CP5002 PARALLEL PROGRAMMING PARADIGMS

DETAILED SYLLABUS

OBJECTIVES

- To familiarize the issues in parallel computing.
- To describe distributed memory programming using MPI.
- To understand shared memory paradigm with P threads and with OpenMP.
- To learn the GPU based parallel programming using OpenCL.

UNIT I FOUNDATIONS OF PARALLEL PROGRAMMING

Motivation for parallel programming – Need-Concurrency in computing – Basics of processes, multitasking and threads – cache – cache mappings – caches and programs – virtual memory– Instruction level parallelism – hardware multi-threading – Parallel Hardware-SIMD – MIMD – Interconnection networks – cache coherence –Issues in shared memory model and distributed memory model –Parallel Software- Caveats- coordinating processes/ threads- hybrid model – shared memory model and distributed memory model - I/O – performance of parallel programs-– parallel program design.

UNIT II DISTRIBUTED MEMORY PROGRAMMING WITH MPI

Basic MPI programming – MPI_Init and MPI_Finalize – MPI communicators – SPMD programs– MPI_Send and MPI_Recv – message matching – MPI- I/O – parallel I/O – collective communication – Tree-structured communication -MPI_Reduce – MPI_Allreduce, broadcast, scatter, gather, allgather – MPI derived types – dynamic process management – performance evaluation of MPI programs- A Parallel Sorting Algorithm

UNIT III SHARED MEMORY PARADIGM WITH PTHREADS

Basics of threads, P threads– thread synchronization– critical sections– busy waiting – mutex– semaphores – barriers and condition variables – read write locks with examples - Caches, cache coherence and false sharing – Thread safety-P threads case study.

UNIT IV SHARED MEMORY PARADIGM: OPENMP

Basics OpenMP – Trapezoidal Rule-scope of variables – reduction clause – parallel for directive– loops in OpenMP– scheduling loops–Producer Consumer problem – cache issues– threads safety in OpenMP – Two- body solvers- Tree Search

UNIT V GRAPHICAL PROCESSING PARADIGMS: OPENCL AND INTRODUCTION TO CUDA

Introduction to OpenCL – Example-OpenCL Platforms- Devices-Contexts – OpenCL programming – Built-In Functions-Programs Object and Kernel Object – Memory Objects - Buffers and Images – Event model – Command-Queue - Event Object - case study. Introduction to CUDA programming.

REFERENCES

1. A. Munshi, B. Gaster, T. G. Mattson, J. Fung, and D. Ginsburg, ―OpenCL programming guide‖, Addison Wesley, 2011
2. M. J. Quinn, ―Parallel programming in C with MPI and OpenMP‖, Tata McGraw Hill, 2003.
3. Peter S. Pacheco, ―An introduction to parallel programming‖, Morgan Kaufmann, 2011.
4. Rob Farber, ―CUDA application design and development‖, Morgan Haufmann, 2011.
5. W. Gropp, E. Lusk, and A. Skjellum, ―Using MPI: Portable parallel programming with the message passing interface‖, Second Edition, MIT Press, 1999