PAPER CODE NO. **COMP308**	EXAMINER : Igor Potapov DEPARTMENT : Computer Science Tel. No. 0151 794 7240

UNIVERSITY OF LIVERPOOL

MAY 2007 EXAMINATIONS

Bachelor of Arts : Year 3
Bachelor of Science : Year 3
Bachelor of Science : Year 4
No qualification aimed for: Year 1

Efficient Parallel Algorithms

TIME ALLOWED : Two and a Half Hours

INSTRUCTIONS TO CANDIDATES

Answer **FOUR** questions.

If you attempt to answer more questions than the required number of questions (in any section), the marks awarded for the excess questions answered will be discarded (starting with your lowest mark).

QUESTION 1

a) Describe the Parallel Random Access Machine (PRAM) model of parallel computation. Why is the PRAM model useful in the conceptual design of parallel algorithms? Explain how to perform the activation of p processors in a constant number of steps on the concurrent read PRAM model or in $\log_2(p)$ steps on the executive read PRAM model.
(15 marks)

b) Explain the concept about speedup called Amdahl's Law. Assume that five percent of the operations in a given computation must be performed sequentially. What is the maximal speedup can be achieved, regardless of how many processors are used?
(5 marks)

c) Define the complexity classes NC and the class of P-complete problems.
(5 marks)

QUESTION 2

a) What is the main difference between *Simultaneous Substitution* and *Pebble Game* methods? Apply the *pebbling game* technique to the evaluation of the following arithmetic expression: $380 - ((5 + (7 * 11)) * (18/((2 * 4) - 2)))$.
(10 marks)

b) Suppose that you have constructed a network that merges two sorted input sequences

$$[a_1, \ldots, a_{\frac{n}{2}}] \text{ and } [a_{\frac{n}{2}+1}, \ldots, a_n]$$

into one sorted output sequence $[a_1, \ldots, a_n]$. Explain how to build a sorting network of any size if you have the merging networks for any possible value of n.
(10 marks)

c) Suppose that you have constructed a parallel algorithm to solve a problem A in $O(n^3)$ time with n^3 number of processors and the optimal sequential algorithm solves A in $O(n^5)$. Is the parallel solution cost-optimal?
(5 marks)

QUESTION 3

a) Suppose that you have constructed a CRCW PRAM algorithm to solve problem A in $O(t(n))$ time. Now when you begin to consider a solution to problem A on a CREW PRAM, what do you already know about an upper bound on the running time to solve this problem on a CREW PRAM? Explain the answer providing the construction for achieving the upper bound solution.
(10 marks)

b) What is the "Zero-one-principle"? Use it to show that the Shearsort algorithm is correct.
(10 marks)

c) What is a Parallel Virtual Machine? Explain how it works.
(5 marks)

QUESTION 4

a) Give an efficient algorithm to solve the following problem. For a linked list with n links, determine for each element the number of elements in the list that follow it. Explain how to solve prefix sum problem using your algorithm.
(10 marks)

b) Give a parallel implementation of Warshall's algorithm for a PRAM model. Run your algorithm for computing transitive closure of the following adjacency matrix:

$$\begin{pmatrix} 1 & 0 & 1 & 1 & 0 \\ 0 & 1 & 0 & 1 & 0 \\ 0 & 0 & 1 & 0 & 1 \\ 0 & 0 & 0 & 1 & 0 \\ 0 & 1 & 0 & 0 & 1 \end{pmatrix}$$

(10 marks)

c) Explain the difference between the "store and forward" and the "cut-through" routing techniques.
(5 marks)

QUESTION 5

a) Describe the hypercube network of processors and the method of assigning unique binary vectors to its nodes. Explain the principle of broadcasting algorithm on a hypercube using store and forward technique.
(10 marks)

b) Assume that you have a program A that computes minimum of n numbers stored in the array in $O(1)$ time with n^2 number of processors. Construct a program B that solves the same problem with $n^{1+\frac{1}{2}}$ processors in $O(1)$ time and a program C that computes minimum of n numbers with $n^{1+\frac{1}{4}}$ processors in $O(1)$ time.
(10 marks)

c) Explain the difference between fine-grained and coarse-grained parallel algorithms. What is the major advantage of the coarse-grained parallel algorithms?
(5 marks)