COM3412: Logic and Computation

Specimen answers to the 2006 examination

1. (a) An logical inference consists of a set of logical formulae called the premises together with a single formula called the conclusion, which is claimed to follow logically from the premises. **(2 marks)**

   An inference is valid if its conclusion is a logical consequence of its premises, i.e., if the truth of the premises is sufficient to guarantee the truth of the conclusion. **(3 marks)**

   (b) $A \vdash A \wedge B$ is invalid since if A is true and B is false then the premise is true but the conclusion is false. **(2 marks)**

   $A \wedge B \vdash A$ is valid since in order for the premise to be true, both A and B must be true, and hence the conclusion must be true. **(2 marks)**

   $A \vdash A \vee B$ is valid, since the conclusion is true whenever at least one of A and B is true, and therefore in particular if the premise, A, is true. **(2 marks)**

   $A \vee B \vdash A$ is invalid, since if B is true and A is false then the premise is true but the conclusion is false. **(2 marks)**

   (c) A proof system for a logic is any collection of procedures by which inferences in the logic may be determined to be valid or invalid. (Examples: truth tables for Propositional Calculus, natural deduction and truth trees for Propositional or Predicate Calculus.) **(2 marks)**

   A proof system for a logic is sound if every inference that it certifies as valid is in fact valid in that logic. **(3 marks)**

   A proof system for a logic is complete if it certifies as valid every inference that is in fact valid in that logic. **(3 marks)**

   A proof system is a decision procedure for a logic if it will correctly determine, for *every* inference, whether or not it is valid in that logic. **(3 marks)**

   (d)
   - Interpretation **I1**, proof system **PS1**.
     If letter A appears in one of the premises then that premise implies that A is true (since it appears as a conjunct of that premise); if **PS1** says the inference is valid then every letter in the conclusion appears in one of the premises and hence is true if the premises are true. The conclusion is the conjunction of its constituent letters and hence is true if they all are. Hence the inference *is* valid, and the proof system is **sound**. **(2 marks)**

     Conversely, for an inference to be valid, each of the letters in the conclusion must be implied by the premises; the only way this can happen is for the letter to appear in at least one of the premises, which means that **PS1** says the inference is valid. Hence the proof system is **complete**. **(2 marks)**

- Interpretation **I1**, proof system **PS2**.
  The proof system is **not sound** since the inference $A \vdash A \circ B$, though invalid (being equivalent to $A \vdash A \wedge B$ under **I1**), is certified as valid by **PS2**. **(2 marks)**
  It is **not complete**, since the inference $A \circ B \vdash A$, though valid (i.e., $A \wedge B \vdash A$), is not recognised as valid by **PS2**. **(2 marks)**

- Interpretation **I2**, proof system **PS1**.
  Not sound since $A \circ B \vdash A$ (i.e., $A \vee B \vdash A$) is invalid but certified as valid by **PS1**. **(2 marks)**
  Not complete since $A \vdash A \circ B$ (i.e., $A \vdash A \vee B$), though valid, is rejected as invalid by **PS1**. **(2 marks)**

- Interpretation **I2**, proof system **PS2**.
  If **PS2** validates a certain inference, then for each premise, every letter appearing in it also appears in the conclusion; the premise is just the disjunction of those letters, and the conclusion is therefore a disjunction of some set of letters including at least those; hence each premise implies the conclusion, so the inference is valid. Hence the proof system is **sound**.

  (2 marks)

  But it is **not complete**, since e.g., the valid inference $A, B \vdash A$, is rejected as invalid. **(2 marks)**

2. (a) If the premises are contradictory, they have no model, and therefore *any* interpretation satisfying them must satisfy the conclusion (since any counter-example to this statement would be a model for the premises which falsifies the conclusion — and there are no models for the premises).

   (4 marks)

   Any inference with a logically true conclusion (i.e., satisfied in every model) is valid.

   (2 marks)

   (b) Suppose $\Sigma \models C$ and $\Sigma \subseteq \Sigma'$. Suppose that interpretation I satisfies Σ'. This means that it satisfies every formula in Σ'. Since $\Sigma \subseteq \Sigma'$, these formulae include all the formulae in Σ, so I satisfies Σ. Since $\Sigma \models C$, this means that I satisfies C. We have now shown that any interpretation satisfying Σ' also satisfies C. Hence $\Sigma' \models C$. This means that $\models$ is monotonic.

   (7 marks)

   (c) If we have $P_1, \ldots, P_n \models C$, then any interpretation satisfying $\{P_1, \ldots, P_n\}$ satisfies C, and therefore falsifies $\neg C$. This means that *no* interpretation satisfies $\{P_1, \ldots, P_n, \neg C\}$. It therefore follows, vacuously, that any interpretation satisfying $\{P_1, \ldots, P_n, \neg C\}$ also satisfies C (or indeed anything else), and so $P_1, \ldots, P_n, \neg C \models C$, in conformity with nonmonotonicity. **(7 marks)**

   (d) There are many possible examples, e.g., no-one answers the doorbell so I "infer" that no-one is at home; then I learn that everyone in the house is fast asleep, and rescind the inference. [Discussion needed.]

   (10 marks)

3. (a) A first-order theory is a set of first-order formulae that is satisfiable and deductively closed. **(3 marks)**

(b) The first-order theory of domain D, relative to a given language, consists of all formulae in the language which are satisfied by D (with the specified interpretation). **(2 marks)**
This set of formulae must be satisfiable, since it is satisfied by D. It is also deductively closed, since any logical consequence of the set is satisfied by any model for the set, hence by D, hence it is already in the set. Therefore it is a first-order theory in the sense of part (a). **(3 marks)**

(c) "Everyone has a mother." **(2 marks)**

(d) Some suitable examples are:

$\forall x \exists y(\neg F(y) \wedge P(y,x))$ (Everyone has a father)
$\forall x,y,z,w(P(y,x) \wedge P(z,x) \wedge P(w,x) \rightarrow y = z \vee y = w \vee w = z)$ (No-one has more than two parents)
$\forall x \neg P(x,x)$ (No-one is their own parent)
$\forall x,y(P(x,y) \rightarrow \neg P(y,x))$ (No-one is their own grandparent)
$\forall x,y,z(P(x,y) \wedge P(y,z) \rightarrow \neg P(z,x))$ (No-one is their own great-grandparent)
$\forall x,y,z(P(x,z) \wedge P(y,z) \wedge x \neq y \rightarrow (F(x) \leftrightarrow \neg F(y))$ (No-one has more than one parent of the same sex)

I would also allow any logical truth (e.g., $\forall x(F(x) \rightarrow F(x))$), but not more than one of these.

1 mark for each formula plus half a mark for its English translation — up to 6 formulae

(e) An *axiomatisation* is a finitely-specifiable set of first-order formulae intended to serve as a basis for the first-order theory in the sense that the theory is exactly the set of logical consequences of the set.

(2 marks)

The axiomatisation is sound if the set of its logical consequences is a subset of the theory. **(2 marks)**

It is complete if the theory is a subset of the logical consequences of the axioms. **(2 marks)**

(f) The main modification is that the formulae saying that everyone has a mother and everyone has a father must be replaced by formulae asserting this of everyone except Adam and Eve. Using constants a and e, we have

$\forall x(x = a \vee x = e \vee \exists y(F(y) \wedge P(y,x)))$
$\forall x(x = a \vee x = e \vee \exists y(\neg F(y) \wedge P(y,x)))$

We also need a formula saying that Adam and Eve have no parents:

$\forall x(\neg P(x,a) \wedge \neg P(x,e))$.

What we can't say, in first-order logic, is that all humans are descended from Adam and Eve—for this we need second-order logic. [Bonus mark for anyone who points this out!] **(5[+1] marks)**

4. (a) i. If started with a blank tape it runs through the sequence

$$\begin{array}{ccc}
\Delta & \boxed{\Delta} & \Delta \\
 & 1 & \\
\Delta & 1 & \boxed{\Delta} \\
 & & 3 \\
\Delta & \boxed{1} & 1 \\
 & 1 & \\
\Delta & \Delta & \boxed{1} \\
 & & 2 \\
\Delta & \boxed{\Delta} & \Delta \\
 & 1 &
\end{array}$$

ad infinitum, hence fails to terminate.

(4 marks)

ii. If started with a tape containing ‘1’ on the initially scanned square and otherwise blank, it reaches the halt state after three steps:

$$\begin{array}{ccc}
\boxed{1} & \Delta & \Delta \\
1 & & \\
\Delta & \boxed{\Delta} & \Delta \\
 & 2 & \\
\Delta & 1 & \boxed{\Delta} \\
 & & 4 \\
\Delta & 1 & \boxed{\Delta} \\
 & & 0
\end{array}$$

(4 marks)

iii. If started with the tape ‘$1\Delta1$’, scanning the first ‘1’, the machine gets stuck in a non-halting state:

$$\begin{array}{ccc}
\boxed{1} & \Delta & 1 \\
1 & & \\
\Delta & \boxed{\Delta} & 1 \\
 & 2 & \\
\Delta & 1 & \boxed{1} \\
 & & 4
\end{array}$$

(4 marks)

(b) Here I’m expecting an account of how Turing reduced the decision problem for first-order logic to the halting problem for Turing machines; together with at least an outline of the proof that the halting problem for Turing machines is recursively unsolvable.

(18 marks)

5. For Gödel, I'm looking for at least:

   - What is meant by the first-order theory of the arithmetic of the natural numbers.
   - What is meant by a complete axiomatisation of a first-order theory.
   - The fact that the first-order theory of arithmetic cannot be finitely axiomatised.
   - A description of how Gödel showed this to be the case, including at least a reference to the arithmetisation of syntax and the Gödel numbering system by which this is accomplished, as well as the idea of a Gödel sentence asserting its own unprovability and hence the deduction from this that the system cannot be both consistent and complete.

   For Cook,

   - What is meant by the term 'NP-complete' (including the idea of a polynomial-time reduction of one problem to another).
   - The satisfiability problem SAT for Propositional Calculus clauses.
   - The existence of a polynomial-time reduction for an arbitrary NP problem to SAT.
   - A description of how Cook showed that such a reduction can always be done, using the Turing-machine formulation of the definition of the class NP.