BSc/MSci EXAMINATION

LOGIC

(CM328X Logic, 6CCM328A Logic)

MAY-JUNE 2010

TIME ALLOWED: TWO HOURS

THIS PAPER CONSISTS OF TWO SECTIONS, SECTION A AND SECTION B.

SECTION A CONTRIBUTES HALF THE TOTAL MARKS FOR THE PAPER. ANSWER ALL QUESTIONS IN SECTION A.

QUESTIONS IN SECTION B CARRY EQUAL MARKS, BUT IF MORE THAN TWO ARE ATTEMPTED THEN ONLY THE BEST TWO WILL COUNT.

QUESTIONS SHOULD BE ANSWERED IN THE SPACES PROVIDED ON THE QUESTION PAPER.

YOU HAVE BEEN PROVIDED A BOOKLET FOR ROUGH WORK.
IT MUST BE ATTACHED TO THIS PAPER, BUT WILL NOT BE MARKED.

NO CALCULATORS ARE PERMITTED.

SECTION A

QUESTION 1 **(10 points)**

(a) Express each of the following in the notation of propositional logic. ***Advice*****: If in doubt leave blank, as each incorrect answer loses as much as a correct answer gains.**

(i) if p then q, (ii) p if q, (iii) p only if q, (iv) only if p do we have q, (v) p provided that q, (vi) p unless q, (vii) p is a necessary and sufficient condition for q, (viii) p precisely if q, (ix) p but q, (x) neither p nor q.

> Answer (5 points worth ½ point each but ½ off for each incorrect answer):
> (i) $p \to q$, (ii) $q \to p$, (iii) $p \to q$, (iv) $q \to p$, (v) $q \to p$, (vi) $\neg q \to p$, (vii) $p \leftrightarrow q$, (viii) $p \leftrightarrow q$, (ix) $p \wedge q$, (x) any one of the following suffices: $\neg p \wedge \neg q$, $\neg(p \vee q)$, $p \downarrow q$.

(b) Use successive transformations to put the following formula into disjunctive normal form: $p \to (\neg q \wedge \neg(r \wedge \neg s))$.

> Answer (3 points): $p \to (\neg q \wedge \neg(r \wedge \neg s))$ -||- $\neg p \vee (\neg q \wedge \neg(r \wedge \neg s))$ -||- $\neg p \vee (\neg q \wedge (\neg r \vee s))$ -||- $\neg p \vee (\neg q \wedge \neg r) \vee (\neg q \wedge s)$.

(c) Express the same formula $p \to (\neg q \wedge \neg(r \wedge \neg s))$ in prefix (Polish) notation.

> Answer (2 points): $\to p \wedge \neg q \neg \wedge r \neg s$.

QUESTION 2 (10 points)

(a) Outline a proof that every substitution instance of a tautology is a tautology.

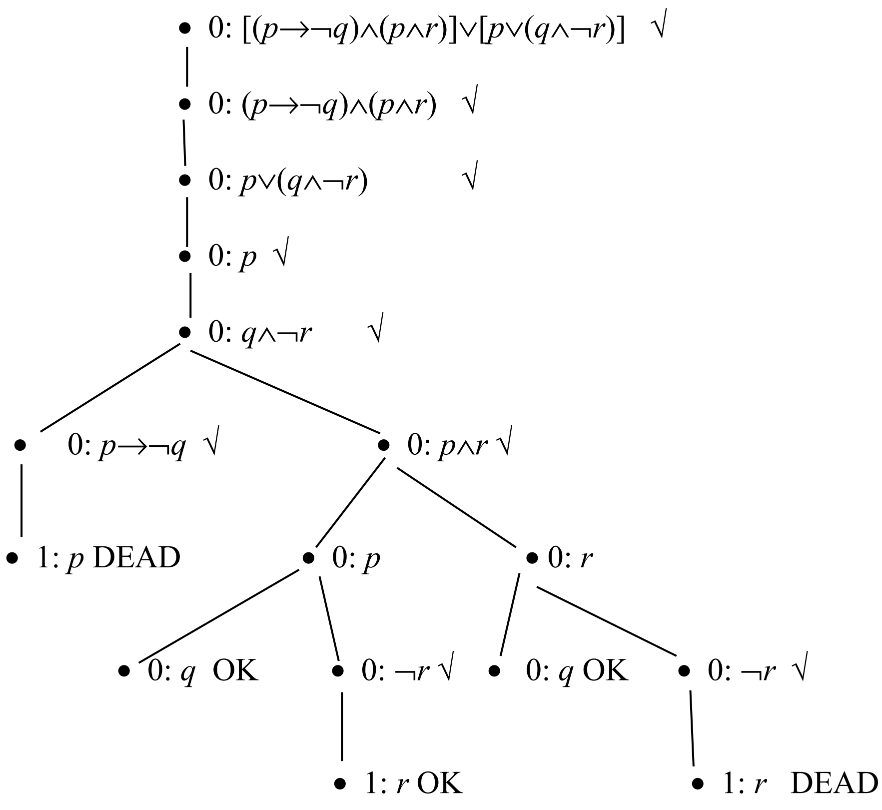
Answer (2 points): Suppose that $\sigma(\alpha)$ is not a tautology. We want to show that α is not a tautology. By the supposition, there is a valuation v with $v(\sigma(\alpha)) = 0$. Let v' be the valuation that puts $v'(p) = v(\sigma(p))$ for every elementary letter p. Then it is straightforward to show by induction (on the recursive definition of the set of formulae) that $v'(\beta) = v(\sigma(\beta))$ for every formula β. In particular, $v'(\alpha) = v(\sigma(\alpha)) = 0$, so that α is not a tautology.

(b) Without writing out a proof, identify any redundant letters in the formula $[(p\vee\neg q)\wedge(r\wedge p)]\vee(r\wedge\neg p)$ and express it in least letter-set form.

Answer (3 points): The redundant letters are p, q, and the formula is equivalent to r.

(c) Construct a semantic decomposition tree to determine whether or not the propositional formula $[(p\rightarrow\neg q)\wedge(p\wedge r)]\vee[p\vee(q\wedge\neg r)]$ is a tautology.

Answer (5 points): Sample tree below. Other trees possible with rules applied in different order. There is at least one OK branch, so the formula is not a tautology. All branches labeled OK must be closed.



QUESTION 3 (10 points)

Let α be the formula $\exists y[Py \wedge \forall y(Szx)] \rightarrow \exists z[\exists x(Rxy) \wedge \forall w(z=f(w))]$.

(a) Mark the free occurrences of *x,y,z,w* in α.

Answer (2 points):
As underlined: $\exists y[Py \wedge \forall y(S\underline{z}\underline{x})] \rightarrow \exists z[\exists x(Rx\underline{y}) \wedge \forall w(z=f(w))]$.

(b) Which of the following six substitutions are clean? $\alpha[x/x]$, $\alpha[y/x]$, $\alpha[z/x]$, $\alpha[g(y,a)/z]$, $\alpha[f(w)/y]$, $\alpha[x/y]$. *Advice* : Answer yes or no in each case; but if in doubt leave blank, as each incorrect answer loses as much as a correct answer gains.

Answer (3 points worth ½ point each but ½ off for each incorrect answer):
Yes, No, Yes, No, Yes, No.

(c) Define the concept of an *x*-variant of an interpretation for quantificational logic.

Answer: (2 points – either the short or the long formulation will do). One interpretation is an *x*-variant of another iff they agree in their domain, the values assigned to all constants, function letters and predicate letters, and in the values assigned to all variables other than the variable *x*. In other words, iff they disagree at most in the value assigned to the variable *x*.

(d) Construct the finite transform of the following formula in a domain $D = \{1,2\}$ of two individuals: $\forall x[\exists y(Rxy) \rightarrow Px]$.

Answer (3 points): $\forall x[\exists y(Rxy) \rightarrow Px] \approx [\exists y(R1y) \rightarrow P1] \wedge [\exists y(R2y) \rightarrow P2]$
$\approx [(R11 \vee R12) \rightarrow P1] \wedge [(R21 \vee R22) \rightarrow P2]$.

QUESTION 4 **(10 points)**

Express the following statements in the language of quantificational logic with identity. In each case specify a domain of discourse and a dictionary for all constants, function letters, and predicate letters employed.

(a) Any two sets that have exactly the same elements are identical.

Answer (2 points):
Domain: Sets.
Dictionary: $x \in y$: x is an element of y, $x = y$: x is identical with y.
Symbolization: $\forall x \forall y[\forall z(z \in x \leftrightarrow z \in y) \rightarrow x = y)$ or anything equivalent**.**

(b) Every set has an element with which it shares no elements.

Answer (2 points):
Domain: Sets.
Dictionary: $x \in y$: x is an element of y
Symbolization: $\forall x \exists y[y \in x \wedge \forall z(z \in x \rightarrow \neg(z \in y))]$ or anything equivalent.

(c) For any rational number greater than zero there is a smaller one greater than zero.

Answer (2 points):
Domain: Rational numbers, 0: zero, $x>y$: x is greater than y.
Symbolization: $\forall x[x>0 \rightarrow \exists y(x>y \wedge y>0)]$ or anything equivalent.

(d) If someone wins then everyone else will congratulate him.

Answer (2 points):
Domain: People. Wx: x wins, Cxy: x congratulates y, $x = y$: x identical with y.
Symbolization: $\forall x(Wx \rightarrow \forall y(\neg(y=x) \rightarrow Cyx))$, or $\forall x \forall y((Wx \wedge \neg(y=x)) \rightarrow Cyx))$ or anything equivalent.

(e) There is a problem that can be solved only if no problem can be solved.

Answer (2 points):
Domain: Problems, Sx: x can be solved.
Actually, this statement is ambiguous, according to whether we think of it with a comma after "there is a problem" or after "solved". Both readings are treated as legitimate.
First symbolization: $\exists x[Sx \rightarrow \forall y(\neg Sy)]$ or $\exists x \forall y[Sx \rightarrow \neg Sy]$ or anything equivalent. If 'problem' is treated as a predicate: $\exists x[Px \wedge (Sx \rightarrow \forall y(Py \rightarrow \neg Sy))]$ or $\exists x \forall y[Px \wedge ((Sx \wedge Py) \rightarrow \neg Sy)]$ etc.

Second symbolization: $\exists x(Sx) \rightarrow \forall y(\neg Sy)]$ or $\exists x(Sx) \rightarrow \neg \exists y(Sy)$ or anything equivalent. If 'problem' is treated as a predicate: $\exists x(Px \wedge Sx) \rightarrow \forall y(Py \rightarrow \neg Sy)]$ or $\exists x(Px \wedge Sx) \rightarrow \neg \exists y(Py \wedge Sy)$ etc.

QUESTION 5 (10 points)

(a) Give (without calculations) an interpretation in a small finite domain that shows that $\exists x \exists y(Py \to Qx)$ does not logically imply $\exists y(Py) \to \exists x(Qx)$.

Answer (3 points): Any finite domain putting $v(Q) = \varnothing$, $v(P)$ a proper non-empty subset of the domain. Simplest: domain $\{1,2\}$, $P := \{1\}$, $Q := \varnothing$.

(b) Show by natural deduction that $\forall x[\exists y(Py) \to \forall z(Rxz)]$ |- $\exists y(Py) \to \forall x(Rxx)$.

Answer (7 points):

n°	Formula	From	Rule	Depends on	Current goal
1	$\forall x[\exists y(Py) \to \forall z(Rxz)$ $]$		premise	1	$\exists y(Py) \to \forall x(Rxx)$
2	$\exists y(Py)$		supposition	2	$\forall x(Rxx)$
3					Rxx
4	$\exists y(Py) \to \forall z(Rxz)$	1	$\forall-$ proviso ok	1	Rxx
5	$\forall z(Rxz)$	2, 4	modus ponens	1, 2	Rxx
6	Rxx	5	$\forall-$ proviso ok	1, 2	$\forall x(Rxx)$
7	$\forall x(Rxx)$	1,2 |- 6	$\forall+$ proviso ok	1, 2	$\exists y(Py) \to \forall x(Qx)$
8	$\exists y(Py) \to \forall x(Qx)$	1,2 |- 7	conditional proof	1 (2 dis)	

SECTION B

QUESTION 6 (25 points)

(a) What does it mean to say that a set of connectives for propositional logic is functionally complete?

Answer (5 points): A set of connectives for propositional logic is functionally complete iff every truth-function of any finite number of arguments may be represented by a formula using only those connectives.

(b) Explain briefly why the set $\{\neg,\wedge,\vee\}$ is functionally complete.

Answer (5 points): Because we can construct an appropriate representation from the truth-table for an arbitrary formula: Take the rows that make the formula true, form the corresponding basic conjunctions, and disjoin them. If no rows make the formula true, choose say the formula $p\wedge\neg p$.

(c) Using (b), show that the set $\{\neg,\rightarrow\}$ is functionally complete.

Answer (5 points): Given $\neg,\rightarrow$, we can define $p\vee q$ as $\neg p\rightarrow q$, and then define $p\wedge q$ by de Morgan as $\neg(\neg p\vee\neg q)$, thus getting the functionally complete set $\{\neg,\wedge,\vee\}$.

(d) Outline a proof that the set $\{\wedge,\vee,\rightarrow,\leftrightarrow\}$ is not functionally complete.

Answer (5 points): We claim that these connectives cannot express negation (nor, indeed, any other truth-function that has value 1 in the top row of its truth-table). It suffices to show that no formula generated by these connectives is tautologically equivalent to $\neg p$. Consider the valuation v that gives every elementary letter the value 1. By induction (on the depth of formulae), $v(\alpha) = 1$ for every such formula, whereas $v(\neg p) = 0$.

(e) Outline a proof that the set $\{\neg,\leftrightarrow\}$ is not functionally complete.

Answer (5 points): We claim that these connectives cannot express conjunction. Clearly, any formula expressing conjunction must have at least two elementary letters, and by substitution can be made to have exactly two elementary letters. Now conjunction comes out true in exactly one of the four rows of its truth table. But it is not difficult to show (by induction on the depth of formulae) that any formula built using only $\neg,\leftrightarrow$ comes out true in an even number of rows.

QUESTION 7 (25 points)

(a) State the least letter-set, finest splitting, interpolation and compactness theorems for propositional logic.

Answer (12 points):

(i) Least letter-set: In a language including $\perp$, every formula α has a least letter-set representation, i.e. there is a (unique) least set of elementary letters such that α is tautologically equivalent to some formula built with only those letters.

(ii) Finest splitting: For any set A of formulae, there is a (unique) finest partition of the set $E(A)$ of elementary letters in formulae in A such that A is tautologically equivalent to a set of formule each of which draws all its letters from a single cell of the partition.

(iii) Interpolation: In a language including $\perp$, whenever α |- β there is a formula γ all of whose letters are common to both α and β, such that α |- γ |- β.

(iv) Compactness: Whenever an infinite set of formulae is unsatisfiable, it has an unsatisfiable finite subset. Contrapositively: If every finite subset of A is satisfiable, so is A itself.

(b) Outline a direct semantic proof of the compactness theorem for propositional logic.

Answer (13 points): Let S be any set of formulae, and suppose that all of its finite subsets are satisfiable. We want to show that S is satisfiable. Call a partial assignment v on an initial segment of the letters *good* (for S) iff for every finite $A \subseteq S$, v can be extended to a (full) assignment v_A with $v_A(\alpha) = 1$ for all $\alpha \in A$. From our supposition that every finite subset of S is consistent, it is clear that the empty partial assignment is good. Moreover, it is easy to check that whenever a partial assignment v on $p_1,\ldots,p_n$ is good, then at least one of its partial assignment extensions $v_{n+1} = v_n \cup \{(p_{n+1},0)\}$, $v_{n+1}' = v_n \cup \{(p_{n+1},1)\}$ is good. So we may define $v = \cup\{v_i : i \geq 0\}$ recursively as follows: Basis: v_0 is the empty partial assignment, Recursion step: For each $n \geq 0$, $v_{n+1} = v_n \cup \{(p_{n+1},0)\}$ if this is good, else $v_{n+1} = v_n \cup \{(p_{n+1},1)\}$. Then we can check that each v_i is a good partial assignment, v is an assignment, and since every formula is of finite length, $v(\alpha) = 1$ for every $\alpha \in S$. Thus S is satisfiable.

QUESTION 8

(25 points)

(a) What is prenex normal form? What is the prenex normal form theorem?

Answer (5 = 3+2 points):

(i) A formula of quantificational logic is in prenex normal form when it is of the form $Q_1x_1 \ldots Q_nx_n(\alpha)$ where the Q_i are quantifiers, the x_i are variables, and there are no quantifiers in α.

(ii) The prenex normal theorem states that every formula of quantificational logic is equivalent to one in prenex normal form.

(b) Let α be the formula $\exists x(Px)\rightarrow \exists x(Qx)$. Get α into prenex normal form by a succession of transformations, using relabeling of bound variables, connective translation, quantifier interchange, and vacuous quantification.

Answer (5 points):

$\exists x(Px)\rightarrow\exists x(Qx)$	-||-	$\exists x(Px)\rightarrow\exists y(Qy)$	relabeling bound variable
	-||-	$\neg\exists x(Px)\vee\exists y(Qy)$	connective translation
	-||-	$\forall x(\neg Px)\vee\exists y(Qy)$	quantifier interchange

From this point on, two answers are possible. The first continues as follows:

-||-	$\forall x(\neg Px\vee\exists y(Qy))$	vacuous quantification
-||-	$\forall x\exists y(\neg Px\vee Qy)$	vacuous quantification
-||-	$\forall x\exists y(Px\rightarrow Qy)$	connective translation.

The first continues as follows:

-||-	$\exists y\forall x(\neg Px)\vee Qy)$	vacuous quantification
-||-	$\exists y\forall x(\neg Px\vee Qy)$	vacuous quantification
-||-	$\exists y\forall x\ Px\rightarrow Qy)$	connective translation.

Exceptionally, in this example, the initial quantifiers may thus be permuted, although of course this is not in general the case. Both answers are treated as correct.

(c) Use natural deduction centring on proof by contradiction to show that $\forall x\forall y\forall z(Rxy\rightarrow\neg Rxz) \vdash \neg\exists x\exists y(Rxy)$.

Answer (15 points)

n°	Formula	From	Rule	Depends on	Current goal
1	$\forall x \forall y \forall z(Rxy \rightarrow \neg Rxz)$		premise	1	$\neg\exists x \exists y(Rxy)$
2	$\exists x \exists y(Rxy)$		supposition	2	$\perp$
3	$\exists y(Rxy)$		supposition	3	$\perp$
4	Rxy		supposition	4	$\perp$
5	$Rxy \rightarrow \neg Rxy$	1	$\forall -$ (thrice) proviso ok	1	$\perp$
6	$\neg Rxy$	4, 5	mp	1, 4	$\perp$
7	$\perp$	4,6	$\wedge +$	1, 4	$\perp$
8	$\perp$	1,4 |- $\perp$	$\exists -$ ($\perp$) proviso ok	1, 3 (4 dis)	$\perp$
9	$\perp$	1,3 |- $\perp$	$\exists -$ ($\perp$) proviso ok	1,2 (3 dis)	$\perp$
10	$\neg\exists x \exists y(Rxy)$	1,2 |- $\perp$	RAA	1 (2 dis)	