SARVAJANIK UNIVERSITY
INCLUSIVE | INTEGRATED | INNOVATIVE

SARVAJANIK UNIVERSITY
Sarvajanik College of Engineering and Technology
Master of Technology

Year: M. Tech. I (Semester – I)

Subject Name: High-Performance Computing **Subject Code:** MTCO24104
Type of course: Professional Elective - II
Prerequisite (if any): Knowledge of computer programming in Python and C,C++/Java

List of Courses where this course will be a prerequisite: --

Rationale: Large-scale simulations of engineering and basic science problems require efficient use of modern high-performance computing (HPC) infrastructure. This course primarily aims to introduce high-performance computing (HPC) concepts to engineering students. This course will discuss parallel computing tools like MPI, OpenMP, and CUDA. Students can analyze the performance metrics of high-performance computing with some benchmarks.

Teaching and Examination Scheme:

Teaching Scheme				Theory Marks			Practical Marks		Total
L	T	P	C	TEE	CA1	CA2	TEP	CA3	
3	0	2	4	60	25	15	30	20	150

CA1: Continuous Assessment (assignments/projects/open book tests/closed book tests CA2: Sincerity in attending classes/class tests/ timely submissions of assignments/self-learning attitude/solving advanced problems TEE: Term End Examination TEP: Term End Practical Exam (Performance and viva on practical skills learned in course) CA3: Regular submission of Lab work/Quality of work submitted/Active participation in lab sessions/viva on practical skills learned in course

Content:

Sr. No.	Content	Total Hrs
1	Parallel Processing Concepts : Levels of parallelism (instruction, transaction, task, thread, memory, function), Models (SIMD, MIMD, SIMT, SPMD, Dataflow Models, Demand-driven Computation, etc), Architectures: N-wide superscalar architectures, multi-core, multi-threaded	6
2	Parallel computers: Taxonomy of parallel computing paradigms, Shared-memory computers, Distributed-memory computers, Hierarchical (hybrid) systems, Networks. Basics of parallelization: Why parallelize, Parallelism, Parallel scalability	7

	Overview of parallel computers and high-performance computing (HPC), history of HPC, numerical and HPC libraries, and performance metrics.	
3	Parallel Programming with OpenMP: Introduction to OpenMP, Parallel constructs, Runtime Library routines, Work-sharing constructs, Scheduling clauses, Data environment clauses, atomic, master Nowait Clause, Barrier Construct, overview of MPI, MPI Constructs, OpenMP vs MPI. MPI: Functional parallelism, collectives in MPI, MPI data types, Communication, process management. OpenMPI: The OpenMP model, Parallel regions, Loop parallelism, Work sharing, Controlling thread data, and Synchronization.	12
4	Parallel Programming with CUDA: Introduction to GPU Computing, Processor Architecture, Interconnect, Communication, Memory Organization, and Programming Models in high-performance computing architectures (Examples: IBM CELL BE, Nvidia Tesla GPU, Intel Larrabee Microarchitecture and Intel Nehalem microarchitecture), Memory hierarchy and transaction-specific memory design, Thread Organization CUDA Programming Model, CUDA API, Simple Matrix, Multiplication in CUDA, CUDA Memory Model, Shared Memory Matrix Multiplication, Additional CUDA API Features	12
5	Recent Trend: Petascale Computing, Optics in Parallel Computing, Quantum Computers, Recent developments in Nanotechnology and its impact on HPCSupercomputing, Cluster Computing, Grid Computing, Petascale Systems	8

Reference Books:

Sr no	Title of book /article	Author(s)	Publisher and details like ISBN	Year of publication	Publication Edition
1	Introduction to High-Performance Scientific Computing	Victor Eijkhout, Edmond Chow, Robert van de Geijn		2016	2nd edition

2	Parallel Programming in MPI and OpenMP	Victor Eijkhout		2022, formatted March 9, 2023	2nd edition
2	CUDA Application Design and Development	Rob Farber	Morgan Kaufmann Publishers	2013	
3	Introduction to parallel computing	Zbigniew J. Czech	Cambridge University Press	2016	2nd edition
4	NVIDIA CUDA™ Programming Guide	(Link: https://uweb.engr.arizona.edu/~ece569a/Readings/NVIDIA_Resources/Programming%20Guide.pdf))			

Course Outcomes:

Sr. No.	CO statement	Marks % weightage
CO-1	To understand and explain different parallel architectures, interconnections, and various memory organizations in modern high-performance architectures.	30
CO-2	To comprehend the various High-Performance Computing Paradigms and Job Management Systems.	25
CO-3	To design and develop various applications with OpenMP, MPI, and CUDA.	25
CO-4	To know the overview and analyze the performance metrics of high-performance computing	15
CO-5	To analyze the benchmarks of high-performance computing.	5

List of Open learning websites:

1. https://nptel.ac.in/courses/106/102/106102114/
2. https://nptel.ac.in/courses/106/102/106102163/

List of Open Source Software:

1. OpenHPC,
2. CUDA

3. OpenMPI
4. MPI
5. TrinityX

FOR LAB SESSIONS:

List of Experiments:

Sr. No.	Practical
1	Simulation of RPC using 'c'
2	Basics of MPI (Message Passing Interface), Communication between MPI processes
3	Program to get familiarized with advanced communication between MPI processes
4	Study of MPI collective operations using Synchronization
5	Study of MPI collective operations using Data Movement
6	Study of MPI collective operations using Collective Computation
7	Distributed Memory Programming With MPI
8	Basics of OpenMP API (Open Multi-Processor API) ,familiarized with OpenMP Directives
9	Sharing of work among threads using Loop Construct in OpenMP
10	Sharing of work among threads in an OpenMP program using Single Construct and Sections Construct
11	Use of Environment Variables in OpenMP API
12	Basis of CUDA- kernel call,passing parameters.
13	Basis of parallel programming with CUDA
14	Multiplication in CUDA
15	Memory Handling with CUDA
16	Implementation of SIMD, MIMD, SIMT, SPMD model
17	Explorer multi-threading with appropriate examples.

Major Equipment Needed: