1 Propositional Calculus

Definition 1 *All expressions containing identifers that represent expressions are called* ***schemas****.*

Definition 2 *Literals take the form Q or $\neg Q$.*

1.1 Propositions

Propositions are any statement that is either **true** or **false**. Neither commands nor Questions are propositions. This is also known as a dichotomy.

Definition 3 *A proposition that contains a single propositional constant or a single propositional variable is called an* ***atomic*** *proposition.*

1.1.1 Logical Connectives

Negation $\neg$
Conjuction $\wedge$
Disjunction $\vee$
Conditional $\Rightarrow$ (In logical arguments the "if ... then" construct expresses the **conditional**.)
Equivalence $\Longleftrightarrow$

1.1.2 The symbols $\models, \Rrightarrow$ are used in Logical Implication

Given some logical expression X, and conclusion y, if $X \Rightarrow y$ is a tautology, then X logically implies y. This is written as $X \Rrightarrow y$. $X' \models y$ is a special form of logical implication where X' is a set of premises. This simply means that the **conjuction** of all the premises **logically implies** the conclusion. To show $X \models y$, **show that** $x_1 \wedge ... \wedge x_n \Rightarrow y$ **is a tautology**.

$\models A$ litterally means that A is a tautology. Consider that the conjunction of anything logically implies A. Since $\models$ is a symbol that says something about a proposition, we conclude that $\models$ is not part of propositional calculus.(p25)[1]

For "sound reasoning" we often write our premises P and our conclusions C using the following notation

$$\begin{aligned} P &\models C \\ p_1, ..., p_n &\models C \end{aligned}$$

p43 [1]

Definition 4 *Let P and Q be two propositions. Then $P \Rightarrow Q$ is called the conditional of P and Q. The conditional $P \Rightarrow Q$. may be translated into English by using the "If ... Then" construct, as in "If P, then Q."*

P Q	$P \Rightarrow Q$
T T	T
T F	F
F T	T
F F	T

P is the **antecedent** and Q is the **consequent**.

Definition 5 *Fully parenthesized expressions:*

1. *Every atomic expression is an FPE*
2. *If A is an FPE, so is $(\neg A)$.*
3. *If A and B are FPEs, so are $(A \wedge B), (A \vee B), (A \Rightarrow B)$ and $(A \Longleftrightarrow B)$.*

4. No other expression is an FPE.

Definition 6 *Subexpressions of an expression E are defined as follows:*

1. *E is a subexpression of E.*
2. *If E is of the form $(\neg A)$, then A is a subexpression of E.*
3. *If E is of the form $(A \wedge B), (A \vee B), (A \Rightarrow B)$, or $(A \iff B)$, then A and B are both subexpressions of E. These subexpression are called* ***immediate subexpressions****.*
4. *If A is a subexpression of E and if C is a subexpression of A, then C is a subexpression of E.*
5. *No other expression is a subexpression of E.*

Definition 7 *Precedence Rules: $\neg, \wedge, \vee, \Rightarrow, \iff$. These rules are not always enough to determine the order. For example: $P \Rightarrow Q \Rightarrow R$. So we must define left and right associative.*

Definition 8 *A binary operator is called* ***left associative*** *if the operator on the left has precedence over the operator on the right. A binary operator is called* ***right associative*** *if the operator on the right has precedence over the operator on the left.* ***All binary logical connectives are left associative.***

Definition 9 *A logical expression is a* ***tautology*** *if it is true under all possible assignments.*

Definition 10 *A logical expression is a* ***contradiction*** *if it is false under all posssible assignments.*

Definition 11 *Given an logical expression E, we define $P : E$ is a tautology, $Q : E$ is a contradiction, $R : E$ is a contingent. We assert that the definition of contingent is $\neg(P \vee Q) \Rightarrow R$*

Definition 12 *If A and B are two logical eqpressions and if A and B always have the same truth value, then A and B are said to be* ***logically equivalent****, and we write them as $A \equiv B$. In other words, $A \equiv B$ iff $A \iff B$ is a tautology. Note we could write this as $A \equiv B$ iff $\models A \iff B$.*

1.2 Reduction Laws

Reducing implication and equivalence

$$P \Rightarrow Q \equiv \neg P \vee Q$$
$$P \iff Q \equiv (P \wedge Q) \vee (\neg P \wedge \neg Q)$$
$$P \iff Q \equiv (P \Rightarrow Q) \wedge (Q \Rightarrow P)$$
$$\equiv (\neg P \vee Q) \wedge (\neg Q \vee P)$$

Laws of Conjunction, disjuction and Negation

Law	Law Name
$P \vee \neg P \equiv T$	Excluded middle
$P \wedge \neg P \equiv F$	Contradiction
$P \vee F \equiv P$	Identity
$P \wedge T \equiv P$	identity
$P \vee P \equiv P$	Idempotent
$P \wedge P \equiv P$	Idempotent
$\neg(\neg P) \equiv P$	Double negation
$P \vee Q \equiv Q \vee P$	Commutative
$P \wedge Q \equiv Q \wedge P$	Commutative
$(P \vee Q) \vee R \equiv P \vee (Q \vee R)$	Associative
$(P \wedge Q) \wedge R \equiv P \wedge (Q \wedge R)$	Associative
$(P \vee Q) \wedge (P \vee R) \equiv P \vee (Q \wedge R)$	Distributive
$(P \wedge Q) \vee (P \wedge R) \equiv P \wedge (Q \vee R)$	Distributive
$\neg(P \wedge Q) \equiv \neg P \vee \neg Q$	DeMorgan's
$\neg(P \vee Q) \equiv \neg P \wedge \neg Q$	DeMorgan's

1.3 Normal Forms

Definition 13 *A logical expression is said to be in **disjuctive normal form (CNF)** if it is written as a disjunction, in which all terms are conjunctions of literals.*

Definition 14 *A logical expression is said to be in **conjunctive normal form (DNF)** if it is written as a conjunction of disjunctions of literals.*

Notice that this precludes the disjunction $\neg(P \wedge Q) \vee R$ from being in DNF because it contains a negated nonatomic subexpression. Only atomic subexpressions can be negated **because negated atomic expressions are literals**. Also $P \wedge (R \vee (P \wedge Q))$ is not in CNF because of the subexpression $(P \vee (P \wedge Q))$.

1.3.1 Three steps to conjunctive normal form

1. Remove all $\Rightarrow$ and $\Longleftrightarrow$.

2. If the expression in question contains any negated compound subexpressions, either remove the negation by using the double-negation law or use De Morgan's laws to reduce the scope of negation.

3. Once an expression with no negated compound subexpression is found, use the following two laws to reduce the scope of $\vee$.

$$\begin{aligned} A \vee (B \wedge C) &\equiv (A \vee B) \wedge (A \vee C) \\ (A \wedge B) \vee C &\equiv (A \vee B) \wedge (B \vee C) \end{aligned}$$

1.3.2 Changing a truth table into a logical expression in DNF (or full DNF).

Definition 15 *A **minterm** is a conjunction of literals in which each variable is represented exactly once.*

Given a truth table for a truth function f, write a logical expression that is a disjunction of minterms for each row where f is true. Each variable will be negated in the minterm only if the row represented contains a F value for that variable.

Example 16 *Given the table*

P	Q	R	f
T	T	T	T
T	T	F	F
T	F	T	T
T	F	F	F
F	T	T	F
F	T	F	F
F	F	T	T
F	F	F	F

We select lines 1,3,7 and create the expression:

$$f \equiv (P \wedge Q \wedge R) \vee (P \wedge \neg Q \wedge R) \vee (\neg P \wedge \neg Q \wedge R)$$

Definition 17 *If a truth function is expressed as a disjunction of minterms, it is said to be in **full disjunctive normal form**.*

2 Systems of Derivation

2.1 Main Rules of Inference

$A, B \models A \wedge B$	Law of Combination
$A \wedge B \models B$	Law of simplification
$A \wedge B \models A$	Variant of law of simplification
$A \models A \vee B$	Law of addition
$B \models A \vee B$	Variant of law of addition
$A, A \Rightarrow B \models B$	Modus ponens
$\neg B, A \Rightarrow B \models \neg A$	Modus tollens
$A \Rightarrow B, B \Rightarrow C \models A \Rightarrow C$	Hypothetical syllogism
$A \vee B, \neg A \models B$	Disjunctive syllogism
$A \vee B, \neg B \models A$	Variant of Disjunctive syllogism
$A \Rightarrow B, \neg A \Rightarrow B \models B$	Law of cases
$A \iff B \models A \Rightarrow B$	Equivalence elimination
$A \iff B \models B \Rightarrow A$	Variant of Equivalence elimination
$A \Rightarrow B, B \Rightarrow A \models A \iff B$	Equivalence introduction
$A, \neg A \models B$	Inconsistency Law because
	$False \Rightarrow B$ is trivially true

Definition 18 *A sound system is one in which the list of rules L only allow you to imply true statements (p51) [1]*

Definition 19 *In a complete system it must be possible to derive **every** conclusion that logically follows from the premises.*

Notice that table 2.1 is not

Definition 20 *If a system is sound and complete then $A \vdash B$ if and only if $A \models B$*

Definition 21 *A **theory** is given by a set of premises, thegether with all conclusions that can be derived from thepremises. The premises are often called **axioms**. The conclusions that can be derived from teh axioms are called **theorems**.*

Example 22 *Suppose that P and $\neg Q$ are axioms. Some theorems that can be derived from thes axioms are $P \vee Q$ and $Q \Rightarrow P$ however $P \wedge Q$ is not.*

2.2 Deduction Proofs

Procedure for the **Deduction Theorem** which says to prove $A \Rightarrow B$ one uses the following informal argument:

1. Assume A and add A to the premises.
2. Prove B, using A if necesary.
3. Discharge A, which means that A is no longer necessarily true, and write $A \Rightarrow B$.

The formal theorem can be stated as follows.

Theorem 23 (Deduction Theorem) *Let A and B be two expressions, and let $A_1, A_2, A_3, ...$ be premises. If $B, A_1, A_2, A_3, ...$ together logically imply ($\Rrightarrow$) C, then $A_1, A_2, A_3, ...$ logically imply ($\Rrightarrow$) $B \Rightarrow C$.*

Together table 2.1 and theorem 23 form a complete system.

2.3 Proving Equivalence

Here we are trying to prove $A \iff B$. To do this we have a three step process.

1. **Assume** A and derive B using 23 to prove $A \Rightarrow B$
2. **Conversly** assume B and derive A using 23 to prove $B \Rightarrow A$
3. Conclude that $A \iff B$

2.4 Indirect Proof

This is where we assume P and prove $\neg P$.

Example 24 *Prove:* $P \Rightarrow Q$, $P \Rightarrow \neg Q \vdash \neg P$

Formal Derivation	*Rule*	*Comment*
$P \Rightarrow Q$	*Premise*	
$P \Rightarrow \neg Q$	*Premise*	
$\quad P$	*Assumption*	*Assume P to derive contradition*
$\quad Q$	*1,3, Modus Ponens*	
$\quad \neg Q$	*2,3, Modus Ponens*	
$\quad Q \wedge \neg Q$	*4,5, C*	*4,5 Provide the contradiction*
$\neg P$	*Neg*	*Assumption P leads to a contradiction, thus we conclude that $\neg P$*

References

[1] Logic and Discrete Mathematics A computer Science Perspective by Winfried Karl Grassmann, Jean-Paul Tremblay