UNIVERSITY OF LONDON
IMPERIAL COLLEGE OF SCIENCE, TECHNOLOGY AND MEDICINE

EXAMINATIONS 1998

BSc Honours Degree in Mathematics and Computer Science Part III
MSci Honours Degree in Mathematics and Computer Science Part III
MSc Degree in Advanced Computing
for Internal Students of the Imperial College of Science, Technology and Medicine

This paper is also taken for the relevant examinations for the
Diploma of Membership of Imperial College
Associateship of the Royal College of Science

PAPER 3.12

LOGIC ENGINEERING
Friday, May 15th 1998, 2.00 - 4.00

Answer THREE questions

For admin. only: paper contains 4 questions

1 **Recursion and Structural Induction.**

a Define a recursive function NDQ which computes the nesting depth of quantifiers in a first-order predicate logic formula. Examples:
$NDQ(P(a)) = 0$
$NDQ(\forall x\ P(x)) = NDQ(\exists x\ P(x)) = NDQ(\forall x\ P(x) \wedge \exists x\ P(x)) = 1$
$NDQ(\forall x\ \exists y R(x,y)) = NDQ(\forall x\ (P(x) \wedge \exists y R(x,y))) =$
$NDQ(S \vee \forall x\ (P(x) \wedge \exists y R(x,y))) = 2$

b Let NC be the function that counts the number of occurrences of connectives and quantifiers in a first-order predicate logic formula. State the definition of NC and prove by structural induction: For all formulae φ: $NC(\varphi) \geq NDQ(\varphi)$.

c Define an algorithm *linear* that checks for a term t whether a variable occurs more than once in t.

Examples: $linear(x) = linear(f(x,y)) = true$,
$linear(f(x,x)) = linear(f(x,g(y,x))) = false$.

Choose an appropriate representation (data structure) for symbols and terms such that the *linear* function does not need to allocate any extra memory during its execution. (This question also appeals to your intuition as computer scientist.)

d Prove the correctness of structural induction. That means if for some statement $\wp$ and some functions O and S the base case and the induction steps have been proved then $\wp(O(\mathsf{t}))$ for all terms t and $\wp(S(\mathsf{F}))$ for all formulas F.
(Hint: Proof by contradiction: Suppose there is a term or formula t, for which the statement $\wp(O(\mathsf{t}))$ does not hold.)

a and d carry 20%, b and c carry 30% of the marks

2 **Semantics of First-Order Predicate Logic (PL1).**

Consider the following interpretation for PL1 (with binary truth value semantics):

- The domain consists of the letters in the alphabet a,b,c,d,e,f,g,h,i,j,k,l,m,n,o,p,q,r,s,t,u,v,w,x,y,z.
- The constant symbol a is mapped to the letter z.
- The constant symbol b is mapped to the letter a.
- The function symbol s is mapped to the 'next' function, i,e. $next(\mathrm{a}) = \mathrm{b}$, $next(\mathrm{b}) = \mathrm{c}, \ldots, next(\mathrm{z}) = \mathrm{a}$. (Notice that $next$ generates a closed loop.).
- The function symbol f is mapped to the *largest* function (in the above order) i.e. $largest(\mathrm{a}, \mathrm{b}) = b$, $largest(\mathrm{b}, \mathrm{a}) = b$, $largest(\mathrm{k}, \mathrm{m}) = m$ etc.
- The function symbol g is mapped to the *smallest* function, i.e. $smallest(\mathrm{a}, \mathrm{b}) = a$, $smallest(\mathrm{b}, \mathrm{a}) = a$, $smallest(\mathrm{k}, \mathrm{m}) = k$ etc.
- The predicate symbol P is mapped to the $before$ relation (in the above ordering), i.e. a $before$ b, a $before$ c, ... , a $before$ z, ... , b $before$ c, etc. x $before$ x is false for all letters x.

a For each of the following formulae check whether they are true or false in this interpretation.

i) $P(a, a)$
ii) $P(a, s(a))$
iii) $\exists x\ P(a, s(x))$
iv) $\exists x\ P(s(x), x)$
v) $\forall y \exists x\ P(x, f(x, y))$
vi) $\forall x\ (x = b \Rightarrow \exists y\ g(y, x) = x)$ ('=' is equality)
vii) $\forall x\ (P(x, b) \Rightarrow \exists y\ g(y, y) = x)$.

b Find for the formulae i-vii interpretations (maybe a different one for each formula) where the formulae have just the opposite truth value as in the interpretation above. Hint: use basically the above interpretation, but change the meaning of one or two symbols.

c What is the recipe for proving the soundness of an inference rule?

d The *abduction* rule of infernce is

From $F \Rightarrow G$ and G infer F.

(Example: from 'battery_empty $\Rightarrow$ car_does_not_start' and 'car_does_not_start' infer 'battery_empty'.)

Is this rule of infernce sound or not? If the inference rule is sound, prove it, if not, give a counter example.

a and b each carry 40% of the marks and c and d carry 10%.

Turn over ...

3 Temporal Logic, Hilbert Systems.

a Define a semantics for temporal logic with an *integer* like time structure and, besides the standard operators, a *next*-operator (in the next moment in time) and a *previously*-operator (in the last moment in time). Formulate the semantics in terms of a satisfiability relation $\models$.

b Check the formula

$$(p \wedge \mathcal{F}(p \Rightarrow \ next\ p)) \Rightarrow \mathcal{F}p$$

a) Is this formula universally valid in the temporal logic with the above semantics (integer time structure)?
b) Can the formula be interpreted in the real number time structure (the semantics given in the manuscript) and is it universally valid there as well?

c Explain what a Hilbert system is, and give a non-trivial example for a Hilbert system. Explain the intuition behind your example.

d The *deduction theorem* allows one to prove an implication $F \Rightarrow G$ by assuming F and deriving G. The *deduction theorem* holds in particular for classical propsitional logic. Does it also hold in temporal logic in the presence of the infernce rule:

from $\vdash$ G infer $\vdash \mathcal{F}$G?

If yes, give an informal argument, if not, give a counter example

each part carries 25% of the marks.

4 **Tableaux systems and many-valued logics.**

Suppose there are two sensors, sensor A and sensor B. Both can signal either 1 or 0. We define four truth values: $t_{11}, t_{10}, t_{01}, t_{00}$. t_{ij} holds if simultaneously sensor A signals i ($i = 0$ or $i = 1$) and sensor B signals j ($j = 0$ or $j = 1$). t_{10} for example means that sensor A signals 1 and sensor B signals 0.

a Define the truth tables for the connectives $\neg_4$, $\wedge_4$ and $\vee_4$ for a four-valued logic with the above truth values $t_{11}, t_{10}, t_{01}, t_{00}$.

b Define a tableaux calculus for this logic.

c Check with this tableaux calculus whether $(p \vee_4 q) \wedge_4 r$ entails $p \wedge_4 r$ or not, by analysing the tableaux for $t_{11} : (p \vee_4 q) \wedge_4 r$ and *not* $t_{11} : p \wedge_4 r$. If the entailment holds, give a tableaux proof, if not, give a counter-model (an open tableaux branch representing an assignment which yields t_{11} for $(p \vee_4 q) \wedge_4 r$ and some other truth value for $p \wedge_4 r$.)
Hint: start the tableaux with $t_{10} : p \wedge_4 r$ ($t_{10} : p \wedge_4 r$ is one of the choices for *not* $t_{11} : p \wedge_4 r$.)

a carries 30% of the marks, b 40% and c carries 30%.

End of Paper