Mathematical Techniques for Computer Science

COMP11120

Renate Schmidt

Renate.Schmidt@manchester.ac.uk
http://www.cs.manchester.ac.uk/pgt/2015/COMP11120/

October 18, 2015

Formal Logic Systems

Outline (6 lectures)

Introduction & Propositional logic

More on semantics

Equivalent formulas

Normal forms, Boolean functions

Propositional natural deduction system

First-order logic

Section 1

Introduction & Propositional logic

Practical Information

Handout: Lecture notes, incl. weekly assessed exercise sheet
Printed copy from outside SSO

Blackboard:

- Lecture notes, slides, solutions to exercises, announcements, discussion forum, podcast
- Visit on a weekly basis

Course website: `www.cs.manchester.ac.uk/pgt/2015/COMP11120/`

- Syllabus description, exam information, etc.

Format of exercise sheets and examples classes:

- The same as for Andrea's parts

Exam: 1 question on my part (33%)

Mid-term test: Will include questions on my part

Material: Significantly reworked and extended from last year

Content

Formal Logic Systems: With emphasis on

- propositional logic
- first-order logic

Two of the most important logical formalisms in computer science

- All you need to know for the exam is in the lecture notes, and it is advisable that you can do the assignments and can solve the unassessed exercises.
- Good strategy: Go through the material discussed in lectures after each lecture, do as many of the unassessed exercises as you can, do the assignments, check the solutions, and read ahead.
  And, ask questions!

Background reading

On the reading list

- Truss, J. K. (1999). Discrete Mathematics for Computer Scientists, Addison-Wesley
- Epp, Susanna S. (2011). Discrete Mathematics with Applications, Brooks/Cole
- Jordan, D. W. and P. Smith (2008). Mathematical Techniques: An Introduction for the Engineering, Physical, and Mathematical Sciences, Oxford University Press

Check the main library, and the internet is a useful resource

There is NO need to buy a book

Don't get confused!

There are variations in the presentation and definitions between different sources and books, and also to those we use

Motivation

Mathematical arguments are very schematic

$$
\begin{aligned}
x \in S \cap T \quad &\text{if} \quad x \in S \text{ and } x \in T \\
x \in S \cup T \quad &\text{if} \quad x \in S \text{ or } x \in T
\end{aligned}
$$

They use key phrases

and, or, if-then, iff, not

History of making logical arguments formal is very long

Logical arguments

Correct arguments can be constructed with simple principles and a simple language

Basic units:

Propositions that are either **true** or **false**, but not both

Example (Propositions)

- 2 divides 4
- 1 is even
- $2 + 2 = 5$
- It is raining
- I study hard
- I get a good grade

Connected with:

not, and, or, if-then, iff

to form complex propositions

Examples of compound propositions

Example (Compound propositions)

- 1 is not even
- I study hard and 1 is not even
- If I do not study hard then I not get a good grade

Compound proposition = proposition formed from atomic propositions and connectives

Atomic proposition = basic unit, proposition which can't be broken down

Propositional logic

Propositional logic = the logic of true and false propositions

Aim:

- Study general principles of correct arguments
- Validate the correctness of arguments

Symbols in the language of propositional logic

Propositional connectives

$\neg$	not		
$\wedge$	and	$\vee$	or
$\rightarrow$	implies/if-then	$\leftrightarrow$	iff

Propositional variables (basic units)

$$P, \quad Q, \quad R, \quad \ldots$$

From propositional variables we can construct propositional formulas using the connectives:

$$((\neg P) \rightarrow (\neg Q))$$

Intuitive meaning

. . . is the same as in mathematics:

$\neg P$ is true	iff	P is false
$P \wedge Q$ is true	iff	both P and Q are true
$P \vee Q$ is true	iff	at least one of P or Q are true
$P \rightarrow Q$ is true	iff	if P is true then Q is true

Moreover we said:

$$P \leftrightarrow Q \quad \text{is the abbreviation for} \quad (P \rightarrow Q) \wedge (Q \rightarrow P)$$

Propositional formulas

Definition (Propositional formulas)

- Each propositional variable is a propositional formula (called **atomic formula**, or simply **atom**)
- If A is a propositional formula, then
$$(\neg A)$$
is a propositional formula
- If A and B are propositional formulas, then
$$(A \land B), \qquad (A \lor B), \qquad (A \to B) \qquad \text{and} \qquad (A \leftrightarrow B)$$
are propositional formulas

Notational convention

Problem: $((P \land Q) \to (Q \lor (\neg P)))$ contains too many brackets

Binding precedence (from highest to lowest)

$$\neg$$
$$\lor \qquad \land$$
$$\to \qquad \leftrightarrow$$

Allows us to write

$$\begin{aligned} P \land Q \to R &\quad \text{for} \quad ((P \land Q) \to R) \\ \neg P \to Q \lor R &\quad \text{for} \\ P \land Q \to Q \lor \neg P &\quad \text{for} \\ P \land Q \lor R & \end{aligned}$$

Useful tip: When in doubt use brackets

Subformulas

Example

Suppose A is this formula

$$(P \wedge Q) \rightarrow (Q \vee \neg P)$$

Subformulas of A

$$\begin{array}{lll} (P \wedge Q) \rightarrow (Q \vee \neg P), & & \\ P \wedge Q, & Q \vee \neg P, & \\ P, \quad Q & \text{and} & \neg P \end{array}$$

Boolean semantics

Semantics gives formal meaning to formulas

In the **Boolean semantics** of propositional logic,
there are two **truth values**:

1 true **0** false

Standard truth tables for the connectives

How do the connectives affect truth values?

If we assign truth values to the variables in a propositional formula then the truth value of the formula can be computed

$$P \wedge \neg Q$$

P	$\neg P$
1	**0**
0	**1**

P	Q	$P \wedge Q$
1	**1**	**1**
1	**0**	**0**
0	**1**	**0**
0	**0**	**0**

Standard truth tables for the connectives (cont'd)

P	Q	$P \vee Q$
1	**1**	**1**
1	**0**	**1**
0	**1**	**1**
0	**0**	**0**

P	Q	$P \to Q$
1	**1**	**1**
1	**0**	**0**
0	**1**	**1**
0	**0**	**1**

P	Q	$P \leftrightarrow Q$
1	**1**	**1**
1	**0**	**0**
0	**1**	**0**
0	**0**	**1**

Truth table for a formula

Example for $(P \wedge Q) \to (Q \vee \neg P)$

P	Q	$(P \wedge Q) \to (Q \vee \neg P)$
1	**1**	**1**
1	**0**	**1**
0	**1**	**1**
0	**0**	**1**

Long version:

P	Q	$P \wedge Q$	$\neg P$	$Q \vee \neg P$	$(P \wedge Q) \to (Q \vee \neg P)$
1	**1**				
1	**0**				
0	**1**				
0	**0**				