LAU
School of Arts and Sciences

Department of Computer Science & Mathematics

Course Title :	Parallel Programming for Multicore and Cluster Systems **- CSC 447-**
Class Time and Location	TR 9:30 – 10:45 am, Frem Civic Center 0503
Instructor	Dr. Wissam HLAYHEL
Course Coordinator	Haidar Harmanani
Course Co-coordinator	
Credits Hours:	3
Semester:	Spring 2024

INSTRUCTOR

Email: wissam.hlayhel@lau.edu.lb
Office Hours: TR 8:30 – 9:20 (Block A, Part-times office near Labs).

CURRENT CATALOG DESCRIPTION

This course provides an introduction to parallel programming with a focus on multicore architectures and GPU programming. Topics include relevant architectural trends and aspects of multicores, writing multicore programs and extracting data parallelism at instruction level (SIMD), thread-level parallelism, task-based parallelism, efficient synchronization, program profiling, and performance tuning. Message-passing cluster-based parallel computing is also introduced. The course includes several programming assignments to provide students first-hand experience with programming, and experimentally analyzing and tuning parallel software.

PREREQUISITE/CO-REQUISITE

CSC310 - Algorithms and Data Structure : algorithms including, sorting, searching, complexity.
CSC326 - Operating Systems: concurrency, synchronization.
CSC320 - Computer Organization : ILP, CPU data dependency, cache memory optimization

TEXTBOOK

1. D. Kirk and W. Hwu, Programming Massively Parallel Processors: A Hands-on Approach, Third Edition, Morgan Kaufman, 2017 .

2. Gerassimos Barlas, Multicore and GPU Programming: An Integrated Approach, Morgan Kaufmann, 2015.

3. Barbara Chapman, Gabriele Jost, and Ruud van der Pas., Using OpenMP: Portable Shared Memory Parallel Programming, MIT Press, 2007.

COURSE TYPE

Required ☒ **E**lective ☐ **S**elective **E**lective ☐

COURSE LEARNING OUTCOMES

CLO.1 Students should understand the challenges of as well as the motivations for using parallel programming.

CLO.2 Students shall demonstrate an ability to analyze the efficiency of a given parallel algorithm.
CLO.3 Students shall demonstrate an ability to design, analyze, and implement programming applications on multicore and manycore systems.

TUDENT OUTCOMES ADDRESSED IN THIS COURSE

Outcome 1 contributes to SO.1

Outcome 2 contributes to SO.1

Outcome 4 contributes to SO.1, SO.2, and SO.6

COURSE GRADING AND PERFORMANCE CRITERIA (SUBJECT TO 5% VARIATION)

- Assignments 15%
- Midterm 25%
- Project 30%
- Final exam 30%

TOPICS COVERED IN THE COURSE

1) Introduction to parallel architectures and parallel programming;

2) Performance measures and analysis;

3) Task and thread programming

4) Synchronization tools

5) Shared-memory programming using OpenMP;

6) GPU Programming Using CUDA;

7) Introduction to message-passing Paradigm using MPI

ASSESSMENT PLAN FOR THE COURSE

End of semester self-assessment. Systematic Progression of Assignments. Reviewed every Spring semester by the coordinators for possible updates in the following Fall. Detailed review of all materials of the course once every three years by the CSC Curriculum Committee.

POLICY ON CHEATING AND PLAGIARISM

Students caught cheating on an exam receive a grade of zero on the exam in their first cheating attempt and receive a warning. Students caught cheating for the second time will receive a grade of "F" in the course and another warning. Plagiarism on assignments and project work is a serious offense. If plagiarism is detected, a student will be subject to penalty, similar to the cheating case, which ranges from receiving a zero on the assignment concerned to an "F" in the course in addition to a warning.

UNIVERSITY ATTENDANCE POLICY

Missing one third of classes implies that a student has to drop the course (It is the student's responsibility to drop the course).

UNIVERSITY ATTENDANCE POLICY

Students are advised to consult the University Official Policy regarding courses withdrawal at the following link: http://www.lau.edu.lb/academics/arp/u/withdrawal-from-university.php. In specific:

• WI (Early Withdrawal) Indicates withdrawal from the course, after the Late Registration Period and until the end of the 5th week of the Fall and Spring semesters, and until the 10th day of the Summer modules. It has no quality points. It does not count in the GPA, and no credits will be added to the student's record.

• WP (Withdrawal Pass) indicates withdrawal from the course, after the 5th week and until the end of the 10th week of the Fall and Spring semesters, and from the 11th day of classes until 18th day of the Summer modules. It has no quality points. It does not count in the GPA, and no credits will be added to the student's record.

• WF (Withdrawal Fail) indicates withdrawal from the course, after the 5th week and until the end of the 10th week of the Fall and Spring semesters, and from the 11th day of classes until 18th day of the Summer modules. It has no quality points. It does not count in the GPA, and no credits will be added to the student's record, but is counted as repeat. A Withdrawal Form must be submitted to the Registrar's Office.