Course Outline
ECE 420 Parallel and Distributed Programming

Course Information

Term: Winter 2021
Lecture dates and time: M W F 9:00AM - 9:50AM
Remote on Zoom

Instructor: Di Niu
Email: dniu@ualberta.ca
Office Hours: Monday 4-5pm.

Lab Instructor
Qikai Lu qikai@ualberta.ca
Lab TAs (QA/Lab Marking)
Mingjun Zhao zhao2@ualberta.ca
Chenglin Li ch11@ualberta.ca

Marker (Assignment and some Lab Marking)
Ali Safari Mamaghani safarima@ualberta.ca

Labs: done in teams of 3, submission due date on eclass.

Overview

This course provides an introduction to parallel programming, as well as the concepts and principles of parallel and distributed systems. The topics that will be studied include shared-memory parallel programming with Pthreads and OpenMP, distributed-memory programming on multicomputers with MPI. Parallel algorithm design and analysis. Big Data processing based on Apache Spark and Scala.

Objectives

After this course, you will be able to implement parallel applications on shared-memory systems and distributed applications on distributed-memory systems, understand the performance tradeoffs in the design of parallel and distributed systems, know how to process data in parallel using latest tools.

Pre-requisites

C, Big-O complexity notations, Operating Systems, Linux

NOTE: **Programming (in the lab sections) constitutes a substantial component of this course.** Familiarity with C and Unix (Linux command) definitely helps a lot. Knowledge of concurrency and basic operating system concepts will help. Knowledge of Big-O complexity notations will help. And it is highly recommended that you spend additional time learning this if you are not familiar with them.

Course Topics

We will cover important and fundamental topics in the design, implementation, and evaluation of parallel and distributed systems. Our focus will be on learning to write parallel programs using MPI, Pthreads, and OpenMP—three widely adopted application programming interfaces (APIs) for parallel programming, and understand the performance tradeoffs of parallel programs implemented on different frameworks. The design of parallel algorithms, and parallelization strategies will also be covered. Approximately 1/2–2/3 of the course will be on practical and hands-on aspects of parallel programming.

The rest of the course will be on principles, concepts and research issues. Some of the topics to be covered (in approximate order) include:

1. Introduction
Why use parallel and distributed systems?
Parallel hardware: SIMD, MIMD, shared-memory systems (multicore processors), distributed-memory systems (clusters)
Parallel software: threads and shared memory, processes and message passing, distributed shared memory
Performance: speedup calculation and Amdahl's Law

2. Shared-Memory Programming with Pthreads
Processes, Threads and Pthreads
Critical sections and locks
Synchronization and semaphores
Barriers and condition variables

3. Shared-Memory Programming with OpenMP
Compilation and execution of OpenMP programs
Parallelizing FOR loops with OpenMP

4. Distributed-Memory Programming with MPI
Compilation and execution
Communication patterns
Data and work partitioning

5. Parallel Algorithm Design
Data and work partitioning
Parallelization strategies
Granularity & Communication
Load balancing

6. Big Data Processing & Spark
Hadoop File System (HDFS) and Data Parallelism
MapReduce computation paradigm
Scala Functional Programming
Spark scripts for data analytics

Course Work and Evaluation

Lab 1: Getting familiar with the working environment and Pthreads	5%
Lab 2: Pthreads	10%
Lab 3: OpenMP	10%
Lab 4: MPI	10%
Lab 5: Spark & Scala	5%

For labs, students should work in teams of 3. Please inform the LI about your team members in the first two week.

Assignment 1 (Due date TBD)	3%
Assignment 2 (Due date TBD)	3%
Assignment 3 (Due date TBD)	3%
Midterm1 (in class, date TBD)	17%
Midterm2 (in class, date TBD)	17%
Midterm3 (date TBD)	17%

The assignments will contain questions that are representative of the questions to appear in the exams.

Course Materials

There is no required textbook for this course. However, the following are optional resources that can be used:

A. Grama, G. Karypis, V. Kumar and A. Gupta, Introduction to Parallel Computing, 2/e, Addison Wesley, 2003. (OPTIONAL).

Peter S. Pacheco, An Introduction to Parallel Programming, Morgan Kaufmann, 2011. (OPTIONAL)

Academic Integrity

"The University of Alberta is committed to the highest standards of academic integrity and honesty. Students are expected to be familiar with these standards regarding academic honesty and to uphold the policies of the University in this respect. Students are particularly urged to familiarize themselves with the provisions of the Code of Student Behaviour (online at http://www.governance.ualberta.ca/en/CodesofConductandResidenceCommunityStandards/CodeofStudentBehaviour.aspx) and avoid any behaviour which could potentially result in suspicions of cheating, plagiarism, misrepresentation of facts and/or participation in an offence. Academic dishonesty is a serious offence and can result in suspension or expulsion from the University."

All forms of dishonesty are unacceptable at the University of Alberta. Any offence will be reported to the Associate Dean who will determine the disciplinary action to be taken. Cheating, plagiarism and misrepresentation of facts are serious offences. Anyone who engages in these practices will receive at minimum a grade of zero for the exam or paper in question and no opportunity will be given to replace the grade or redistribute the weights. As well, in the Faculty of Engineering, the sanction for cheating will include a disciplinary failing grade (no exceptions) and senior students should expect a period of suspension or expulsion from the University of Alberta.

STUDENTS ELIGIBLE FOR ACCESSIBILITY-RELATED ACCOMMODATIONS (students registered with Specialized Support & Disability Services - SSDS): Eligible students have both rights and responsibilities with regard to accessibility-related accommodations. Consequently, scheduling exam accommodations in accordance with SSDS deadlines and procedures is essential. Please note that adherence to procedures and deadlines is required for U of A to provide accommodations. Contact SSDS (www.ssds.ualberta.ca) for further information.

STUDENT SUCCESS CENTRE: Students who require additional help in developing strategies for better time management, study skills or examination skills should contact the Student Success Centre (2-300 Students' Union Building).