. $p \leftrightarrow q$ means that p and q have the same truth value.
3. this truth table is the exact opposite of $\oplus$'s!
   – $p \leftrightarrow q$ means $\neg(p \oplus q)$
4. $p \leftrightarrow q$ does not imply p and q are true, or cause each other.

Example 9: Let p be the statement "You can take the flight," and let q be the statement

"You buy a ticket."

Then $p \leftrightarrow q$ is the statement

"You can take the flight if and only if you buy a ticket."

Summary of all connectives

Example 10 :

Let p: 2 is a prime ….. **T**

q: 6 is a prime ….. **F**

• Determine **the truth value** of the following statements:

1. $\neg p$ **F**	2. $p \wedge q$ **F**	3. $p \wedge \neg q$ **T**	4. $p \vee q$ **T**
5. $p \oplus q$ **T**	6. $p \rightarrow q$ **F**	7. $q \rightarrow p$ **T**	8. $q \leftrightarrow p$ **F**

Truth Tables of Compound Propositions

We can use these connectives to build up complicated compound propositions.

The following Table displays the precedence levels of the logical operators, $\neg$, $\wedge$, $\vee$, $\rightarrow$, and $\leftrightarrow$

Precedence of Logical Operators.	
Operator	***Precedence***
$\neg$	1
$\wedge$	2
$\vee$	3
$\rightarrow$	4
$\leftrightarrow$	5

Example 11: Construct the truth table of the compound proposition

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

Solution:
Simpler if we decompose the sentence to elementary and intermediate propositions

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

Example 12 : Construct a truth table for
$(p \rightarrow q) \wedge (\neg p \leftrightarrow q)$

Solution:

p	q	¬p	p → q	¬p ↔ q	(p→q)∧ (¬p↔q)
T	T	F	T	F	F
T	F	F	F	T	F
F	T	T	T	T	T
F	F	T	T	F	F

Logic and Bit Operations

- Computers represent information using bits. A **bit** is a symbol with two possible values, namely, 0 (zero) and 1 (one).
- Logical truth values – True and False.
- A bit is sufficient to represent two possible values:
  – 0 (False) or 1(True).

Truth Value	*Bit*
T	1
F	0

- A variable is called a **Boolean variable** if its value is either 1(True) or 0 (False). Consequently, a Boolean variable can be represented using a bit.

- Computer **bit operations** correspond to the logical connectives. By replacing True and False with 1 and 0 in the truth tables for the operators ∧, ∨, and ⊕.
- We will also use the notation *OR*, *AND*, and *XOR* for the operators ∨,∧, and ⊕, respectively as is done in various programming languages.

TABLE 9 Table for the Bit Operators *OR*, *AND*, and *XOR*.

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

- Information is often represented using bit strings, which are lists of zeros and ones.

Definition:

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

Example 13 : 101010011 is a bit string of length nine.

Bitwise operations

- We can extend bit operations to bit strings.
- The **bitwise *OR*(∨)** , **bitwise *AND*(∧)**, and **bitwise *XOR*(⊕)** of two strings of the same length are obtained by taking by taking the *OR*, *AND*, and *XOR* of the corresponding bits, respectively.

Example 13: Find the bitwise *OR*, bitwise *AND*, and bitwise *XOR* of the bit strings 01 1011 0110 and 11 0001 1101.

Solution:

	OR		AND		XOR
	01 1011 0110		01 1011 0110		01 1011 0110
∨	11 0001 1101	∧	11 0001 1101	⊕	11 0001 1101
	11 1011 1111		01 0001 0100		10 1010 1011

1.2 Applications of Propositional Logic

- Logic has many important applications to computer science such as:
  - Inference and reasoning
  - specification of software and hardware
  - design computer circuits
  - construct computer programs
  - verify the correctness of programs,
  - build expert systems.

Translation of English sentences

- English sentences are translated into compound statements to remove the ambiguity.
- Once we have translated sentences from English into logical expressions we can
  - analyze these logical expressions to determine their truth values.
  - manipulate them, and use rules of inference to reason (**Inference and reasoning)** about them.

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

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

Solution:

Parse:

- **If (**You can access the Internet from campus **) then (**you are a computer science major **or** you are not a freshman**)**

Atomic (elementary) propositions:

a: " you can access the Internet from campus"

c: "you are a computer science major "

f: "you are a freshman"

- **Translation:** $a \rightarrow (c \vee \neg f)$.

Example 2: How can this English sentence be translated into a logical expression?

"You cannot ride the roller coaster if you are under 4 feet tall unless you are older than 16 years old."

Solution

Let *q:* "You can ride the roller coaster,"
r : "You are under 4 feet tall,"
and
s: "You are older than 16 years old," .
Then the sentence can be translated to $(r \wedge \neg s) \rightarrow \neg q$.

General rule for translation.
• Look for patterns corresponding to logical connectives in the sentence and use them to define elementary propositions.

Example 3: How can this English sentence be translated into a logical expression?
"You can have free coffee if you are senior citizen and it is a Tuesday"

Solution:
Do the following steps:
1. *find logical connectives*

"You can have free coffee if you are senior citizen and it is a Tuesday"

2. *break the sentence into elementary propositions*

"You can have free coffee if you are senior citizen and it is a Tuesday"

a b c

3. *rewrite the sentence in propositional logic*

$$\mathbf{b \wedge c \rightarrow a}$$

Test your self
Assume two elementary statements:
p:" you drive over 65 mph" ; q:" you get a speeding ticket"
• Translate each of these sentences to logic
1. you do not drive over 65 mph. **(¬p)**
2. you drive over 65 mph, but you don't get a speeding ticket. **(p ∧ ¬q)**
3. you will get a speeding ticket if you drive over 65 mph. **(p → q)**
4. if you do not drive over 65 mph then you will not get a speeding ticket. **(¬p → ¬q)**
5. driving over 65 mph is sufficient for getting a speeding ticket. **(p → q)**

6. you get a speeding ticket, but you do not drive over 65 mph. **(q ∧ ¬p)**

1.3 Propositional Equivalences

Definition:

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*.

Example 1:

- p ∨ ¬p is a **tautology.**

p	¬p	p ∨ ¬p
T	F	T
F	T	T

- p ∧ ¬p is a **contradiction.**

p	¬p	p ∧ ¬p
T	F	F
F	T	F

Logical Equivalences

Definition:

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

- One way to determine whether two compound propositions are equivalent is to use a truth table.
- In particular, the compound propositions p and q are equivalent if and only if the columns giving their truth values agree.

De Morgan's Laws.
$\neg(p \wedge q) \equiv \neg p \vee \neg q$
$\neg(p \vee q) \equiv \neg p \wedge \neg q$

Example 2: (*The 2nd De Morgan's Law*) Show that $\neg(p \vee q)$ and $\neg p \wedge \neg q$ are logically equivalent.

Solution:
To convince us that two propositions are logically equivalent use the truth table

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

Example 3: Show that $p \rightarrow q$ and $\neg p \vee q$ are logically equivalent.

Solution:

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

Because the truth values of $\neg p \vee q$ and $p \rightarrow q$ agree, they are logically equivalent.

Example 4: Show that $p \vee (q \wedge r)$ and $(p \vee q) \wedge (p \vee r)$ are logically equivalent. This is the *distributive law* of disjunction over conjunction.

Solution:

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

Because the truth values of $p \vee (q \wedge r)$ and $(p \vee q) \wedge (p \vee r)$ agree, these compound propositions are logically equivalent.

Important logical equivalences

Equivalence	*Name*
$p \wedge \mathbf{T} \equiv p$ $p \vee \mathbf{F} \equiv p$	Identity laws
$p \vee \mathbf{T} \equiv \mathbf{T}$ $p \wedge \mathbf{F} \equiv \mathbf{F}$	Domination laws
$p \vee p \equiv p$ $p \wedge p \equiv p$	Idempotent laws
$\neg(\neg p) \equiv p$	Double negation law
$p \vee q \equiv q \vee p$ $p \wedge q \equiv q \wedge p$	Commutative laws
$(p \vee q) \vee r \equiv p \vee (q \vee r)$ $(p \wedge q) \wedge r \equiv p \wedge (q \wedge r)$	Associative laws
$p \vee (q \wedge r) \equiv (p \vee q) \wedge (p \vee r)$ $p \wedge (q \vee r) \equiv (p \wedge q) \vee (p \wedge r)$	Distributive laws
$\neg(p \wedge q) \equiv \neg p \vee \neg q$ $\neg(p \vee q) \equiv \neg p \wedge \neg q$	De Morgan's laws
$p \vee (p \wedge q) \equiv p$ $p \wedge (p \vee q) \equiv p$	Absorption laws
$p \vee \neg p \equiv \mathbf{T}$ $p \wedge \neg p \equiv \mathbf{F}$	Negation laws

Some useful equivalences for compound propositions involving conditional statements and biconditional statements.

Logical Equivalences Involving Conditional Statements.
$p \to q \equiv \neg p \lor q$
$p \to q \equiv \neg q \to \neg p$
$p \lor q \equiv \neg p \to q$
$p \land q \equiv \neg(p \to \neg q)$
$\neg(p \to q) \equiv p \land \neg q$
$(p \to q) \land (p \to r) \equiv p \to (q \land r)$
$(p \to r) \land (q \to r) \equiv (p \lor q) \to r$
$(p \to q) \lor (p \to r) \equiv p \to (q \lor r)$
$(p \to r) \lor (q \to r) \equiv (p \land q) \to$

Logical Equivalences Involving Biconditional Statements.
$p \leftrightarrow q \equiv (p \to q) \land (q \to p)$
$p \leftrightarrow q \equiv \neg p \leftrightarrow \neg q$
$p \leftrightarrow q \equiv (p \land q) \lor (\neg p \land \neg q)$
$\neg(p \leftrightarrow q) \equiv p \leftrightarrow \neg q$

Constructing New Logical Equivalences

The logical equivalences can be used to construct additional logical equivalences.

Example 5: Show that $\neg(p \to q)$ and $p \land \neg q$ are logically equivalent.

Solution:

We have the following equivalences.

$\neg(p \to q) \equiv \neg(\neg p \lor q)$ *by Example 22*

$\equiv \neg(\neg p) \land \neg q$ *by the second De Morgan law*

$\equiv p \land \neg q$ *by the double negation law*

Example 6: Show that $\neg(p \vee (\neg p \wedge q))$ and $\neg p \wedge \neg q$ are logically equivalent by developing a series of logical equivalences.

Solution:
(*Note:* we could also easily establish this equivalence using a truth table.)
We have the following equivalences.

$\neg(p \vee (\neg p \wedge q)) \equiv \neg p \wedge \neg(\neg p \wedge q)$ by the second De Morgan law

$\equiv \neg p \wedge [\neg(\neg p) \vee \neg q]$ by the first De Morgan law

$\equiv \neg p \wedge (p \vee \neg q)$ by the double negation law

$\equiv (\neg p \wedge p) \vee (\neg p \wedge \neg q)$ by the second distributive law

$\equiv \mathbf{F} \vee (\neg p \wedge \neg q)$ because $\neg p \wedge p \equiv \mathbf{F}$

$\equiv (\neg p \wedge \neg q) \vee \mathbf{F}$ by the commutative law for disjunction

$\equiv \neg p \wedge \neg q$ by the identity law for **F**

Consequently $\neg(p \vee (\neg p \wedge q))$ and $\neg p \wedge \neg q$ are logically equivalent.

Example 7: Show that $(p \wedge q) \rightarrow (p \vee q)$ is a tautology.

Solution:
We will use logical equivalences to demonstrate that it is logically equivalent to **T**.

$(p \wedge q) \rightarrow (p \vee q) \equiv \neg(p \wedge q) \vee (p \vee q)$ by Example 2

$\equiv (\neg p \vee \neg q) \vee (p \vee q)$ by the first De Morgan law

$\equiv (\neg p \vee p) \vee (\neg q \vee q)$ by the associative and commutative laws for disjunction

$\equiv \mathbf{T} \vee \mathbf{T}$ by Example 1 and the commutative law for disjunction

$\equiv \mathbf{T}$ by the domination law