UNIVERSITY OF LONDON
IMPERIAL COLLEGE OF SCIENCE, TECHNOLOGY AND MEDICINE

EXAMINATIONS 1996

BEng Honours Degree in Computing Part III
BSc Honours Degree in Mathematics and Computer Science Part III
MSc Degree in Computing Science
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
Associateship of the Royal College of Science

PAPER 3.25

FUNCTIONAL PROGRAMMING—TECHNOLOGY
Monday, April 29th 1996, 10.00 - 12.00

Answer THREE questions

For admin. only: paper contains
5 questions
3 pages (excluding cover page)

1a Explain what is meant by the term *data parallelism* and discuss how this model is realised by the SIMD (Single Instruction Multiple Data) and SPMD (Single Program Multiple Data) computational models. Discuss the types of parallel machines that suit these two models.

b Use SCL co-ordination forms to specify a SPMD parallelisation of the following sequential program for execution on an n-processor distributed memory parallel computer:

```
Function computepi (int : interval)
    \* approximation to pi *\
    int i;
    double sum, width, x;
    sum =  0.0;
    width = 1.0/interval;
    for  i =0 to  interval step 1 do {
       x = ( i + 0.5 ) * width;
       sum = sum + (4.0/(1.0 + x*x));
    }
```

2a Explain the terms *co-ordination language* and *skeleton* in the context of parallel programming languages and discuss how these ideas are combined in the SCL language.

b Explain how the requirements for abstraction and efficiency are reconciled in the SCL co-ordination framework.

c Recursive bisection takes a divide-and-conquer approach to partition a domain (two dimensional array) into subdomains of approximately equal computational cost . The domain is first cut in one dimension to yield two subdomains. Cuts are then made recursively in the new subdomains. Given a grid as a two dimensional array, define an SCL program to partition the grid into n row blocks based on the recursive bisection method. You may assume a partitioning function 'balanced_partition' is available that divides an array into two row blocks of equal predicted computational cost.

3a Explain what is meant by the *meaning* and *behaviour* of a functional language program. Using an example explain how *program transformation* can be used to safely improve the performance of functional programs executing on parallel machines.

b Given the following function definitions

```
map f [] = []
map f (x:l) = (f x) : (map f l)

fold1 f (x:[]) = x
fold1 f (x:y:l) = f x (fold1 f (y:l))

(f o g) x = f (g x)
```

i) Prove that

```
map o (fold1 (o) ) = (fold1 (o)) o (map map)
```

Be sure to state clearly any lemmas you assume.

ii) In terms of parallel execution give a behavioural interpretation of both sides of the equation in i) and explain, with an example, how it can be said that the right hand expression corresponds to pipelined execution

4a In terms of program transformation systems explain what is meant by the terms *partial correctness* and *completeness.* Illustrate your answer by considering both the unfold/fold transformation system and transformations presented as a set of algebraic equivalences. What other properties do you consider are important for the practicality of program transformation?

b Given the following function definitions

```
foldr f b [] = b
foldr f b (x:l) = f x (foldr f b l)

foldl f acc [] = acc
foldl f acc (x:l) = foldl (f acc x) l
```

i) State what properties the following equivalences between foldr and foldl depend upon and prove these equivalences.

```
f (foldl s f l) b = f s (foldr f b l)

    foldl f s l = foldr f b l
```

ii) Using an example explain why programs expressed using foldl can often be executed more efficiently than equivalent programs expressed using foldr

Turn over ....

5a Explain what is meant in SCL by a *configuration* and discuss how this construction abstracts the physical distribution of data around a parallel machine. What constructs are available in SCL for constructing and manipulating configurations. Illustrate your answer by showing how array-structured data-parallel programs can be specified in SCL.

b Given the following definitions in SCL

```
distribution (f, p) (g, q) A B = align  (p o partition f A) (q o partition g B)

Matrix_add p A B =

    C where
        C  = gather  o (map SEQ_ADD)  (distribution f1 d1)
        f1 = [ (row_block p, id), (row_block p, id), (row_block p, id) ]
        d1  = [A, B, C]
```

Where SEQ_ADD is an imperative subroutine that pointwise adds together two arrays.

i) Use diagrams to illustrate the data distribution and re-distribution involved in the evaluation of the following expression

```
Matrix_add A (Matrix_add B C)
```

ii) Identify any unnecessary data movement that occurs during this execution and give an SCL program that adds the three matrices together in parallel without any unnecessary data re-distribution.

End of Paper