Choose five questions.

(1) (i) [5 points] Define: task parallelism and data parallelism;
   (ii) [5 points] give a practical example with division of the work for each;
   (iii) [10 points] include pseudo-code illustrating both approaches.

(2) (i) [2 points] What is a process?
   (ii) [10 points] List five entities of which it consists.
   (iii) [2 points] How does a thread differ from a process?
   (iv) [3 points] Name two resources that can be shared by threads.
   (v) [3 points] Name two resources that are specific to each thread.

(3) (i) [10 points] Give diagrams illustrating the communication patterns of: (single-node)
   broadcast, (single-node) scatter, (single-node) gather, all-gather.
   (ii) [10 points]
       − What does all-reduce do? Illustrate an implementation using tree structures for a global
         sum and for the distribution of the result.
       − Illustrate an implementation of all-reduce using a butterfly network.
       − What does all-gather do?

(4) (i) [10 points] Explain the static, dynamic, guided and runtime schedule types (OpenMP).
   (ii) [10 points] Illustrate schedule(static,1), schedule(static,2), schedule(static,4) for 16
   loop iterations and 4 threads. Illustrate schedule schedule(guided,1) for 39 loop iterations
   and two threads.

(5) (i) [12 points] Define speedup $S$, efficiency $E$ using $t$ threads. What is a cache line or block, write-miss, read-miss? What is thread safety?
   (ii) [8 points] Describe and explain possible cache related effects (in general) on perfor-
   mance, (a) for a sequential implementation (one thread), (b) in parallel (multiple threads).

(6) [20 points] Consider the loop

\[
\begin{align*}
a[0] &= 0; \\
\text{for} \ (i = 1; \ i < n; \ i++) \\
\quad a[i] &= a[i-1] + 1;
\end{align*}
\]

There is a loop-carried dependence, as the value of $a[i]$ cannot be computed without the
value of $a[i-1]$. Give a way to eliminate this dependence, and a (pseudo-code) paralleliza-
\text{tion for OpenMP (actual code is not needed).}