Exam Parallel Programming 14 January 2010
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) Explain what a hypercube topology is. What are the diameter and bisection width of a hypercube with dimension N? (Give formula's.)
   (b) What is the main disadvantage of the hypercube topology?
2. What is Flynn's taxonomy? Which classes of computer systems does Flynn define?
3. Given below is a sequential algorithm that makes a fixed number of sweeps over an N-by-M array and during each sweep updates elements:

```
float G[1:N, 1:M], Gnew[1:N, 1:M];
for (step = 0; step < NSTEPS; step++) {
  for (i = 3; i < N-1; i++) {
    for (j = 3; j < M-1; j++) {
      Gnew[i,j] = (G[i,j] + G[i-1,j] + G[i-2,j] +  G[i+1,j] + G[i+2,j]
                   + G[i,j-1] + G[i,j-2] + G[i,j+1] + G[i, j+2]) / 9;
    }
  }
  G = Gnew;
}
```

   What is the communication scheme if this algorithm is parallelized by partitioning the array row-wise over P processors (i.e., giving each processor N/P consecutive rows)? Make clear which data the processors will exchange.
4. MPI has many forms of message passing:
   (a) It provides both blocking and nonblocking sends (not to be confused with synchronous and asynchronous sends); describe the differences and advantages/disadvantages of these two forms of transmission.
   (b) It provides four different modes for sending: standard, buffered, synchronous and ready mode. Again describe their differences and advantages/disadvantages.
5. Explain how a select statement can be used to implement a server process with the following property. The server contains an integer variable X (initialized to zero) and accepts two different types of messages, to increase resp. decrease the value of X. For both messages, the server returns a message to the sender with the new value of X. The value of X, however, should always be between zero and ten. Messages that try to decrease X below zero or increase it above ten should therefore not be serviced immediately but should be delayed until X has an appropriate value. Implement this server process with a select statement, using (clearly explained) pseudo-code.

6. Automatic parallelization of sequential programs is extremely difficult. Languages that try to do (more or less) automatic parallelization therefore make compromises, such as

   - They make restrictions on the source program
   - They restrict the kind of parallelism that can be used
   - They use a semi-automatic approach and let the programmer still do part of the work.

   Discuss which compromises or restrictions HPF(High Performance Fortran) makes.

7. The parallel Barnes-Hut algorithm for hierarchical N-body problems tries to improve the *data locality* of the parallel program. Explain why this is important and how the algorithm manages to improve data locality.

8. Consider the following four different parallel search algorihms:
   1 IDA* (a search algorithm based on work-stealing) without a shared transposition table
   2 IDA* with a shared replicated transposition table
   3 IDA* with a shared partitioned transposition table
   4 Transposition-Driven Search (TDS)

   (a) The four algorithms differ in the number of search-nodes they analyse (expand and evaluate) if they are run on a large-scale parallel machine. Rank the four algorithms in order of increasing number of nodes searched and explain your ordering.

   (b) The four algorithms also have different communication overheads for handling transposition table lookups and stores. Discuss for each algorithm from what type of communication overhead it suffers.

9. The direction of research in the field of Grid Computing is driven by a certain 'visionary aim', or 'promise'.

   (a) Explain what is meant by this 'promise of the Grid'.

   (b) Explain in what ways this promise is realized by the Parallel-Horus (or Jorus) framework for multimedia computing and its Ibis-based extensions. Discuss at least three techniques (solutions) that this framework provides to realize the promise of the grid.

Points

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

Total: 90 (+ 10 = 100)