Implication

Definition: Let p and q be propositions. The proposition "p implies q" denoted by $p \to q$ is called implication. It is false when p is true and q is false and it is true otherwise.

In $p \to q$, p is called hypothesis and q is called the conclusion.

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

$p \to q$ is read in a variety of equivalent ways.

* if p then q
* p only if q
* p is sufficient for q
* q whenever p.

Examples:

① If today is holiday then 2 is prime
If F then T
What is the truth value ?
Ans: T.

②. If you live in Karachi then Karachi is capital of Pakistan
If T then F.
What is the truth value?
Ans: F.

③ If your major is languistics then you learn literature.
If F then F
What is the truth value?
Ans: T.

Inverse, Converse and Contrapositive.

The inverse of $p \rightarrow q$ is. $\neg p \rightarrow \neg q$

The converse of $p \rightarrow q$ is $q \rightarrow p$.

The contrapositive of $p \rightarrow q$ is. $\neg q \rightarrow \neg p$.

Examples:

If it rains, then traffic moves slowly.

If p then q $\qquad p \rightarrow q$.

p: It rains.

q: Traffic moves slowly.

The converse: $q \rightarrow p$.

If the traffic moves slowly, then it rains

The inverse: $\neg p \rightarrow \neg q$

If it does not rain, then traffic does not move slowly.

The contrapositive: $\neg q \rightarrow \neg p$.

If traffic does not move slowly, then it does not rain.

BICONDITIONALS

Definition: Let p and q be propositions. The bidirectional $p \leftrightarrow q$, read as, "p if and only if q", is true when p and q have the same truth values and is false otherwise.

- p is necessary & sufficient for q
- if p then q, & conversely
- p iff q

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

Observation: ① negation of $p \oplus q$
② bidirectionals are not always explicit in natural language. Bidirectionals are often expressed using an "if, then" or an "only if" construction instead of "if and only if."

Example: Constructing a truth table for compound propositions.

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

p	q
T	T.
T	F
F	T.
F	F.

No. of rows = ? Columns = ?

Example: (Refresher).

p: 2 is prime .... T

q: 6 is prime .... F.

Determine the truth values of the following statements.

$\neg p$:

$p \wedge q$:

$p \wedge \neg q$:

$p \vee q$:

$p \oplus q$:

$p \to q$:

$q \to p$:

PRECEDENCE OF LOGICAL OPERATORS.

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

COMPUTER REPRESENTATION OF TRUE AND FALSE.

- Computers represent data and programs using 0s and 1s.
- A bit is sufficient to represent two logical truth values i.e True & False.

  0 ≅ False.

  1 ≅ True.

- A variable that takes on values 0 or 1 is called boolean variable.

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

p	q	p $\vee$ q	p$\wedge$q.
1	1	1	1
1	0	1	0
0	1	1	0
0	0	0	0.

p	$\neg$p.
1	0
0	1

BITWISE OPERATIONS:

Examples:

$$\begin{array}{r} 10\ 11\ 00\ 11 \\ \vee\ 01\ 10\ 10\ 10 \\ \hline 11\ 11\ 10\ 11 \end{array} \qquad \begin{array}{r} 1011\ 0011 \\ \wedge\ 0110\ 1010 \\ \hline 0010\ 0010 \end{array} \qquad \begin{array}{r} 1011\ 0011 \\ \oplus\ 0110\ 1010 \\ \hline 1101\ 1001. \end{array}$$

1.2 APPLICATIONS OF PROPOSITIONAL LOGIC

Logic has many applications to mathematics, computer science and othe disciplines.

* Translating natural language sentences: statements in natural language such as English are often imprecise or ambiguous. To make such statements precise, they can be translated into the language of logic.

* Specification of software & hardware → in order to make them precise before development begins.

* To design computer circuits.

* To construct computer programs, to verify correctness of computer programs.

* To build expert systems.

* To analyze and solve many familiar puzzles.

* Software system based on the rules of logic have been developed for constructing proofs automatically.

Translating English Sentences: Application #1.

Steps to convert an English sentence into a statement in propositional logic:

1. Identify simple propositions and represent using propositional variables.

2. Determine appropriate logical connectives.

Example:

If you are older than 13 or you are with your parents then you can attend PG-13 movie.

Parse

If (you are older than 13 or you are with your parents) then (you can attend a PG-13 movie).

Attomic (Elementary) propositions:

A: you are older than 13.
B: you are with your parents.
C: you can attend a PG-13 movie.

Translation: $A \lor B \rightarrow C$

Example 2:

You can have a free coffee if you are a senior citizen ~~or~~ and it is a Tuesday.

Solution: in class.

Elementary / Atomic propositions? A B C.
Logical connectives ? if and.

Translation:

$b \land c \rightarrow a$.

System Specifications

Application #2.

Systems and software engineers gather requirements in natural language and produce precise and unambiguous specifications that can be used as the basis for system development.

Example: Express the following specification.
"The automated reply cannot be sent when the file system is full".

Solution:

p: The automated reply can be sent.
q: The file system is full.

$\neg p$: The automated reply cannot be sent.

Translation: $q \rightarrow \neg p$.

Example: Determine whether these system specifications are consistent.

① "The diagnostic message is stored in 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"

Solution: To determine whether these specifications are consistent, first express them using logical expressions.

Let

p: The diagnostic message is stored in the buffer.
q: The diagnostic message is retransmitted.
Then the specifications can be written as.

$p \vee q$, $\quad \neg p$ & $p \rightarrow q$.

reach. same conclusion by truth table

Because we want $p \vee q$ true but p must be false, q must be true.
$p \rightarrow q$ is true when p is false & q is true $\Rightarrow$ conclusion ....

Logic Puzzles

Application #3.

Puzzles that can be solved using logical reasoning are called —.

Example:

An island has two types of inhabitants,

knights, who always tell the truth. and knaves, who always lie.

You encounter two people A and B.
What are A and B, if A says "B is knight" and B says "Two of us are opposite types".

Solution:

p: A is knight. so $\neg p$: A is knave.
q: B is knight. so $\neg q$ B is knave.

We first consider A is knight (i.e consider p is true).

If A is knight and as knight always tells truth, means. A says B is knight is true (i.e q is true).

However, if B is knight, then B's statement that A and B are of opposite type (i.e $(p \wedge \neg q) \vee (\neg p \vee q)$), would have to be true, which is not.

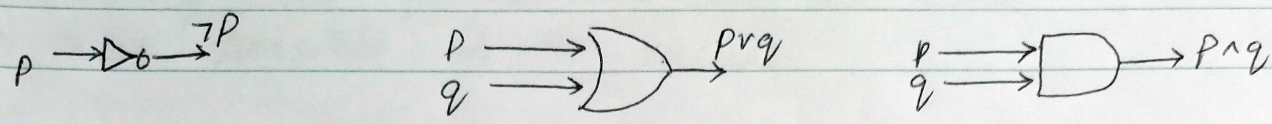
Consequently we conclude A is not knight, that is, p is false.

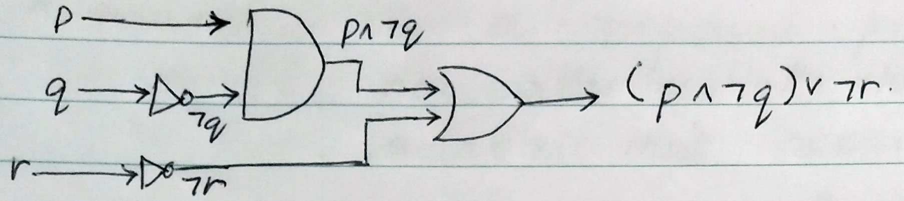
A's statement, B is knight, i.e. q is true, is a lie. It means q is false and B is knave.

Furthermore, if B is knave, then B's statement that A & B are opposite type is a lie, which is consistent —.
We conclude both A & B are knaves.

Example 8 from Book in class.

Logic Circuits   Application #4.





Example: Build a digital logic circuit that produces the output $(p \lor \neg r) \land (\neg p \lor (q \lor \neg r))$.

Other Applications:

— The field of AI.
- Build programs that act intellegently.
- Programs often rely on symbolic manipulations.

— Expert Systems.   Example: MYCIN
- Encode knowledge about the world in logic.
- Support inferences where new facts are infered from existing facts following the sementics of logic.

— Theorem provers.
- Encode existing knowledge (about math) using logic.
- Show that some hypothesis are true.

— Diagnosis of faults in Electrical Systems

Tautologies, Contradictions and Contigencies

Compound propositions can be classified according to their possible truth values.

A tautology is a compound proposition that is always true, no matter what the truth values of propositional variables that occur in it.

Example: $p \vee \neg p$.

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

A contradiction is a compound proposition that is always false, no matter what the truth values are of propositional variables that occur in it.

Example $p \wedge \neg p$.

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

A proposition that is neither tautology nor contradiction is called contigency.

Propositional Equivalence / Logical Equivalences.

Equivalent propositions:

- Two propositions are equivalent if they always have same truth value.

Two compound propositions are logically equivalent if $p \leftrightarrow q$ is a tautology.

* We write this as $p \Leftrightarrow q$ or $p \equiv q$.

Two compound propositions p and q are equivalent if and only if the columns in the truth table giving their truth values agree.

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

$(\neg p \vee q)$

From above: observe that.

$$p \to q \equiv \neg q \to \neg p \quad \{\text{contrapositive}\}$$

$$q \to p \equiv \neg p \to \neg q$$

$$p \to q \equiv \neg p \vee q$$

In construction of mathetical arguments, an important step is the replacement of a statement with another statement with the same truth value (equivalent statement).

De Morgan's Laws.

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

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

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

Example: Show that $(p \vee (q \wedge r)) \equiv (p \vee q) \wedge (p \vee r)$.

This is distributive law of disjunction over conjuction.

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

no: of rows =?

Key Logical Equivalences:

Identity Laws. $p \wedge T \equiv p$, $p \wedge F \equiv p$.

Domination Laws.

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

Idempotent Laws: $p \vee p \equiv p$ $p \wedge p \equiv p$.

Double Negation Law: $\neg(\neg p) \equiv p$.

Negation Laws : $p \vee \neg p \equiv T$, $p \wedge \neg p \equiv F$.

Commutative Laws: $p \vee q \equiv q \vee p$, $p \wedge q \equiv q \wedge p$

Associative Laws: $(p \wedge q) \wedge r \equiv p \wedge (q \wedge r)$
$(p \vee q) \vee r \equiv p \vee (q \vee r)$

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

Absorption Laws: $p \vee (p \wedge q) \equiv p$.
$p \wedge (p \vee q) \equiv p$.

More Logical Equivalences:

Table#7 & Table#8 in the book.