Studying Logic at the University of Sydney

Nicholas J.J. Smith
http://phi.njjsmith.com

28 May 2020

The latest version of this document can be found at the following URL:
http://njjsmith.com/philosophy/teaching/studylogic.pdf
Feel free to share this document or the above URL with anyone you think might be interested in studying logic at the University of Sydney.

This document describes the four logic units offered by the Philosophy department:

- PHIL1012 Introductory Logic
- PHIL2610 Exploring Nonclassical Logics
- PHIL2615 Logic and Proof
- PHIL3610 Logic and Computation

1 Structuring your study

PHIL1012 is a prerequisite for each of the other three units. (If you have not taken PHIL1012 but have equivalent knowledge, you can get Special Permission to enrol in the other units.)

Once you have taken PHIL1012, you may take any one, two, or all three of PHIL2610, PHIL2615 and PHIL3610 ***in any order***.

PHIL1012 is offered every year, usually in semester two and winter school/July intensive. The other units are offered on rotation, usually at least once every two years.

S1 = first semester; S2 = second semester:

PHIL2610 Exploring Nonclassical Logics:

S1 2007, S2 2009, S1 2011, S2 2012, S1 2014, S2 2016, S2 2018, S2 2020

PHIL2615 Logic and Proof:

S2 2013,* S2 2014,* S2 2017, S2 2019, S1 2020, S1 2021

* under the title Intermediate Logic

PHIL3610 Logic and Computation:

S2 2010,* S1 2012,* S1 2013,* S1 2015,* S1 2018,* S1 2020, S2 2021

* under the code PHIL2650

2 Brief descriptions of the units

PHIL1012 Introductory Logic

An introduction to modern logic: the investigation of the laws of truth. One essential aspect of good reasoning or argumentation is that it is valid: it cannot lead from true premisses to a false conclusion. In this course we learn how to identify and construct valid arguments, using techniques such as truth tables, models and truth trees. Apart from being a great aid to clear thinking about any subject, knowledge of logic is essential for understanding key areas of contemporary philosophy, linguistics, mathematics and computing. This unit provides a thorough grounding in classical logic, covering both model theory and proof theory of propositional and predicate (also known as quantificational or first order) logic with identity.

PHIL2610 Exploring Nonclassical Logics

Classical logic is what you study in introductory units such as PHIL1012. This unit covers major extensions of and alternatives to classical logic, such as temporal, modal, intuitionist, relevance, many-valued and fuzzy logics. As well as looking at the internal workings of these logics, we examine some of their applications, and the philosophical issues surrounding them.

PHIL2615 Logic and Proof

We examine the major ways of proving things in logic: tableaux (trees), axiomatic proofs, natural deduction and sequent calculus. We learn to construct proofs of each of these kinds and then establish fundamental adequacy results (e.g. soundness and completeness) for each kind of proof system.

PHIL3610 Logic and Computation

This unit covers central topics and results concerning the nature of logic, the nature of computation, and the relationships between the two, such as Turing machines, computability and uncomputability, the undecidability of first order logic, computational complexity, and Gödel's incompleteness theorems.

3 Further information about the units

This section is primarily intended for students who have taken PHIL1012 and are considering further logic units (i.e. it will not be fully comprehensible to someone who does not know any logic at all).

Here are two important distinctions in logic:

1. classical *vs* nonclassical logics

   By *classical* logic we mean predicate logic with identity—the final logical system introduced in PHIL1012—and its various subsystems also covered in PHIL1012 (propositional logic, monadic predicate logic and predicate logic without identity). These are important logical systems but they are far from the only ones. Nonclassical logics either extend classical logic with new kinds of logical resources or reject or revise parts of classical logic.

2. logic *vs* metalogic

   One thing we do in logic is construct formal proof systems and then use them to prove things. For example, in PHIL1012 we introduce a system of tree proofs for predicate logic with identity and then we use it to show for example that $\forall x \forall y \forall z((x = y \land y \neq z) \rightarrow x \neq z)$ is a logical truth (we show this by starting a tree with the negation of this formula and seeing that all paths close). Call this activity of constructing formal proof systems and constructing formal proofs within such systems *logic*. A second thing that logicians do is prove things *about* formal proof systems. This is what we mean by *metalogic*. For example, we can prove that a set of formulas of propositional logic is unsatisfiable (there is no truth table row on which all the formulas are true) iff all paths close in a finished tree that begins with those formulas. This tells us something important about the system of tree proofs for propositional logic: it agrees perfectly with the truth tables.

In the following we say how the four logic units fall with respect to the two distinctions just introduced.

PHIL1012 Introductory Logic

entirely classical: no nonclassical logics

all logic: no metalogic

PHIL2610 Exploring Nonclassical Logics

entirely nonclassical. The unit falls into two main parts. In Part I we look at logics of vagueness and indeterminacy. The major families of logics covered here are many-valued, fuzzy and supervaluationist logics. In Part II we look at intensional logics. The major families of logics covered here are modal, tense and intuitionist logics.

a mix of logic and metalogic. Logic: introducing new nonclassical formal systems and constructing proofs in these systems. Metalogic: examining the properties of the new nonclassical systems and comparing them with the properties of classical logic.

PHIL2615 Logic and Proof

entirely classical

a mix of logic and metalogic. The proof system used in PHIL1012 is the tree (aka tableaux) system. There are other kinds of proof system in logic, and in this unit we look at the other major kinds: axiomatic proofs (aka Hilbert systems), natural deduction and sequent calculus. We introduce proof systems of each of these kinds and learn to construct proofs in these systems (just as we learned to construct tree proofs in PHIL1012). That is the logic part of the unit. However unlike in PHIL1012 we also prove the basic metalogical results about each of these kinds of proof systems (including trees): soundness and completeness results (and as a corollary, compactness). Roughly, soundness means that the proof system does not prove anything we do not want it to prove and completeness means that it does prove everything we do want it to prove.

PHIL3610 Logic and Computation

entirely classical

all metalogic. We look at what is *hard* and what is *impossible* in logic. Consider the question whether a formula α is satisfiable (true on some model). Could we programme a computer to answer this question (when given any formula α as input)? Where α is a formula of propositional logic or monadic predicate logic, the answer is (in principle) Yes. Where α is a formula of general predicate logic, the answer is No: not just in practice but even in principle, the task is an *impossible* one. This result is known as the *undecidability* of first order logic. We lead up to proving it via an examination of models of computation including Turing machines. Now consider propositional logic. As noted, the question whether α is satisfiable is in principle answerable by a computer—but nevertheless it is a *hard* question. More precisely, it is 'NP-complete'. We explain what this means—via a more general consideration of computational complexity—and then prove the result. Finally, we turn to another task that turns out to be impossible. It is possible to give a set of basic axioms and rules from which all the truths of logic can be deduced—i.e. there are *complete* proof systems for logic. However, it turns out to be impossible to give a set of basic axioms and rules from which all the truths of arithmetic can be deduced. This is Gödel's famous first *incompleteness* theorem. We lead up to proving it via a consideration of the issue of formalising arithmetical reasoning within a logical system.