Department of Computer Science & Engineering

Academic Year	:	2025-26	Course Name	:	Parallel computing
Semester	:	7A	Course Code	:	BCS702
Scheme	:	22	L: T: P: C	:	3:0:2:4
Total Contact hours	:	40 hours Theory + 8-10 Lab slots	CIE Marks	:	50
Course Plan Author	:	Latha A	SEE Marks	:	50
Date	:	1/8/25	Total Marks	:	100

Course Prerequisites:

- Familiarity with programming languages like C, C++, or Python is typically required, along with an understanding of how computers work at a hardware level.

Learning Objectives:

- To demonstrate how to apply OpenMP pragma and directives to parallelize the suitable programs

Course Outcomes:

CO		*At the end of the course, student should be able to . . .*	Blooms' Level
CO1	:	Understand the need for parallel programming	L2
CO2	:	Demonstrate parallelism in MIMD system.	L2
CO3	:	Apply MPI library to parallelize the code to solve the given problem.	L3
CO4	:	Apply OpenMP pragma and directives to parallelize the code to solve the given problem.	L3
CO5	:	Design a CUDA program for the given problem.	L3

Blooms' Taxonomy:

L1	L2	L3	L4	L5	L6
Remembering	Understanding	Applying	Analyzing	Evaluating	Creating

Program Outcomes:

PO1	:	Engineering knowledge	PO7	:	Environment and sustainability
PO2	:	Problem analysis	PO8	:	Ethics
PO3	:	Design/development of solutions	PO9	:	Individual and team work

PO4	**:**	Conduct investigations of complex problems	**PO10**	**:**	Communication
PO5	**:**	Modern tool usage	**PO11**	**:**	Project management and finance
PO6	**:**	The engineer and society	**PO12**	**:**	Life-long learning

Program Specific Outcomes:

PSO1:	Model computational problems by applying mathematical concepts and design solutions using suitable data structures & algorithmic techniques.
PSO2:	Demonstrate basic knowledge of computer science in efficient design of problem solutions of varying complexity.
PSO3:	Create career path to become a successful computer science professional, entrepreneur and relish for higher studies

CO-PO-PSO Mapping:

CO	**Program Outcomes**														
	PO 1	**PO 2**	**PO 3**	**PO 4**	**PO 5**	**PO 6**	**PO 7**	**PO 8**	**PO 9**	**PO 10**	**PO 11**	**PO 1 2**	**PSO 1**	**PSO 2**	**PSO 3**
CO1	2					1						2	2		
CO2	2	2							2			2	2	2	
CO3	3	2	2	2	2				2	2	2	2	3	2	2
CO4	3	2	2	2	2				2	2	2	2	3	2	2
CO5	3	2	3	2	2				2	2	2	2	3	3	3
Target	2.6	2	2.3	2.0	2.0	1.0			2.0	2.0	2.0	2.0	2.8	2.3	2.3

Course Content (Syllabus)

Module-1	**08**
Introduction to parallel programming, Parallel hardware and parallel software – Classifications of parallel computers, SIMD systems, MIMD systems, Interconnection networks, Cache coherence, Shared-memory vs. distributed-memory, Coordinating the processes/threads, Shared-memory, Distributed-memory.	
Module-2	**08**

GPU programming, Programming hybrid systems, MIMD systems, GPUs, Performance – Speedup and efficiency in MIMD systems, Amdahl's law, Scalability in MIMD systems, Taking timings of MIMD programs, GPU performance.
Module-3 **08**
Distributed memory programming with MPI – MPI functions, The trapezoidal rule in MPI, Dealing with I/O, Collective communication, MPI-derived datatypes, Performance evaluation of MPI programs, A parallel sorting algorithm.
Module-4 **08**
Shared-memory programming with OpenMP – openmp pragmas and directives, The trapezoidal rule, Scope of variables, The reduction clause, loop carried dependency, scheduling, producers and consumers, Caches, cache coherence and false sharing in openmp, tasking, tasking, thread safety.
Module-5 **08**
GPU programming with CUDA - GPUs and GPGPU, GPU architectures, Heterogeneous computing, Threads, blocks, and grids Nvidia compute capabilities and device architectures, Vector addition, Returning results from CUDA kernels, CUDA trapezoidal rule I, CUDA trapezoidal rule II: improving performance, CUDA trapezoidal rule III: blocks with more than one warp.

Schedule of Instruction:

Class No	Topic	Date	RBT	CO	Mode
Module-1					
Introduction to parallel programming					
1.	Introduction to CO, Subject,Subject Prerquisites	11-08-25	L2	CO1	C&B
2.	Introduction and classifications of Parallel computers	12-08-25	L2	CO1	C&B
3.	SIMD systems	18-08-25	L2	CO2	C&B
4.	MIMD systems	19-08-25	L2	CO2	C&B
5.	Cache coherence	22-08-25	L2	CO1	C&B
6.	Shared-memory	25-08-25	L2	CO1	C&B
7.	distributed-memory	29-08-25	L2	CO1	C&B
8.	Coordinating the processes/threads	1-09-25	L2	CO1	C&B

9.	Shared-memory	2-09-25	L2	CO2	C&B
10.	Distributed-memory	8-09-25	L2	CO2	C&B
Module-2					
GPU programming, MIMD systems					
11.	Speedup	9-09-25	L2	CO2	C&B
12.	Efficiency in MIMD systems	10-09-25	L2	CO2	C&B
13.	Amdahl's law	12-09-25	L2	CO2	C&B
14.	Scalability in MIMD systems	15-09-25	L2	CO2	C&B
15.	Taking timings of MIMD programs	16-09-25	L2	CO2	C&B
16.	GPU performance	19-09-25	L2	CO1	Video
Module-3					
Distributed memory programming with MPI					
19.	MPI functions	22-09-25	L3	CO3	C&B
20.	The trapezoidal rule in MPI	23-09-25	L3	CO3	C&B
22.	Dealing with I/O	29-09-25	L3	CO3	PPT
23.	Collective communication	30-09-25	L3	CO3	PPT
24.	MPI-derived datatypes	03-10-25	L3	CO3	PPT
25.	Performance evaluation of MPI programs	06-10-25	L3	CO3	PPT
26.	A parallel sorting algorithm	07-10-25	L3	CO3	
28.	openmp pragmas and directives	10-10-25	L2	CO3	PPT
Module-4					
29.	The trapezoidal rule	13-10-25	L2	CO4	PPT
30.	Scope of variables	14-10-25	L3	CO4	C&B
31.	The reduction clause	17-10-25	L3	CO4	C&B
32.	The reduction clause	24-10-25	L3	CO4	C&B
33.	loop carried dependency	27-10-25	L3	CO4	C&B
34.	scheduling, producers and consumers	28-10-25	L2	CO4	PPT
35.	Caches, cache coherence and false sharing in openmp	31-10-25	L2	CO4	PPT
Module-5					
GPU programming with CUDA					
36.	GPUs and GPGPU	3-11-25	L2	CO5	PPT
37.	GPU architectures	4-11-25	L3	CO5	PPT
38.	Heterogeneous computing	7-11-25	L3	CO5	PPT

39.	Threads, blocks, and grids	10-11-25	L3	CO5	PPT
40.	Nvidia compute capabilities and device architectures	11-11-25	L3	CO5	PPT
41.	Vector addition	14-11-25	L2	CO5	PPT
42.	Returning results from CUDA kernels,CUDA trapezoidal rule	17-11-25	L2	CO5	PPT

PRACTICAL COMPONENT OF IPCC

Sl.NO	Experiments	Date	RBT	CO	Mode
1	Write a OpenMP program to sort an array on n elements using both sequential and parallel mergesort(using Section). Record the difference in execution time.	18-08-25	L3	CO4	EBL
2	Write an OpenMP program that divides the Iterations into chunks containing 2 iterations,respectively(OMP_SCHEDULE =static,2). Its input should be the number of iterations, and its output should be which iterations of a parallelized for loop are executed by which thread. For example, if there are two threads and four iterations, the output might be the following: a. Thread 0 : Iterations 0 –– 1 b. Thread 1 : Iterations 2 – 3	25-08-25	L3	CO4	EBL
3	Write a OpenMP program to calculate n Fibonacci numbers using tasks.	01-08-25	L3	CO4	EBL
4	Write a OpenMP program to find the prime numbers from 1 to n employing parallel for directive. Record both serial and parallel execution times.	08-09-25	L3	CO4	EBL
5	Write a MPI Program to demonstrate MPI_Send and MPI_Recv.	15-09-25	L3	CO3	EBL
6	Write a MPI program to demonstration of deadlock using point to point communication and avoidance of deadlock by altering the call sequence	22-09-25	L3	CO3	EBL
7	Write a MPI Program to demonstrate Broadcast operation.	29-09-25	L3	CO3	EBL
8	Write a MPI Program demonstration of MPI_Scatter and MPI_Gather	06-09-25	L3	CO3	EBL
9	Write a MPI Program to demonstration of MPI_Reduce and MPI_Allreduce (MPI_MAX, MPI_MIN, MPI_SUM, MPI_PROD)	13-10-25	L3	CO3	EBL

Textbooks:

T1	Peter S Pacheco, Matthew Malensek – An Introduction to Parallel Programming, second edition, Morgan Kauffman.
T2	Michael J Quinn – Parallel Programming in C with MPI and OpenMp, McGrawHill.

Reference books:

R1	Calvin Lin, Lawrence Snyder – Principles of Parallel Programming, Pearson
R2	Barbara Chapman – Using OpenMP: Portable Shared Memory Parallel Programming Scientific and Engineering Computation

Web links and Video Lectures (e-Resources):

1.	https://sites.google.com/view/sumaisecoursewebsite
2.	Introduction to parallel programming: https://nptel.ac.in/courses/106102163

Assessment Schedule:

S.N.	Assessment Type	Content	CO	Duration	Marks	Date
1.	CIE 1 (Theory)	M1,M2,M3	CO1,CO2,CO3	1:00	15	
2.	CIE 2 (Theory)	M3,M4,M5	CO3,CO4,CO5	1:00		
3.	CIE (Lab) - Record	M1,M2,M3 M4,M5	CO1,CO2,CO3 CO4,CO5	8-9 labs of each 2 hrs	15	
4.	CIE (Lab) - Test	M1,M2,M3 M4,M5	CO1,CO2,CO3 CO4,CO5	2:00	10	
5.	Quiz	M1,M2,M3 M4,M5	CO1-C05		10	
7.	Semester End Examination	M1,M2,M3 M4,M5	CO1,CO2,CO3 CO4,CO5	3:00	50	

RB** – Text Book/**R**eference **B**ook, ***L** – **L**ecture, **V**- Videos or any other mode, ***RBT** – **R**evised **B**looms' **T**axonomy, **L: T: P: C** – Theory/**L**ecture: **T**utorial: **P**ractical/Drawing: **C**redits, **SEE**: **S**emester **E**nd **E**xamination, **CIE**: **C**ontinuous **I**nternal **E**valuation, **Seminar**: Group of 6-8 students, **Module** 1,2,3,4 & 5,**The sum of total marks of three tests, two assignments, and seminar will be out of 100 marks and will be scaled down to 50 marks. (As per the scheme), **CIE + SEE = 50 + 50 = 100 marks

Faculty In charge **Course Coordinator** **HoD**