Reg. No. : | | | | | | | | | | | | |

Question Paper Code : 20379

B.E./B.Tech. DEGREE EXAMINATION, NOVEMBER/DECEMBER 2018.

Eighth Semester

Computer Science and Engineering

CS 6801 – MULTI-CORE ARCHITECTURES AND PROGRAMMING

(Regulations 2013)

(Common to PTCS 6801 – Multi-Core Architectures and Programming for B.E. (Part-Time) – Computer Science and Engineering – Regulations 2014)

Time : Three hours | Maximum : 100 marks

Answer ALL questions.

PART A — (10 × 2 = 20 marks)

1. State Amdahl's law.
2. What is symmetric shared memory?
3. List down the various synchronization primitives in parallel programming.
4. Compare deadlock and livelock in terms of resource reservation.
5. State the trapezoidal rule in OpenMP.
6. What are loop-carried dependencies?
7. Write a note on distributed memory machines.
8. How to compile an MPI program?
9. Name any two OpenMP environment variables.
10. List any two data scoping clauses in OpenMP.

PART B — (5 × 16 = 80 marks)

11. (a) (i) Outline the distributed shared-memory architecture with a diagram. (8)

(ii) Present an outline of parallel program design. (8)

Or

(b) Highlight the limitations of single core processors and outline how multicore architectures overcome these limitations. (16)

(a) What is deadlock? Explain the four conditions for deadlock and present an example for deadlock in a parallel computing environment. (16)

Or

(b) (i) Outline the critical section problem with an example. (6)

(ii) Explain how semaphores can be used to accomplish mutual exclusion of parallel-process synchronization with an example. (10)

(a) (i) Outline the OpenMp execution model. (8)

(ii) Discuss about OpenMp directives with relevant examples. (8)

Or

(b) (i) What is loop-carried dependence? Explain with an example. (8)

(ii) Outline with an example the use of the greatest common divisor test to determine whether dependences exist in a loop. (8)

(a) Explain the structure of an MPI program with an example. (16)

Or

(b) (i) Outline collective vs point-to-point communications in MPI with an example. (8)

(ii) What is a MPI derived data type? How to create a MPI derived data type? Give any two examples. (8)

5. (a) Outline the process of parallelizing depth-first search algorithm using OpenMP with an example. (16)

Or

(b) Write a note on thread paradigm and compare OpenMP and MPI programming models. (16)
