Logic & Proofs (Lecture – 1)

Dr. Nirnay Ghosh

Formal Definition: Proposition

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

"It is not the case that p."

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

- Michael's PC runs Linux
  - It is not the case that Michael's PC runs Linux
  - Michael's PC does not run Linux
- Vandana's smartphone has at least 32GB of memory
  - It is not the case that Vandana's smartphone has at least 32GB of memory
  - Vandana's smartphone does not have at least 32GB of memory
  - Vandana's smartphone has less than 32GB of memory

Truth Tables & Connectives

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

p	$\neg p$
T	F
F	T

- Each row shows the truth value of $\neg p$ corresponding to the truth value of p for that row

- **Connectives**: logical operators that are used to form new propositions from two or more existing propositions

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

Truth Tables & Connectives

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

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

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

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

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

TABLE 4 The Truth Table for the Exclusive Or of Two Propositions.

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

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.

Conditional Statements

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

TABLE 5 The Truth Table for the Conditional Statement $p \to q$.

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

"if p, then q"
"if p, q"
"p is sufficient for q"
"q if p"
"q when p"
"a necessary condition for p is q"
"q unless $\neg p$"
"p implies q"
"p only if q"
"a sufficient condition for q is p"
"q whenever p"
"q is necessary for p"
"q follows from p"

Let p be the statement "Maria learns discrete mathematics" and q the statement "Maria will find a good job." Express the statement $p \to q$ as a statement in English.

Contrapositive, Converse & Inverse

- Given conditional statement: $p \rightarrow q$
  - **Contrapositive**: $\neg q \rightarrow \neg p$
  - **Converse**: $q \rightarrow p$
  - **Inverse**: $\neg p \rightarrow \neg q$
- *Construct truth tables for converse, contrapositive, and inverse of a conditional statement. What do you observe?*
- Statement: "The home team wins whenever it is raining"
  - $\boldsymbol{p \rightarrow q}$: "If it is raining, then the home team wins" (conditional)
  - $\boldsymbol{\neg q \rightarrow \neg p}$: "If the home team does not win, then it is not raining" (contrapositive)
  - $\boldsymbol{q \rightarrow p}$: "If the home team wins, then it is raining" (converse)
  - $\boldsymbol{\neg p \rightarrow \neg q}$: "If it is not raining, then the home team does not win" (inverse)

Contrapositive, Converse & Inverse

Contrapositive Truth Table

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

Converse Truth Table

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

Inverse Truth Table

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

Biconditional Statement

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

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

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

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

- Find out the truth table for: $(p \rightarrow q) \wedge (q \rightarrow p)$.

Truth Table for Compound Propositions

- Five basic logical connectives: *conjunction, disjunction, negation, conditional statements, biconditional statements*
- Three derived logical connectives: *contrapositive, converse, inverse*
- These can be used to build up complicated propositions involving any number of propositional variables
- Construct truth table for: $(p \vee \neg q) \rightarrow (p \wedge q)$

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

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

Precedence of Logical Operators

TABLE 8 Precedence of Logical Operators.

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

- Logic bit operators:
  - Information is represented in form of **bits**
  - Symbol with two possible values: 0 (zero) and 1 (one).
  - It can also be used to represent truth values: *true* and *false*
  - In practice, 1 represents T (true) and 0 represents F (false)
  - **Boolean variable**: value is either true or false

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

Application of Propositional Logic

- Statements in mathematics and the sciences and in natural language often are imprecise or ambiguous.
- To make such statements precise, they can be translated into the language of logic
- Examples:
  - Specification of software and hardware: need precise specification before the actual development phase
  - Design of computer circuits, to construct computer programs, to verify correctness of programs, and so on.

Application of Propositional Logic

- We will look into the following application domains:
  - Translating English Sentences
  - System Specifications
  - Boolean Searches
  - Logic Puzzles
  - Logic Circuits

Logic Circuit

- Logic circuit (or digital circuit) receives input signals $p_1, p_2, \ldots, p_n$ each a bit [either 0 (off) or 1 (on)), and produces output signals $s_1, s_2, \ldots, s_n$, each a bit.

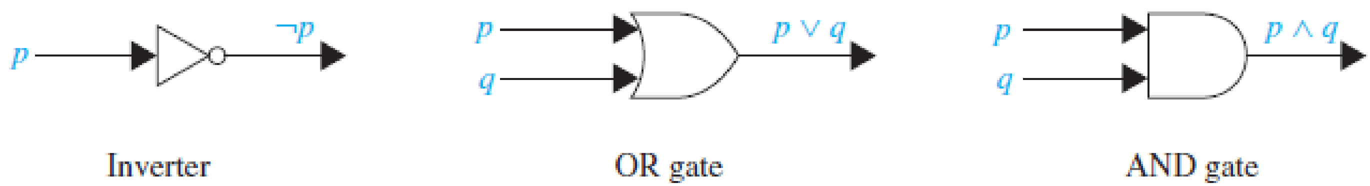

FIGURE 1 **Basic logic gates.**

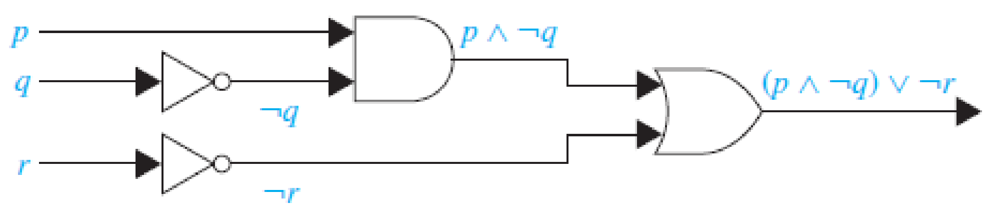

FIGURE 2 **A combinatorial circuit.**

Logic Circuit

- Build a logic circuit for: $(p \vee \neg r) \wedge (\neg p \vee (q \vee \neg r))$