Section 1.2 & 1.3

1.2: Applications of Propositional Logic

For problems in this section, the idea is to look for "logical fallacies." We can assign a truth value to a proposition and check if the propositions that follow make logical sense (for logic puzzles). Alternatively, we can look at the given propositions and "reverse engineer" these propositions to determine the truth value of each component to check for consistency.

Important words/definitions to know:

- Consistent – should not contain conflicting requirements that could be used to derive a contradiction
- Boolean searches
  - connective AND is used to match records that contain both of two search terms
  - connective OR is used to match one or both of two search terms,
  - connective NOT (sometimes written as AND NOT) is used to exclude a particular search term
  - Ex.: Web page search
- Logic puzzles – Puzzles that can be solved using logical reasoning

Example (from textbook) Determine whether these system specifications are consistent:

- "The diagnostic message is stored in the buffer or it is retransmitted" $p \vee q$
- "The diagnostic message is not stored in the buffer" $\neg p$
- "If the diagnostic message is stored in the buffer, then it is retransmitted" $p \rightarrow q$
- "The diagnostic message is not retransmitted" $\neg q$

p: The diagnostic message is stored in the buffer
q: It is transmitted

Assume $\neg q$ is T, then q is F.
Assume $\neg p$ is T, then p is F.

If both p, q are F, then we get:
$p \rightarrow q$: $F \rightarrow F$ is T
$p \vee q$: $F \vee F$ is F

Which we get: Inconsistent

Example When three friends are seated in a restaurant, the hostess asks them: "Does everyone want coffee?" The first friend says, "I don't know." The second friend says "I don't know." The last friend says "No, not everyone wants coffee." The hostess comes back and gives coffee to the group of friends that wanted coffee. How did she figure this out?

The first friend(A) wants coffee. If A doesn't want, A must have said "NO". Similarly the second friend(B) only knows that A wants coffee, but doesn't know about the third one and as he doesn't say "NO" so he wants coffee. But the third one(C) says "NO" after knowing that the first(A) and second(B) friend wants coffee. So, the third friend doesn't want coffee.

1.3: Propositional Equivalences

Important words/definitions to know:

- Tautology - A compound proposition that is always true.
  - No matter what truth value is has
- Contradiction - A compound proposition that is always false.
- Contingency - A compound proposition that is neither a tautology nor a contradiction.
- Example:
  - $p \vee \neg p$: always true, tautology
  - $p \wedge \neg p$: always false, contradiction

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

- logically equivalent - if $p \leftrightarrow q$ is a tautology
  - denoted: $p \equiv q$
- DeMorgan laws
  - Named after an English mathematician

TABLE 2 De Morgan's Laws.

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

- Logical Equivalences

Identity	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

Logical Equivlances Involving Conditional Statements
$p \to q \equiv \neg p \vee q$
$p \to q \equiv \neg q \to \neg p$
$p \vee q \equiv \neg p \to q$
$p \wedge q \equiv \neg(p \to \neg q)$
$\neg(p \to q) \equiv q \wedge \neg q$
$(p \to q) \wedge (p \to r) \equiv p \to (q \wedge r)$
$(p \to r) \wedge (q \to r) \equiv (p \vee q) \to r$
$(p \to q) \vee (p \to r) \equiv p \to (q \vee r)$
$(p \to r) \vee (q \to r) \equiv (p \wedge q) \to r$

Logical Equivalences Involving Biconditional Statements
$p \leftrightarrow q \equiv (p \to q) \wedge (q \to p)$
$p \leftrightarrow q \equiv \neg p \leftrightarrow \neg q$
$p \leftrightarrow q \equiv (p \wedge q) \vee (\neg p \wedge \neg q)$
$\neg(p \leftrightarrow q) \equiv p \leftrightarrow \neg q$

Example Show the logical equivalence without using a truth table: $\neg(p \vee (\neg p \wedge q)) \equiv \neg p \wedge \neg q$.

$\neg(p \vee (\neg p \wedge q))$	$\equiv \neg p \wedge \neg(\neg p \wedge q)$	DeMorgan's Law
	$\equiv \neg p \wedge [\neg(\neg p) \vee \neg q]$	DeMorgan's Law
	$\equiv \neg p \wedge [p \vee \neg q]$	Double Negation
	$\equiv (\neg p \wedge p) \vee (\neg p \wedge \neg q)$	Distributive
	$\equiv F \vee (\neg p \wedge \neg q)$	Negation
	$\equiv \neg p \wedge \neg q$	Identity law