

14M1WCI431: Parallel Programming Techniques

Course Credit: 3

Semester: VIII/ M.Tech 2nd Sem

Introduction

Parallel programming developed as a means of improving performance and efficiency in a parallel program, the processing is broken up into parts, each of which can be executed concurrently. The instructions from each part run simultaneously on different CPUs. These CPUs can exist on a single machine, or they can be CPUs in a set of computers connected via a network. Parallel programming is an efficient form of information processing which emphasizes the exploitation of concurrent events in the computing process. The objective of the subject is to provide the knowledge and basic applications of parallel processing concepts, parallel environments and architectures, parallel algorithms and parallel programming. The format of the course will be lecture-discussions, assignments. Students are strongly encouraged to participate actively in class discussions.

Course Objectives (Post-conditions)

Knowledge objectives:

At the conclusion of the course, following learning objectives are expected to be achieved:

1. Describe different parallel processing architectures based on relationships between processing elements, instruction sequence, memory and interconnected network
2. Identify algorithms, which require parallelization as part of system design or performance enhancement
3. Classify shared and distributed memory parallel systems according to their properties and usage models
4. Design and develop parallel algorithms for shared and distributed memory models
5. Evaluate the performance of parallel algorithms designed based on shared and distributed memory models as well as against serial based algorithm designs

Application objectives:

The homework portions of the course are intended to help you apply your understanding,

1. Ability to acquire and apply fundamental principles of science and engineering
2. Ability to apply parallel programming constructs to make software execution parallel
3. Ability to think about different phenomenon in nature for their occurrence in parallel.

Expected Student Background (Preconditions)

Students are expected to have a solid grasp of the fundamentals of computer system, including a basic understanding of the operation of the computer, especially CPU. In addition, students are expected to know application development environment and programming concepts. Assembly

programming ability will be helpful, as we will be looking to understand architecture of contemporary computers.

Topics Outline:

S NO	Topics	Hrs
1	Introduction to Parallel Computers	3
2	Message-Passing Computing and Programming	5
3	Multithread Programming, OpenMP Programming	5
4	OpenMP Programming	3
5	Embarrassingly Parallel Computations	5
6	Partitioning and Divide-and-Conquer Strategies	5
7	Pipelined Computations and Synchronous Computations	5
8	Load Balancing and Termination Detection	5
9	Sorting Algorithms	3
10	Numeric Algorithms	3
	Total	42

References

1. Barry Wilkinson and Michael Allen, "Parallel Programming: Techniques and Applications Using Networked Workstations and Parallel Computers", Pearson Prentice Hall, 2004. (Textbook) .
2. Multi-Core Programming - Increasing Performance through Software Multi-Threading, Shameem Akhter and Jason Roberts, Intel Press 2006.
3. Parallel Programming in C with MPI and OpenMP, Michael J. Quinn, McGraw Hill 2003.

Evaluation Scheme:

S.No	Examination	Marks
1	T-1	15
2	T-2	25
3	T-3	35
4	*Internal Marks	25

*Internal Marks Breakdown:

Assignments 9 marks (3x3)

Quizzes 12 marks (3x4)

Regularity 4 Marks