CSE 321 Discrete Structures

January 6, 2010

Lecture 02

Propositional Logic

Announcements

- Homework 1, Due January 13th
- Reading: sections 1.1, 1.2, 1.3
- Read handout on natural deduction !!
- Quiz section Thursday
  - 1:30 – 2:20 or 2:30 – 3:20
- Office hours
  - Dan Suciu, Monday 2:30-3:30
  - Andrew Hunter, CSE 218, T 3:30-4:30, F 12:30-1:30

Highlights from Lecture 1

- Propositional logic
  - Proposition: statement with a truth value
  - Basic connectives
    - $\neg$, $\vee$, $\wedge$, $\rightarrow$, $\oplus$, $\leftrightarrow$
  - Truth table for implication

p	q	$p \rightarrow q$

Terminology

- A compound proposition is a
  - **Tautology** if it is always true
  - **Satisfiable** if it is not always false
  - Contradiction if it is always false
  - Contingency if it can be either true or false

$p \vee \neg p$

$(p \oplus p) \vee p$

$p \oplus \neg p \oplus q \oplus \neg q$

$(p \rightarrow q) \wedge p$

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

Logical Equivalence

- *p* and *q* are **Logically Equivalent** if $p \leftrightarrow q$ a tautology.
- The notation $p \equiv q$ denotes *p* and *q* are logically equivalent
- Example: $(p \rightarrow q) \equiv (\neg p \vee q)$

p	q	$p \rightarrow q$	$\neg p$	$\neg p \vee q$	$(p \rightarrow q) \leftrightarrow (\neg p \vee q)$

The Main Problems in Propositional Logic

- Given p, prove that p is a tautology
- Given p, q, prove that $p \equiv q$
- These are basically the same thing:
  - WHY ?

The Main Problems in Propositional Logic

- Given p, prove that p is a tautology

- Given p, q, prove that p ≡ q

- These are basically the same thing:
  - A proposition p is a tautology iff p ≡ T
  - p ≡ q iff $p \leftrightarrow q$ is a tautology

Three Fundamental Approaches

- Truth table
  - We have seen that already
- Algebra:
  - Using logical equivalences
  - *Boolean Algebra*
- Logic
  - Using formal proof systems
  - We will use: *natural deduction*

1. Truth Table

- Describe an algorithm for checking whether p is a tautology

- What is the run time of the algorithm?

2. Boolean Algebras

A Boolean algebra is a set A, with two binary operations $\wedge$ and $\vee$, one unary operation $\neg$, and two constant 0, 1, satisfying the following:

$a \vee (b \vee c) = (a \vee b) \vee c$	$a \wedge (b \wedge c) = (a \wedge b) \wedge c$	associativity
$a \vee b = b \vee a$	$a \wedge b = b \wedge a$	commutativity
$a \vee (a \wedge b) = a$	$a \wedge (a \vee b) = a$	absorption
$a \vee (b \wedge c) = (a \vee b) \wedge (a \vee c)$	$a \wedge (b \vee c) = (a \wedge b) \vee (a \wedge c)$	distributivity
$a \vee \neg a = 1$	$a \wedge \neg a = 0$	complements

This list is complete. All other equivalences are derived.

2. Boolean Algebras

- Give examples of Boolean algebras:
  - …

- Stone’s theorem: every Boolean algebra is isomorphic to an algebra of sets

2. Boolean Algebras

- Theorem: every Boolean algebra satisfies the following equations, called **De Morgan's laws**

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

- What are the negations of:
  - Casey has a laptop and Jena has an iPod
  - Clinton will win Iowa or New Hampshire

2. Boolean Algebras

- There is no implication in a Boolean algebra
- Define as:
  - $p \rightarrow q \equiv \neg p \vee q$
- This allows us to derive several equivalences for implication

2. Boolean Algebras

Equivalences for implication

- $p \rightarrow q \equiv \neg p \vee q$
- $p \rightarrow q \equiv \neg q \rightarrow \neg p$
- $p \vee q \equiv \neg p \rightarrow q$
- $p \wedge q \equiv \neg (p \rightarrow \neg q)$
- $p \leftrightarrow q \equiv (p \rightarrow q) \wedge (q \rightarrow p)$
- $p \leftrightarrow q \equiv \neg p \leftrightarrow \neg q$
- $p \leftrightarrow q \equiv (p \wedge q) \vee (\neg p \wedge \neg q)$
- $\neg (p \leftrightarrow q) \equiv p \leftrightarrow \neg q$

3. Logical Proofs

- Natural deduction
- (on the white board)