UNIVERSITY OF LONDON
IMPERIAL COLLEGE OF SCIENCE, TECHNOLOGY AND MEDICINE

EXAMINATIONS 1998

BEng Honours Degree in Computing Part I
MEng Honours Degrees in Computing Part I
BSc Honours Degree in Mathematics and Computer Science Part I
MSci Honours Degree in Mathematics and Computer Science Part I
for Internal Students of the Imperial College of Science, Technology and Medicine

This paper is also taken for the relevant examinations for the
Associateship of the Royal College of Science
Associateship of the City and Guilds of London Institute

PAPER 1.1 / MC1.1

LOGIC

Friday, May 1st 1998, 4.00 - 5.30

Answer THREE questions

For admin. only: paper contains 4 questions

1 In each of (a), (b) and (c) below is an English description of an operation. There is also an incorrect and/or incomplete attempt to formalise the operation using logic. You may assume the pre-conditions are correct.

a removesome :: [Char] -> [Char] -> [Char]
-- pre: none
-- post: all occurrences in ys of characters in xs (and no others) are removed,
-- leaving zs, where zs = removesome xs ys.
-- $\exists u[\text{merge}(u,zs,ys) \wedge \forall k[\text{in}(k,u) \rightarrow \text{in}(k, xs)]]$

b middle :: [Int] -> Int
-- pre: $\exists x,y,z[\ \text{in}(x,xs) \wedge \text{in}(y,xs) \wedge \text{in}(z, xs) \wedge x \neq y \wedge x \neq z \wedge y \neq z]$
-- post: m is a value in xs that is neither the largest nor the smallest element in
-- xs, where m = middle xs.
-- $\exists xy[\text{in}(x,xs) \wedge \text{in}(y,xs) \wedge m \geq x \wedge m \leq y]$

c procedure insert (inval: int, var A: array 1 .. * of int)
% pre: ascending (A) $\wedge$ upper(A) ≥ 1
% post: A' is ascending, and, either inval is in A already and A' = A, or,
% inval has been inserted into A to form A', by moving elements >
% inval to make space for it and removing the element at index
% A[upper(A)] . A' refers to the array A at the end of the
% procedure and upper(A) the maximum index of A.
% $(\forall i\ [\ 1 \leq i \leq \text{upper}(A) \rightarrow A[i] = A'[i]\ ]\ ) \vee$
% $(\exists m\ [1 \leq m \leq \text{upper}(A) \wedge$
% $\forall j[m \leq j \leq \text{upper}(A) \rightarrow A'[j] = A[j\text{-}1]])$

In each of the cases (a), (b) and (c):

i) Give an example input and output that satisfies the given pre-condition and post-condition, but which ***does not*** satisfy the intention given in the English description.

Explain your answer. ***In particular*** , explain the mistakes in the given logic. This will involve translating the logic into natural English.

ii) Amend the given logic post-condition so that it correctly formalises the intention of the described operation.

You may use the following predicates without definition:
merge(x,y,z) , which holds for lists x, y and z iff z is a permutation of x ++y such that the relative order of elements in x and y is retained in z,
ascending(x) , which holds iff the elements in array x are in ascending order,
in(x,y) , which holds iff element x is in list y, together with the standard predicates ≤, ≥, =, <, >. Any other predicates you use must be defined.

The three parts carry, respectively, 30%, 30%, 40% of the marks.

2 a The figure below shows a 4 x 4 board on which objects can be placed. There are available 5 objects, called a,b,c,d,e. Each object is either red (R) or green (G) but not both. The object a, which is known to be red, is already placed on the board at position (1,3).

		aR	

Find a position and colour for the other 4 objects (b,c,d,e) such that the sentences in (1) - (4) below are simultaneously true for the domain {a,b,c,d,e}. *Justify your answer carefully.*

The intended meanings of the predicates are as follows:
red(x) means x is red; green(x) means x is green;
next(x,y) means x lies on the same column or in the same row as y;

(1) $\neg\,(\forall x.\text{red}(x)\ )$

(2) $\exists x,y,z[\ \text{next}(x,y) \wedge \text{next}\ (x, z) \wedge \neg\ \text{next}(y,z)\ ]$

(3) $\forall x[\text{red}(x) \text{ -> } \neg\,\exists\ y\ [\ \text{green}(y) \wedge \text{next}(x,y)]\ ]$

(4) $\forall x,y[\ \text{next}(x,y) \text{ -> } (\text{red}(x) \wedge \text{red}(y)) \vee (\text{green}(x) \wedge \text{green}(y))]$

b What is the signature (S) of the sentences in (5) - (7) below?
Complete the following structure for S with domain {integers $\geq$ 1} such that it makes all sentences (5) - (7) true. *Explain your answer carefully.*

P(x,y) means $x < y$; Q(x,y) means $x \geq y$

(5) $\exists x\forall y.Q(y,x)$

(6) $\forall x,y[\ P(x, f(y)) \rightarrow Q(y, x)]$

(7) $\forall x\exists z[\ Q(x, z) \wedge P(x, f(z))]$

c Rewrite formally using natural deduction the following outline proof of

$$\exists y\forall x[P(x) \leftrightarrow x = y] \ \vdash \exists z.\ P(z) \wedge \forall uv[P(u) \wedge P(v) \rightarrow u = v]$$

Proof:
Let Y be an arbitrary y, such that $\forall x[P(x) \leftrightarrow x = Y]$ (*).
Since Y=Y, we obtain P(Y) from (*) and hence $\exists z.\ P(z)$.
Next we show for arbitrary U and V that $P(U) \wedge P(V) \rightarrow U = V$.
Suppose $P(U) \wedge P(V)$.
Hence, from (*), U = Y and V = Y.
Hence U = V as required and the proof is done.

The three parts carry, respectively, 35%, 35%, 30% of the marks.

Turn over ...

3 *Without rewriting by equivalences* use natural deduction to show the following:

a $B \to D,\ A \to (B \vee C),\ (G \to D) \to E,\ \neg (C \wedge G) \ \vdash A \to E$

b $g(a) = a,\ \forall x[\ (\exists y.P(y,x)\ ) \to x{=}a\ ],\ \forall u.P(u,u)\ \ \vdash \forall v\ [g(v) = v]$

c $\exists z.P(z),\ \forall u,v[\ P(u) \wedge P(v) \to u = v\ ] \vdash \exists y \forall x[\ \neg(P(x) \wedge x \neq y)\ ]$

The three parts carry, respectively, 35%, 30%, 35% of the marks.

4 a Show, using only equivalences, that $(X \vee Y) \wedge (\neg X \vee Y) \equiv Y$.
State each of the equivalences you use.

b Define the relation $\models$ for propositional sentences.
The following two facts about $\models$ will be used in part c).

Fact 1. $X \models A \vee X$, for any formulas A and X.

Fact 2. If $X1 \models Y1$ and $X2 \models Y2$
then $X1 \wedge X2 \models Y1 \wedge Y2$, for any X1, X2, Y1, Y2.

Show, using a truth analysis for X and A, or otherwise, that Fact 1 is true.

c Using part (a), Facts 1 and 2 of part (b) and the commutativity and associativity of $\vee$, show carefully (giving every step) that

$$(A \vee B) \wedge (\neg A \vee C) \models B \vee C.$$

d Show by a truth analysis that $B \vee C \not\models (A \vee B) \wedge (\neg A \vee C)$.

e State the property between $\models$ and $\vdash$ that allows one to conclude from part c) (the derived rule) that $(A \vee B) \wedge (\neg A \vee C) \vdash B \vee C$.
What is this property called?

Using the above derived rule, the natural deduction rules $\forall$E and $\wedge$I, and the fact that $X \vee X \equiv X$, or otherwise, show

$$\forall x[X(a,x) \vee Y(x,b)],\ \forall y,z[\neg X(y,y) \vee Y(y,z)\ ] \vdash Y(a,b)$$

The five parts carry, respectively, 15%, 15%, 25%, 20%, 25% of the marks.

End of paper

Department of Computing Examinations

Paper 1.1 = MC1.1 1998

Additional Aide Memoir

Useful Heuristics for Natural Deduction

(i) Work backwards from the goal.

(ii) Apply $\exists E$ as soon as it is possible in a proof.

(iii) Given data $X \rightarrow Y$, then to show Y, try to show X.

(iv) Given data $\neg X$, then to show $\perp$, try to show X.

(v) Remember the useful rule that anything can be derived from $\perp$.

Useful Equivalences

(i) $X \vee (Y \vee Z) \equiv (X \vee Y) \vee Z$

(ii) $X \vee Y \equiv Y \vee X$

(iii) $X \wedge (Y \wedge Z) \equiv (X \wedge Y) \wedge Z$

(iv) $X \wedge Y \equiv Y \wedge X$

(v) $X \wedge (Y \vee Z) \equiv (X \wedge Y) \vee (X \wedge Z)$

(vi) $(Y \vee Z) \wedge X \equiv (Y \wedge X) \vee (Z \wedge X)$

(vii) $X \vee (Y \wedge Z) \equiv (X \vee Y) \wedge (X \vee Z)$

(viii) $(Y \wedge Z) \vee X \equiv (Y \vee X) \wedge (Z \vee X)$

(ix) $X \rightarrow Y \equiv \neg X \vee Y$

(x) $(X \vee Y) \rightarrow Z \equiv (X \rightarrow Z) \wedge (Y \rightarrow Z)$