UNIVERSITY OF LONDON
IMPERIAL COLLEGE OF SCIENCE, TECHNOLOGY AND MEDICINE

EXAMINATIONS 1996

MEng Honours Degrees in Computing Part IV
MSc Degree in Foundations of Advanced Information Technology
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 City and Guilds of London Institute

PAPER 4.90

FUNCTIONAL PROGRAMMING—FOUNDATIONS
Wednesday, May 1st 1996, 10.00 - 12.00

Answer THREE questions

For admin. only: paper contains
4 questions
2 pages (excluding cover page)

1a i Define the relation $\#$

ii Show that $\lambda xy.yx \# \lambda x.x$

iii Show that $\neg \exists F s.t. \forall M,N[F(MN) = N]$

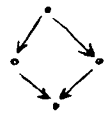
b Let M be the term $(\lambda x.xx)((\lambda x.x)(\lambda x.x))$

i Reduce M to β-normal form by standard reduction.

ii Reduce M to β-normal form by a non-standard reduction.

iii Which is the head redex of M?

iv Which is an internal redex of M?

c A "pearl" is the graph  and a "diamond" is the graph 

i Define a term whose β-reduction graph is a diamond

ii Define a term which has pearls and diamonds in its β-reduction graph.

2a Write the definitions of

i The Krivine machine.

ii The Eager machine.

b Let M be the term

$$(\lambda xyz.yy(\lambda u.ywy(\lambda za.azy))yax)(\lambda x.x)(\lambda x.x)(\lambda x.x)$$

i Translate M into the De Bruijn notation.

ii Evaluate $(\lambda x.xx)(\lambda x.x)$ in the Krivine machine.

c Can you find terms M_1, M_2, M_3, M_4 (if yes, define them) such that:

i M_1 has normal form and its evaluation in the Krivine machine is longer than its evaluation in the eager machine?

ii M_2 has normal form and its evaluation in the eager machine is longer than its evaluation in the Krivine machine?

iii The evaluation of M_3 in the Krivine machine terminates and it doesn't in the eager machine?

iv M_4 has the opposite property of M_3?

3a Define the Call-by-name operational semantics for PCF

b i Define the Call-by-value operational semantics for PCF

ii Find a PCF program that converges according to the Call-by-name operational semantics but doesn't according to the Call-by-value semantics.

c Consider the extension PCFpar of PCF with a new constant `POR`: **bool** $\rightarrow$ **bool** $\rightarrow$ **bool** which implement the Parallel Or function:

i Write a PCFpar term which implements the `w-POR` function, where `w-POR` is the function of type **bool** $\rightarrow$ **bool** $\rightarrow$ **bool** which is true whenever at least one of its arguments is true and undefined otherwise.

ii Write a PCFpar term of type **bool** $\rightarrow$ **bool** $\rightarrow$ **bool** $\rightarrow$ **bool** which implements the "truth majority" function, i.e. the function that returns True if at least two of its arguments are True and is undefined otherwise.

4a Using the the denotational semantics of PCF write:

i The interpretation of the following PCF constants: `cond`, `succ`, Y.

ii The interpretation of PCF terms.

b i State the Fixpoint theorem on domains.

ii Prove that if D, E are finite domains then the set of continuous functions from D to E is exactly the set of monotone functions from D to E.

c Consider the language W.

i Write the denotational semantics of commands in W.

ii Write the interpretation of the command

`while` $\neg$`(x =0 ) do x:= x-1`

in the denotational semantics of W.

End of paper