University of Central Florida

Department of
Electrical Engineering & Computer Science

COP 4520: Parallel Algorithms and Programming
Spring 2016

Syllabus

Instructor: Dr. Damian Dechev Office tel.: 407-823-2549 email: dechev@eecs.ucf.edu

Lecture meetings: Mo We 4:30 PM - 5:45 PM @ HEC 0118

Office hours: Mo 3:00pm – 4:30pm @ HEC 0211

Teaching Assistant: Kishore Debnath email: kishore.debnath@knights.ucf.edu

TA office hours: TBA

Course Outline: This course is designed to provide a comprehensive presentation of the principles and tools available for programming parallel machines. The paradigm shift in the core computing architecture requires a fundamental change in how we program. The art of multiprocessor programming, currently mastered by few, is complex and requires understanding of new computational principles, algorithms, and programming tools. While parallel programming is of significant theoretical and industrial importance, there are few books addressing how to approach this challenging problem. Most engineers must learn intricate details of multi-processor synchronization through a frustrating trial and error process. The goal of the course is to enable the students to master the state-of-the-art modern programming techniques in concurrent programming as well as gain a solid knowledge base in the fundamental concepts of parallel computing.

Course Topics: Introduction to the core concepts of concurrent programming, design and implementation of concurrent algorithms and data structures, lock-free synchronization, programming tools and techniques for parallel computing, emerging parallel programming models, recent advances and future trends in concurrent programming, validation and verification of parallel processes, industrial applications.

Reference Guide:
The textbook for the course is: *The Art of Multiprocessor Programming,* Nir Shavir and Maurice Herlihy, Revised First Edition, Morgan Kaufmann 2012.

Style of Class Meetings:
The class meetings will not consist of traditional lectures, with the instructor doing most of the talking and the student doing most of the listening. Rather, meetings will consist of discussions on each topic and the instructor will help guide the discussion by asking questions.

Grading Policy:

- (40%) **Programming Assignments**
- (40%) **Homework Assignments and Solution Presentation**
- (10%) **Quizzes**
- (10%) **Data Structure Project Presentation**

Letter grades: 90-100: A, 80-89: B, 70-79: C, 50-69: D, Below 50: F.

Note: Any academic dishonesty (including, but not limited to, Cheating, copying and/or plagiarism) with respect to any exam or assignment in this class will result in a grade of **F**, following by the usual procedures for dealing with such behavior, as describe in the *UCF Golden Rule : a handbook for students.*

The Semester Plan: Tentative.

- Mutual exclusion
- Concurrent objects: consistency and semantics
- Shared memory data structures
- Synchronization primitives
- Spin-locks, read-write locks, contention
- Nonblocking shared data structures: linked-lists, queues, vectors, hash tables
- Hazardous concurrency bugs: ABA Problem, race-conditions, progress guarantees, linearizability
- Validation and verification of multi-processor algorithms
- Scheduling and work distribution
- Transactional memory
- Industrial applications for multi-core programming techniques: real-time systems, HPC applications, advanced simulations
- Programming language support for concurrency: new languages and language standards
- The application of static and dynamic program analysis for concurrency
- New programming models and frameworks for multi-core computing
- The role of advanced architecture simulators for the design of alternative programming models