Exam Parallel Programming 22 October 2009
Department of Computer Science, Faculty of Sciences

Your answers should be to the point: address the questions and omit information that is not asked for.

1. (a) What is a "fat tree" topology? What problem of normal (non-fat) tree topologies does it try to solve?

   (b) Flynn's taxonomy describes 4 classes of machine architectures. The class of MISD architectures is essentially empty, so there are no existing machines with an MISD architecture. Why is this the case? How would such a machine work?

2. Describe 3 reasons why cluster computers are currently dominating the top 500 supercomputer list, whereas 10 years ago there were hardly any clusters in the top 500.

3. 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?

4. A problem with asynchronous message passing is that the buffer space for storing outgoing messages is finite, so the sender still may have to be blocked if the buffer space fills up. This is confusing to programmers, who assume that asynchronous sends continue immediately and don't block. How does MPI deal with this problem?

5. Consider the following HPF program fragment

```
!HPF$ PROCESSORS pr(3)
integer A(8), B(8)
!HPF$ ALIGN B(:) WITH A(:)
!HPF$ DISTRIBUTE A(BLOCK) ONTO pr
FORALL (i=1,7) A(i) = B((i+1)/2)
```

   Explain how many messages will be generated for the FORALL statement.

   For clarity: the FORALL statement is equivalent to the C statement

```
for (i = 1; i < 8; i++) A[i] = B[(i+1)/2]
```

6. An astronomer wants to use a parallel Barnes-Hut program based on the "costzone" approach to simulate a spectacular new type of galaxy where stars can move in one timestep from their current position to a completely different (remote) part of the galaxy. Explain how well the costzone approach will perform in this case. Will its optimizations still work efficiently?

7. The Transposition-Driven Search (TDS) algorithm communicates very much but nevertheless it obtains nearly perfect speedups compared to a sequential algorithm that also uses transposition tables. Explain why this large communication overhead does not prevent the TDS algorithm from obtaining a very high performance.

8. The IPL (Ibis Portability Layer) is a message passing library designed for grids, whereas MPI is designed for clusters and supercomputers. Explain which functionality IPL provides that MPI does not have.

9. Give three reasons why 'data parallelism' is generally a better approach than 'task parallelism' for low level image processing applications.

Points

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

Total: 90 (+ 10 = 100)