Week 2 Logic, Proofs, Boolean Algebra

Wednesday, 27 May 2020 11:20 AM

Logical Equivalence

Theorem

Excluded Middle	$p \vee \neg p \equiv \top$	*Identity*	$p \vee \bot \equiv p$
Contradiction	$p \wedge \neg p \equiv \bot$		$p \wedge \top \equiv p$
Idempotence	$p \vee p \equiv p$		$p \vee \top \equiv \top$
	$p \wedge p \equiv p$		$p \wedge \bot \equiv \bot$
Double Negation	$\neg\neg p \equiv p$		

Theorem

Commutativity	$p \vee q \equiv q \vee p$
	$p \wedge q \equiv q \wedge p$
Associativity	$(p \vee q) \vee r \equiv p \vee (q \vee r)$
	$(p \wedge q) \wedge r \equiv p \wedge (q \wedge r)$
Distribution	$p \vee (q \wedge r) \equiv (p \vee q) \wedge (p \vee r)$
	$p \wedge (q \vee r) \equiv (p \wedge q) \vee (p \wedge r)$
De Morgan's laws	$\neg(p \wedge q) \equiv \neg p \vee \neg q$
	$\neg(p \vee q) \equiv \neg p \wedge \neg q$
Implication	$p \Rightarrow q \equiv \neg p \vee q$
	$p \Leftrightarrow q \equiv (p \Rightarrow q) \wedge (q \Rightarrow p)$

Validity of Formulas

A formula ϕ is **valid**, or a **tautology**, denoted $\models \phi$, if it evaluates to T for *all* assignments of truth values to its basic propositions.

Example

A	B	$(A \Rightarrow B) \Rightarrow (\neg B \Rightarrow \neg A)$
F	F	T
F	T	T
T	F	T
T	T	T

Notation: $\forall$ means "for all" and $\exists$ means "there exist(s)"

Substitution Rules

If $\models \phi(P)$ then $\models \phi(\alpha)$.

If $\alpha \equiv \beta$ then $\phi(\alpha) \equiv \phi(\beta)$.

Boolean Functions

Formulae can be viewed as **Boolean functions** mapping valuations of their propositional letters to truth values.

A Boolean function of one variable is also called **unary**.
A function of two variables is called **binary**.
A function of n input variables is called **n-ary**.

Definition: Boolean Algebra

Every structure consisting of a set T with operations *join*: $a, b \mapsto a + b$, *meet*: $a, b \mapsto a \cdot b$ and *complementation*: $a \mapsto a'$, and distinct elements 0 and 1, is called a **Boolean algebra** if it satisfies the following laws, for all $x, y, z \in T$:

commutative:
- $x + y = y + x$
- $x \cdot y = y \cdot x$

associative:
- $(x + y) + z = x + (y + z)$
- $(x \cdot y) \cdot z = x \cdot (y \cdot z)$

distributive:
- $x + (y \cdot z) = (x + y) \cdot (x + z)$
- $x \cdot (y + z) = (x \cdot y) + (x \cdot z)$

identity: $x + 0 = x, \quad x \cdot 1 = x$

complementation: $x + x' = 1, \quad x \cdot x' = 0$

- $x + x = x, \quad xx = x$
- $xx' = 0, \quad x + x' = 1$
- $x \cdot 0 = 0,\ x \cdot 1 = x,\ x + 0 = x,\ x + 1 = 1$

Terminology and Rules

- A **literal** is an expression p or p', where p is a propositional atom.
- An expression is in CNF (conjunctive normal form) if it has the form
$$\prod_i C_i$$
where each **clause** C_i is a disjunction of literals e.g. $p + q + r'$.
- An expression is in DNF (disjunctive normal form) if it has the form
$$\sum_i C_i$$
where each clause C_i is a conjunction of literals e.g. pqr'.

Fact

1. $x + xy = x$ *(absorption)*
2. $xy + xy' = x$ *(combining the opposites)*

Canonical Form DNF

Given a Boolean expression E, we can construct an equivalent DNF E^{dnf} from the lines of the truth table where E is true:
Given an assignment π of $0, 1$ to variables $x_1 \dots x_i$, define the literal

$$\ell_i = \begin{cases} x_i & \text{if } \pi(x_i) = 1 \\ x_i' & \text{if } \pi(x_i) = 0 \end{cases}$$

and a product $t_\pi = \ell_1 \cdot \ell_2 \cdot \dots \cdot \ell_n$.

Example

If $\pi(x_1) = 1$ and $\pi(x_2) = 0$ then $t_\pi = x_1 \cdot x_2'$

The **canonical DNF** of E is

$$E^{dnf} = \sum_{E(\pi)=1} t_\pi$$

If E is defined by

x	y	E
0	0	1
0	1	0
1	0	1
1	1	1

then $E^{dnf} = x'y' + xy' + xy$
Note that this can be simplified to

$$x + y'$$

Karnaugh Maps

For optimisation, the idea is to cover the $+$ squares with the minimum number of rectangles. One *cannot* cover any empty cells (they indicate where $f(w, x, y, z)$ is 0).

- The rectangles can go 'around the corner'/the actual map should be seen as a *torus*.
- Rectangles must have sides of 1, 2 or 4 squares (three adjacent cells are useless).

Exercises

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

$\boxed{2.2.18}$ Prove or disprove:

(a) $p \Rightarrow (q \Rightarrow r) \;\equiv\; (p \Rightarrow q) \Rightarrow (p \Rightarrow r)$

(c) $(p \Rightarrow q) \Rightarrow r \;\equiv\; p \Rightarrow (q \Rightarrow r)$

You are planning a party, but your friends are a bit touchy about who will be there.

1. If John comes, he will get very hostile if Sarah is there.
2. Sarah will only come if Kim will be there also.
3. Kim says she will not come unless John does.

Who can you invite without making someone unhappy?

$\boxed{\text{2.7.14 (supp)}}$

Which of the following formulae are *always* true?

(a) $(p \wedge (p \Rightarrow q)) \Rightarrow q$ — always true

(b) $((p \vee q) \wedge \neg p) \Rightarrow \neg q$ — not always true

(e) $((p \Rightarrow q) \vee (q \Rightarrow r)) \Rightarrow (p \Rightarrow r)$ — not always true

(f) $(p \wedge q) \Rightarrow q$ — always true

Which of the following is a tautology?

- $\forall x\,(\exists y\,(P(x,y))) \Rightarrow \exists y\,(\forall x\,(P(x,y)))$ n
- $\exists y\,(\forall x\,(P(x,y))) \Rightarrow \forall x\,(\exists y\,(P(x,y)))$ a

$\boxed{\text{Example 10.1.2}}$ Define a Boolean algebra for 2-bit vectors $\mathbb{B}^2$

$0 \stackrel{\text{def}}{=} (0,0)$
$1 \stackrel{\text{def}}{=} (1,1)$
join: $(a_1, a_2) + (b_1, b_2) \mapsto (a_1 + b_1, a_2 + b_2)$
meet: $(a_1, a_2) \cdot (b_1, b_2) \mapsto (a_1 \cdot b_1, a_2 \cdot b_2)$
complementation: $(a_1, a_2)' \mapsto (a_1{}', a_2{}')$

Check that all Boolean algebra laws hold for $x, y \in \mathbb{B} \times \mathbb{B}$

Boolean Expression

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

CNF/DNF in Propositional Logic

$$\neg(\neg p \wedge ((r \wedge s) \Rightarrow q)) \equiv \neg(\neg p \wedge (\neg(r \wedge s) \vee q))$$

$\boxed{10.2.3}$ Find the canonical DNF form of each of the following expressions in variables x, y, z

- xy
- z'
- $xy + z'$
- 1

	yz	yz'	$y'z'$	$y'z$
x	$+$	$+$		$+$
x'	$+$		$+$	$+$

	yz	yz'	$y'z'$	$y'z$
wx	$+$	$+$		$+$
wx'	$+$	$+$	$+$	$+$
$w'x'$			$+$	$+$
$w'x$	$+$			$+$

$\boxed{\text{Example 10.1.1}}$ Define a Boolean algebra for the power set $\mathrm{Pow}(S)$ of $S = \{a, b, c\}$

$0 \stackrel{\text{def}}{=} \emptyset$
$1 \stackrel{\text{def}}{=} \{a, b, c\}$
join: $X, Y \mapsto X \cup Y$
meet: $X, Y \mapsto X \cap Y$
complementation: $X \mapsto \{a, b, c\} \setminus X$

Additional exercise:
Verify that all Boolean algebra laws (cf. slide 39) hold for