# SE212

Andy Zhang

Fall 2014

# Contents

# Chapter 1

# Module 1 — Sept 8

## 1.1 Logic and Computation, module 1

### 1.1.1 Contact

Amirhossein Vakili, avakili@uwaterloo.ca
SE212 Mon/Wed 11:30 to 12:50 RCH 305
Office Hours: Tues 2:30 to 3:30 DC2551

### 1.1.2 Course content

How do you know what a program is supposed to do? (specification/correctness)

- Inspection
- Testing
- **Formal verification**

### 1.1.3 Logic

**Formal verification**
Logic is based on logical reasoning, also called 'formal methods' or 'computer-aided verification'. It checks the correctness of a program for all outputs. Since this takes a lot of effort, it is complementary to testing and inspection.

**Logic**
A logic consists of:

- *syntax* What is an acceptable sentence
- *semantics* What do the symbols and sentences in the language mean?
- *proof theory* How do we construct valid proofs?

Logic provides a way to express knowledge precisely and to reason consequences of that knowledge

### 1.1.4 Course Outline

Four main topics:

- Propositional logic
- Predicate logic

- Set Theory and Specification
- Program correctness

### 1.1.5 Marking Scheme

- 20% assignments (top 7 out of 8 assignments)
- 25% Midterm exam
- 55% Final exam

# Chapter 2

# Module 2

## 2.1 Definitions

### 2.1.1 Elements of a logic

Logic consists of **syntax**, **semantics** and **proof theory**

### 2.1.2 Syntax

'well-formed formula' (wff), is a word that is a part of a formal language

### 2.1.3 Semantics

$\vdash$ is entails.
Ex: '$\vdash p$' means the formula $p$ is valid, where $p$ is a wff in the logic.
Ex: $p_1, p_2, \ldots, p_n \vdash q$ means from the premises ($p$'s), we may conclude $q$ where they're all wff.

### 2.1.4 Proof Theory

Define '$\models$' as proves. It's a way to calculate $p_1, p_2, \ldots, p_n \models q$, meaning there's a way to determine if $q$ is true if $p_1, p_2, \ldots, p_n$ are true
There may be multiple proof theories indicated by a subscript e.g. $\models_{ND}$ for natural deduction proof theory.

**Proof Theory** are methods that manipulate strings of symbols base on pattern matching. There may be multiple ways to prove a formula.

**Sound** If $p_1, p_2, \ldots, p_n \models q$ (proof), then $p_1, p_2, \ldots, p_n \vdash q$ (valid)

**Complete** If $p_1, p_2, \ldots, p_n \vdash q$ (valid) then $p_1, p_2, \ldots, p_n \models q$ (proof)

## 2.2 Syntax

### 2.2.1 Composition

A formula in propositional logic consists of constant symbols (**true** and **false**), proposition letters, propositional connectives ($\neg, \wedge, \vee, \Rightarrow, \Leftrightarrow$), and brackets

### 2.2.2 Symbol Definitions

$\neg$ Negation
$\wedge$ And
$\vee$ Or
$\Rightarrow$ Implication
$\Leftrightarrow$ Equivalent

### 2.2.3 Rules and Definitions

- Brackets around the outermost formula are usually omitted. Priority is ($\neg, \wedge, \vee, \Rightarrow, \Leftrightarrow$)
- All binary logical connectives are **right associative**
- $a \wedge b$ is **conjucts**, $a \vee b$ is **disjuncts**
- Contrapositive of $a \Rightarrow b$ is $\neg b \Rightarrow \neg a$
- Prime propositions are declarative sentences i.e. sentences that are true or false
- Propositional letters are $a, b, p, q$
- Prime compositions are indecomposable, compound composition are decomposable
- The connective 'and' used in logic is commutative
- Watch out for the false implies everything problem

## 2.3 Semantics

### 2.3.1 Definition

**Semantics** means meaning, providing an interpretation/functions of expressions in one world in terms of values in another world. Proof theories transform wff in ways that respect semantics. The syntax of the propositional logic is the **domain** of the semantic function whereas the set of truth values is the **range**

**Boolean Valuation** is a function $v$ from the set of formulas in propositional logic to the set $T_r$. Boolean valuation is also called a model or an interpretation.

- $v(false) = F, v(true) = T$
- $v(\neg p) = NOT(v(p))$
- For the connectives:
  - $v(p \wedge q) = v(p) AND v(q)$
  - $v(p \vee q) = v(p) OR v(q)$
  - $v(p \Rightarrow q) = v(p) IMP v(q)$
  - $v(p \Leftrightarrow q) = v(p) IFF v(q)$

### 2.3.2 Semantics of Propositional Logic

**NOT** takes a truth value and returns a truth value

**AND, OR, IMP, IFF** take two truth values and return a truth value that correspond to $\wedge, \vee, \Rightarrow, \Leftrightarrow$

**Truth Tables** have a row for each possible boolean valuation, a column for each subformula for the formula, and the formula itself, and each cell contains the truth value given by the boolean valuation of that row. We can use truth tables to determine if a formula is satisfiable/tautology/-contradiction/contingent

**Satisfiability** If a formula is **satisfiable**, then there exists a Boolean valuation $v$ such that $v(p) = T$

**Tautologies** A propositional formula $p$ is a **tautology** or **valid** if $v(p) = T$ for all Boolean valuations $v$. When a formula $q$ is a tautology, we write

$$\vdash q$$

**Logical Implication** A formula $p$ **logically implies** a formula $q$ iff for all Boolean valuations $v$, if for all premises $v(p_i) = T$, then $v(q) = T$, meaning

$$p \vdash q \text{ which is equivalent to } \vdash p \Rightarrow q$$

**Contradiction** A propositional formula $a$ is a **contradiction** if $v(a) = F$ for all Boolean valuations $v$

**Contingent** A **contingent** is one that is neither a **tautology** nor a **contradiction**

**Logical Equivalence** Two formulas are **logically equivalent** iff their equivalence is a tautology i.e. $v(p) = v(q)$ for all $v$.

$$p \leftrightarrow q \text{ which also means } \vdash p \Leftrightarrow q$$

$\leftrightarrow$ Logical equivalence
$\Leftrightarrow$ Material equivalence

**Consistency** A collection of formulas is **consistent** if there exists a boolean valuation where all the formulas can be true simultaneously. If a set of formulas in the antecedent of an implication, they can be used to prove a contradiction.

## 2.4 Proof Theories

### 2.4.1 Use Case

Since truth tables grow exponentially, we can use a **proof theory** instead to determine whether a formula is a tautology. As long as the proof theory is **sound**, we can use proof theory in place of truth tables to determine tautologies (and valid arguments).

### 2.4.2 Transformational Proofs

**Transformational Proofs** is a means of determining that two wff formulas of propositional logic $p$ and $q$ are logically equivalent by the repeated exchange of subformulas of $p$ for logically equivalent subformlas that result in $p$ being transformed into $q$. Each step must follow a logical law expressed by $\Longleftrightarrow$.
$p \Leftarrow^{?}\Rightarrow q$ means 'Show by transformational proof that $p \Longleftrightarrow q''$'

**Transformational Proof Rules**

Comm $p \wedge q \Longleftrightarrow q \wedge p$
Lem $p \vee \neg \Longleftrightarrow true$
Contr $p \wedge \neg p \Longleftrightarrow false$
Impl $p \Rightarrow q \Longleftrightarrow \neg p \vee q$
Idemp $p \wedge p \Longleftrightarrow p$
Neg $\neg(\neg p) \Longleftrightarrow p$
Simp1 $p \wedge true \Longleftrightarrow p$
Assoc $p \wedge (q \wedge r) \Longleftrightarrow (p \wedge q) \wedge r$
Dm $\neg(p \wedge q) \Longleftrightarrow \neg \vee \neg q$
Distr $p \vee (q \wedge r) \Longleftrightarrow (p \vee q) \wedge (p \vee r)$
Contrapos $p \Rightarrow q \Longleftrightarrow \neg q \Rightarrow \neg p$
Equiv $p \Leftrightarrow q \Longleftrightarrow (p \Rightarrow q) \wedge (q \Rightarrow p)$
Simp2 $p \vee (p \wedge q) \Longleftrightarrow p$

**Rules**

1. **Rule of substitution** substituting an equivalent for a subformula
2. **Rule of transitivity** If $p \Longleftrightarrow q$ and $q \Longleftrightarrow r$, then $p \Longleftrightarrow r$

**Rule of Thumb**

1. Eliminate implication ($\Rightarrow$) and equivalence ($\Leftrightarrow$) using the law of implication, the law of equivalence and the contrapositive law backwards
2. Simplify as soon as you can (simp 1, simp 2, idempotence, negation, kaw of contradiction, law of excluded middle)
3. Sometimes use the various kinds of simplification backwards to prepare for using distributivity

### 2.4.3 Semantics

Transformational proof satisfies the following:

1. If $p \Longleftrightarrow q$ can be proved, then $p \leftrightarrow q$ (**soundness**)
2. If $p \leftrightarrow q$, then $p \Longleftrightarrow q$ can be proved (**completeness**)

Thus, transformational proof is sound and complete for propositional logic, and we use this to show the logical equivalence of two formulas