The foundations of Logic and Proofs

Logic

- Crucial for mathematical reasoning
- Used for designing electronic circuitry

- Logic is a system based on propositions.
- A proposition is a statement that is either **true** or **false** (not both).
- We say that the truth value of a proposition is either true (T) or false (F).

- Corresponds to 1 and 0 in digital circuits

The Statement/Proposition Game

- “Elephants are bigger than mice.”

Is this a statement?	yes
Is this a proposition?	yes
What is the truth value of the proposition?	true

The Statement/Proposition Game

- “520 < 111”

Is this a statement? yes

Is this a proposition? yes

What is the truth value of the proposition? false

The Statement/Proposition Game

- “$y > 5$”

Is this a statement? yes

Is this a proposition? no

Its truth value depends on the value of y, but this value is not specified.

We call this type of statement a propositional function or open sentence.

The Statement/Proposition Game

- “Today is January 1 and $99 < 5$.”

Is this a statement? yes

Is this a proposition? yes

What is the truth value of the proposition? false

The Statement/Proposition Game

- “Please do not fall asleep.”

Is this a statement? no

It’s a request.

Is this a proposition? no

Only statements can be propositions.

The Statement/Proposition Game

- “If elephants were red,
- they could hide in cherry trees.”

Is this a statement? yes

Is this a proposition? yes

What is the truth value of the proposition? probably false

The Statement/Proposition Game

- "x < y if and only if y > x."

Is this a statement? yes

Is this a proposition? yes

… because its truth value does not depend on specific values of x and y.

What is the truth value of the proposition? true

Combining Propositions

- As we have seen in the previous examples, one or more propositions can be combined to form a single **compound proposition**.

- We formalize this by denoting propositions with letters such as p, q, r, s, and introducing several logical operators.

Logical Operators (Connectives)

- We will examine the following logical operators:

- Negation (NOT)
- Conjunction (AND)
- Disjunction (OR)
- Exclusive or (XOR)
- Implication (if – then)
- Biconditional (if and only if)

- Truth tables can be used to show how these operators can combine propositions to compound propositions.

Negation (NOT)

- Unary Operator, Symbol: $\neg$

P	$\neg P$
true (T)	false (F)
false (F)	true (T)

the **truth table** for the negation

Conjunction (AND)

- Binary Operator, Symbol: $\wedge$

The Truth Table

P	Q	P∧Q
T	T	T
T	F	F
F	T	F
F	F	F

Definition

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

Disjunction (OR)

- Binary Operator, Symbol: $\vee$

The Truth Table

P	Q	P∨Q
T	T	T
T	F	T
F	T	T
F	F	F

Definition

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

Exclusive Or (XOR)

- Binary Operator, Symbol: $\oplus$

The Truth Table

P	Q	P$\oplus$Q
T	T	F
T	F	T
F	T	T
F	F	F

Definition

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

Implication (if - then)
(Also called Conditional statement)

- Binary Operator, Symbol: $\rightarrow$

The Truth Table

P	Q	P→Q
T	T	T
T	F	F
F	T	T
F	F	T

Definition

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. In the conditional statement p → q, p is called the hypothesis (or antecedent or premise) and q is called the conclusion (or consequence).

Biconditional (if and only if)

- Binary Operator, Symbol: $\leftrightarrow$

The Truth Table

P	Q	P↔Q
T	T	T
T	F	F
F	T	F
F	F	T

Definition

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

Statements and Operators

- Statements and operators can be combined in any way to form new statements.

P	Q	$\neg P$	$\neg Q$	$(\neg P)\vee(\neg Q)$
T	T	F	F	F
T	F	F	T	T
F	T	T	F	T
F	F	T	T	T

The Truth Table

Truth Tables of Compound Propositions

- Statements and operators can be combined in any way to form new statements.

P	Q	$P \wedge Q$	$\neg (P \wedge Q)$	$(\neg P) \vee (\neg Q)$
T	T	T	F	F
T	F	F	T	T
F	T	F	T	T
F	F	F	T	T

The Truth Table

Examples:

- Construct the truth table of the compound proposition
  $(p \vee \neg q) \rightarrow (p \wedge q)$.

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

Example:

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

TABLE 8
Precedence of Logical Operators.

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

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.
  - bit can be used to represent a truth value, because there are two truth values, namely, *true* and *false*.

Truth Value	*Bit*
T	1
F	0

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

Logic and Bit Operations

- 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 : 101010011 is a bit string of length nine
  - The bitwise OR, bitwise AND, and bitwise XOR of these strings are obtained by taking the OR, AND, and XOR of the corresponding bits, respectively

bitwise OR	bitwise AND	bitwise 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**

Equivalent Statements

P	Q	$\neg(P \wedge Q)$	$(\neg P) \vee (\neg Q)$	$\neg(P \wedge Q) \leftrightarrow (\neg P) \vee (\neg Q)$
T	T	F	F	T
T	F	T	T	T
F	T	T	T	T
F	F	T	T	T

- The statements $\neg(P \wedge Q)$ and $(\neg P) \vee (\neg Q)$ are logically equivalent, since $\neg(P \wedge Q) \leftrightarrow (\neg P) \vee (\neg Q)$ is always true.

Tautologies and Contradictions

- **Definitions:**
  - A compound proposition that is always true, no matter what the truth values of the propositional variables that occur in it, is called a <u>tautology</u>.
  - A compound proposition that is always false is called a <u>contradiction</u>.
  - A compound proposition that is neither a tautology nor a contradiction is called a <u>contingency</u>.
  - The compound propositions p and q are called <u>logically equivalent</u> if $p \leftrightarrow q$ is a tautology. The notation $p \equiv q$ denotes that p and q are logically equivalent.
- Tautology Examples:
  - $R \vee (\neg R)$
  - $\neg(P \wedge Q) \leftrightarrow (\neg P) \vee (\neg Q)$

  - If $S \rightarrow T$ is a tautology, we write $S \Rightarrow T$.
  - If $S \leftrightarrow T$ is a tautology, we write $S \Leftrightarrow T$.

Tautologies and Contradictions

- Contradictions Examples:
  - $R \wedge (\neg R)$
  - $\neg(\neg(P \wedge Q) \leftrightarrow (\neg P) \vee (\neg Q))$

TABLE 1 Examples of a Tautology and a Contradiction.

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

- The negation of any tautology is a contradiction, and the negation of any contradiction is a tautology.

TABLE 2 De Morgan's Laws.

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

Examples

- We already know the following tautology:

$$\neg(P \wedge Q) \Leftrightarrow (\neg P) \vee (\neg Q)$$

- Show that $\neg(P \vee Q) \Leftrightarrow (\neg P) \wedge (\neg Q)$.

TABLE 3 **Truth Tables for $\neg(p \vee q)$ and $\neg p \wedge \neg q$.**

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

Examples

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

TABLE 4 **Truth Tables for $\neg p \vee q$ and $p \rightarrow q$.**

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

Logical Equivalences

Properties of logical connectives (Logical Identities)

$p \wedge \mathbf{T} \equiv p$ $p \vee \mathbf{F} \equiv p$	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

Examples

- Show that $\neg(p \vee (\neg p \wedge q))$ and $\neg p \wedge \neg q$ are logically equivalent
  - We could use a truth table to show that these compound propositions are equivalent
  - However, we want to illustrate how to use logical identities

$$
\begin{aligned}
\neg(p \vee (\neg p \wedge q)) &\equiv \neg p \wedge \neg(\neg p \wedge q) && \text{by the second De Morgan law} \\
&\equiv \neg p \wedge [\neg(\neg p) \vee \neg q] && \text{by the first De Morgan law} \\
&\equiv \neg p \wedge (p \vee \neg q) && \text{by the double negation law} \\
&\equiv (\neg p \wedge p) \vee (\neg p \wedge \neg q) && \text{by the second distributive law} \\
&\equiv \mathbf{F} \vee (\neg p \wedge \neg q) && \text{because } \neg p \wedge p \equiv \mathbf{F} \\
&\equiv (\neg p \wedge \neg q) \vee \mathbf{F} && \text{by the commutative law for disjunction} \\
&\equiv \neg p \wedge \neg q && \text{by the identity law for } \mathbf{F}
\end{aligned}
$$

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

Examples

- Show that $(p \land q) \rightarrow (p \lor q)$ is a tautology

$$
\begin{aligned}
(p \land q) \rightarrow (p \lor q) &\equiv \neg(p \land q) \lor (p \lor q) && \text{by Example 3} \\
&\equiv (\neg p \lor \neg q) \lor (p \lor q) && \text{by the first De Morgan law} \\
&\equiv (\neg p \lor p) \lor (\neg q \lor q) && \text{by the associative and commutative} \\
& && \quad \text{laws for disjunction} \\
&\equiv \mathbf{T} \lor \mathbf{T} && \text{by Example 1 and the commutative} \\
& && \quad \text{law for disjunction} \\
&\equiv \mathbf{T} && \text{by the domination law}
\end{aligned}
$$

Predicates and Quantification

Predicates

- statement involving one or more variables,

- e.g.: $x-3 > 5$.

- Let us call this propositional function $P(x)$, where P is the predicate and x is the variable.

What is the truth value of $P(2)$? false

What is the truth value of $P(8)$? false

What is the truth value of $P(9)$? true

Predicates

- Let us consider the propositional function $Q(x, y, z)$ defined as:

- $x + y = z$.

- Here, Q is the predicate and x, y, and z are the variables.

What is the truth value of $Q(2, 3, 5)$? true

What is the truth value of $Q(0, 1, 2)$? false

What is the truth value of $Q(9, -9, 0)$? true

Predicates

- Let $Q(x, y)$ denote the statement "$x = y + 3$." What are the truth values of the propositions
- $Q(1, 2)$ and $Q(3, 0)$ defined as:

- Here, Q is the predicate and x, y are the variables.

What is the truth value of $Q(3, 0)$?	true
What is the truth value of $Q(1, 2)$?	false
What is the truth value of $Q(5, 2)$?	true

Universal Quantification

- Let $P(x)$ be a propositional function.

- **Universally quantified sentence:**
- For all x in the universe of discourse $P(x)$ is true.

- Using the universal quantifier $\forall$:
- $\forall x\, P(x)$ "for all x $P(x)$" or "for every x $P(x)$"

- (Note: $\forall x\, P(x)$ is either true or false, so it is a proposition, not a propositional function.)

Universal Quantification

- Example:
- S(x): x is a UMBC student.
- G(x): x is a genius.

- What does $\forall x\ (S(x) \rightarrow G(x))$ mean ?

- "If x is a UMBC student, then x is a genius."
- or
- "All UMBC students are geniuses."

Existential Quantification

- **Existentially quantified sentence:**
- There exists an x in the universe of discourse for which P(x) is true.

- Using the existential quantifier ∃:
- $\exists x\, P(x)$ “There is an x such that P(x).”
- “There is at least one x such that P(x).”

- (Note: $\exists x\, P(x)$ is either true or false, so it is a proposition, but no propositional function.)

Existential Quantification

- Example:
- P(x): x is a UMBC professor.
- G(x): x is a genius.

- What does $\exists x\ (P(x) \wedge G(x))$ mean ?

- "There is an x such that x is a UMBC professor and x is a genius."
- or
- "At least one UMBC professor is a genius."

Quantification

- Another example:
- Let the universe of discourse be the real numbers.

- What does $\forall x \exists y\ (x + y = 320)$ mean ?

- "For every x there exists a y so that $x + y = 320$."

Is it true? yes

Is it true for the natural numbers? no

Disproof by Counterexample

- A counterexample to $\forall x\, P(x)$ is an object c so that P(c) is false.

- Statements such as $\forall x\, (P(x) \rightarrow Q(x))$ can be disproved by simply providing a counterexample.

Statement: "All birds can fly."

Disproved by counterexample: Penguin.

Negation

- $\neg(\forall x\, P(x))$ is logically equivalent to $\exists x\, (\neg P(x))$.
- $\neg(\exists x\, P(x))$ is logically equivalent to $\forall x\, (\neg P(x))$.
- See Table 3 in Section 1.3.
- I recommend exercises 5 and 9 in Section 1.3.