UNIVERSITY OF LONDON
IMPERIAL COLLEGE OF SCIENCE, TECHNOLOGY AND MEDICINE

EXAMINATIONS 1997

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

Wednesday, May 7th 1997, 4.00 - 5.30

Answer THREE questions

For admin. only: paper contains 4 questions

1 a Give *pre* and *post* conditions in logic for the following operations on *finite lists of integers* ≥ 0. You may make use of the predicate goodlist(x), which holds when x is a finite list of integers ≥0.

i) AddFront :: num -> [num] -> [num]
% The result is formed by placing the given integer at the front of the given list.

ii) RemoveLast :: [num] -> [num]
% The result is the list obtained by removing the last integer from the given list.

b Consider the following specification of the operation Once on finite lists of integers ≥0 which removes all but one occurrence of each integer in the given list.

```
Once :: [num] -> ( [num], [num] )
|| Pre:   goodlist (xs)
|| Post:  merge(z,r,xs) ∧ ∀ x [ x ε r → x ε z] ∧
||          ¬ ∃ p,u,v,w [ z = u ++ [p] ++ v ++ [p] ++ w]
||          where ( z, r ) = Once xs
```

(merge(z,r,xs) holds iff xs is a permutation of z ++r such that the relative order of elements in z and r is retained in xs.)

The following input list and pair of output lists satisfies *all three* conjuncts of the specification:

xs = [1, 2, 3, 2]   z = [1, 3, 2]   r = [2]

For each conjunct C of the post-condition, give an example of input list and pair of output lists that satisfies the other two conjuncts but does not satisfy C.

c A finite list of integers ≥0 (no more than 100 in the list) is represented in Turing by the type

listrep = array 1..100 of int

in which the elements of the list are the first n elements of the array and the remaining 100-n array elements are all -1.

Give the *pre* and *post* conditions for the following operations of part (a), now redefined for arguments of type listrep.

i) AddFront (n: int, var xs: listrep)

ii) RemoveLast(var xs:listrep)

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

2 a The following is the outline of a proof that there can be at most one smallest element in a set ordered by a relation R that is transitive and irreflexive.

Proof:
Given R is transitive and irreflexive
(i.e. given transitive_R and irreflexive_R).
Suppose there are at least two smallest elements of R.
Call them a and b.
Then, since a is a smallest element, R(a,b).
Similarly, R(b,a).
By transitivity R(a,a) - a contradiction
(since $\neg$ R(a,a) by irreflexivity).
Hence there are not two smallest elements of R.

The proof uses the following definitions:

$$\text{smallest\_element\_of\_R}(x) \text{ iff } \forall u\,.R(x,u)$$
$$\text{irreflexive\_R iff } \forall x \neg R(x,x)$$
$$\text{transitive\_R iff } \forall x,y,z\,[R(x,y) \wedge R(y,z) \rightarrow R(x,z)]$$

i) State in logic the property that it is not the case that there are at least two smallest elements of R.

(Your answer should begin $\neg \exists \ldots$)

ii) Formalise the proof in natural deduction and complete any missing steps.

b Use natural deduction to show

$$\forall x, y[g(x, f(y)) = f(g(x, y))],$$
$$\forall y\, [\, L(y) \text{ -> } f(y) = y],$$
$$\forall y\, [\, f(y) = y \text{ -> } L(y)]$$
$$\vdash \forall u\, [\, L(u) \rightarrow L(g(e,u))]$$

(**Hint:** Work backwards from the goal.)

c Prove the following using natural deduction:

Given R is a reflexive, symmetric and anti-symmetric relation
show $\forall x,y\ [(R(x,y) \text{ -> } x=y\ ) \wedge (\ x = y \text{ -> } R(x,y))]$

The proof should use the definitions:

$$\text{reflexive\_R iff } \forall x.R(x,x)$$
$$\text{symmetric\_R iff } \forall x,y\,[\,R(x,y) \rightarrow R(y,x)]$$
$$\text{antisymmetric\_R iff } \forall x,y\,[R(x,y) \wedge R(y,x) \rightarrow x = y]$$

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

Turn over ...

3 ai) State the ∃- elimination rule of natural deduction.

How does it differ from the typed ∃- elimination rule?

ii) Translate the statement $\forall$ x: D3 $\exists$ y: D2 . f(y) = x into a non-typed form.

You should use the predicates *is_D2(x)*, and *is_D3(x)* , which hold when x is of type D2 and of type D3 respectively.

bi) Show, using natural deduction,

$$\forall x \exists y . f(y) = x$$
$$\forall u \exists v . g(v) = u$$
$$\vdash \forall w \exists z . f(g(z)) = w$$

ii) Given that the statement $\forall$ x $\exists$ y . f(y) = x is the definition of what it means for a function f: D -> D to be an onto-function, what property of onto-functions is proved in part (bi)?

c Show that (1) does not logically imply (2), where

(1) $\forall x\ \exists y\,[\ f(y) = x]]$

(2) $\forall w\ \exists z\,[\, f(\, g(z)\,)= w]$

by finding an interpretation with domain {0,1} for f and g that makes (1) true and (2) false. Explain your answer.

(**Hint**: Choose a meaning for g so that it makes $\forall$ u $\exists$ v . g(v) = u false.)

The three parts carry, respectively, 25%, 45%, 30% of the marks.

4 a Show by natural deduction

$$a \wedge w \rightarrow p, \quad \neg i \rightarrow a, \quad \neg w \rightarrow m, \quad \neg p, \quad e \rightarrow \neg i \wedge \neg m \quad \vdash \neg e$$

Do not rewrite any of the assumptions by equivalences.

b Show by a truth analysis that

$((p \wedge q) <-> p) <-> ((p \vee q) <-> q)$ is always true.

c Use the equivalences

(1) $X -> (Y -> Z) \equiv (X \wedge Y) -> Z$

(2) $X \wedge (X -> Y) \equiv X \wedge Y$

(3) $X -> True \equiv True$

(4) $X -> X \equiv True$

together with the associativity and commutativity of $\wedge$ to show by equivalences that

$((p -> q) \wedge (q -> r)) -> (p -> r)$ is always true.

di) Explain why, for propositional sentences X and Y,

$X <-> Y$ is always true iff $X \equiv Y$.

ii) Use the results of parts (b) and (di), together with the associativity and commutativity of $<->$, to show that

$p <-> q \equiv (p \wedge q) <-> (p \vee q)$ is always true.

(**Hint**: Associativity and commutativity of $<->$ can be used to show that $(x <-> y) <-> (u <-> z)$ is equivalent to $(u <-> (x <-> y)) <-> z$.)

The four parts carry, respectively, 30%, 15% , 25%, 30% of the marks.

End of paper