Exam Parallel Programming 13 February 2007
Department of Computer Science, Faculty of Sciences

1. (a) What are the diameter, bisection width, and number of edges per node of a 3-dimensional mesh (lattice) topology that has 125 nodes in total?

   (b) What is a processor array (vector machine)? Explain how it differs from a multiprocessor and why it is programmed differently.

2. Someone measures the performance of a parallel program on two different clusters; one cluster uses 1 GHz CPUs, the other cluster uses 3 GHz CPUs, but except for this difference the clusters are identical (same number of CPUs, amount of memory, local network, etc.). On which cluster does the parallel program obtain the best speedups (relative to a sequential program running on the same CPU)? Explain your answer with simple calculations.

3. Explain how an efficient multicast primitive can be implemented on Myrinet by changing the software (firmware) of the Myrinet network interface cards. Why is such a multicast primitive faster than a normal spanning-tree multicast primitive on top of point-to-point message passing over Myrinet?

4. Consider a parallel program in which each process gets a large piece of work, executes it, and then terminates. This program has to be extended with the capability to terminate (kill) processes prematurely. In the extended program, one process (the master) should be able to send a termination message to all other processes (the slaves). Each slave process should terminate as soon as it receives such a termination message (e.g., by invoking the 'exit' system call). The termination messages may arrive at any point in time during the execution of the slaves, making it somewhat difficult to receive and handle such messages.

   (a) How would you implement the receipt of such termination messages if the program is written in SR?

   (b) How would you implement the receipt of such termination messages if the program is written in MPI?

5. Implement a shared queue data structure using Linda's tuple space. The queue should represent each queue element in a separate tuple, and it should support operations to enqueue an item at the tail of the queue and to dequeue an item from the head of the queue; the latter operation should block if the queue is empty. For the Tuple Space operations, indicate clearly which parameters are actuals and which are formals.

6. What are the most important advantages of HPF over message-passing languages?

7. The parallel Barnes-Hut algorithm for hierarchical N-body problems can use several different load balancing mechanisms. One approach is to partition the physical space into equal parts and assign all bodies in this part of the space to one processor (so, different processors work on different parts of the physical space). Explain how good or

bad both the load balancing and the communication overhead of this approach are. For each of them, explain whether they are better or worse than for the costzone approach.

8. The parallel retrograde analysis program used to solve Awari communicates very much. It sends about 1 Petabit of data over the local network (Myrinet) during the entire execution. Nevertheless, the parallel program still performs well. Explain why this large communication overhead does not prevent the Awari program from obtaining a high performance.

9. Explain why divide-and-conquer parallelism is a good model for programming computational grids. Why do divide-and-conquer systems like Satin obtain nearly the same speedup on a 64-node local cluster as on a wide-area grid with 4 clusters of 16 nodes (64 nodes in total), despite the fact that the wide-area network between the clusters is about 1000 times slower than the local network (e.g., Myrinet) used within a cluster?

Points

1a	1b	2	3	4a	4b	5	6	7	8	9
5	5	10	10	5	5	10	10	10	10	10

Total: 90 (+ 10 = 100)