HIGH PERFORMANCE COMPUTING

Code: CS31227CS	Course Type: Elective	Semester: II
Evolution Scheme (TA-MSE-ESE): 20-30-100	Total Marks: 150	L-T-P (Credits): 3-0-0 (3)

Pre-Requisites: NIL

Course Objective: To understand parallel architectures, programming models, and performance metrics in high performance computing, and to design, implement, and evaluate parallel algorithms and HPC systems for solving real-world applications.

Course Outcomes: At the end of the course, students will be able to

CO-1	Analyze parallel architectures and performance metrics.
CO-2	Apply parallel programming using MPI, OpenMP, and GPU models.
CO-3	Design and implement parallel algorithms for real-world problems
CO-4	Evaluate advanced HPC systems and real-world applications.

Course Articulation Matrix:

PO	PEO-1	PEO-2	PEO-3	PEO-4	PEO-5
CO-1	3	3	1	1	2
CO-2	3	3	2	1	3
CO-3	3	3	2	1	3
CO-4	3	3	2	2	3

1 - Slightly; 2 - Moderately; 3 – Substantially

Syllabus:

Unit-1: Fundamentals of Parallel Computing
Introduction to High Performance Computing, Flynn's classification (SISD, SIMD, MISD, MIMD), Parallel architectures: shared memory, distributed memory, hybrid systems, Performance metrics: speedup, efficiency, scalability (Amdahl's & Gustafson's laws), Parallel programming basics and design principles.

Unit-2: Parallel Programming Models and Tools
Shared memory programming: OpenMP, Distributed memory programming: MPI, Hybrid programming (MPI + OpenMP), GPU programming basics (CUDA/OpenCL), Parallel algorithm design techniques (task/data parallelism).

.

Unit-3: Parallel Algorithm Design and Performance Optimization
Parallel algorithms: sorting, searching, matrix operations, Load balancing and task scheduling, Synchronization and communication overhead, Performance analysis and profiling, Scalability and optimization techniques.

Unit-4: Real-world HPC Systems and Applications

Cluster, grid, and cloud computing, Distributed file systems and storage, Fault tolerance and reliability, Energy-efficient computing (Green HPC), Applications: AI/ML, scientific computing, simulations.

Learning Resources:

Text Books:

1. Ananth Grama et al., Introduction to Parallel Computing, 2nd ed., Addison-Wesley, 2003.

2. Kai Hwang and Jack Dongarra, Distributed and Cloud Computing, Morgan Kaufmann, 2011.

Reference Books:

1. Georg Hager and Gerhard Wellein, Introduction to High Performance Computing for Scientists and Engineers, CRC Press, 2010.

2. John L. Hennessy and David A. Patterson, Computer Architecture: A Quantitative Approach, Morgan Kaufmann, 2019.