

Engr 664: Theory of Concurrent Programming Syllabus, Fall 2016

H. Conrad Cunningham, D.Sc.
Department of Computer and Information Science
The University of Mississippi

22 August 2016

Course Overview

Catalog Description

Engr 664: Theory of Concurrent Programming. Topics in the theory of concurrent programming. Models of concurrency. Programming logics. Emphasis on the formal specification and verification of concurrent programs.

Prerequisite

This class is intended for students admitted in full standing in the graduate emphasis in Computer Science. Students are expected to have a background in finite state machines, operating systems, algorithms, data structures, programming languages, and discrete mathematics typical of an undergraduate computer science program. Use of the the Java programming language will likely be required in programming exercises.

Graduate students in other fields should discuss their backgrounds with the instructor for this course.

Position in Curricula

This is a graduate level course applicable to the MS and PhD emphases in computer science.

Course Goal

The primary goal of this course is to develop the students' knowledge of and experience with concurrent programming and use of formal methods.

Course Student Outcomes

Upon successful completion of the course, students will be able to:

1. describe the fundamental concepts for concurrent programming (e.g., processes, threads, interaction)
2. recognize the common problems that can arise in concurrent programs (e.g., interference, deadlock, starvation)
3. apply appropriate concurrent programming techniques (e.g., mutual exclusion, condition synchronization, message passing) to avoid the problems
4. describe the fundamental concepts of modeling concurrent systems using finite state machines (e.g., the Finite State Processes notation)
5. build appropriate finite state machine models for simple concurrent systems
6. formally express desirable and undesirable properties of concurrent systems (e.g., safety, liveness)
7. apply a model-checking tool (e.g., the Labelled Transition System Analysis tool) to evaluate whether the models have the desirable properties and do not have the undesirable ones
8. explain the concurrent programming features of the chosen programming language (e.g., Java)
9. develop finite state machine models as concurrent programs in the chosen programming language
10. analyze a concurrent programming problem to design an appropriate concurrent architecture
11. describe selected advanced techniques and architectures (e.g., dynamic systems, timed systems, filter pipelines).
12. relate the concepts and methods for concurrent programming to their previous programming knowledge and experiences
13. be more confident programming concurrent systems and using formal methods
14. appreciate the elegance and usefulness of formal methods for tackling difficult programming issues
15. study and learn new programming and analysis techniques

Fall 2016 Section Details

Instructor

Dr. H. Conrad Cunningham, Professor, Computer and Information Science
Office: 211 Weir Hall
Telephone: (662) 915-5358
Email: hcc@cs.olemiss.edu
Web: Professor Cunningham's Homepage (<http://www.cs.olemiss.edu/~hcc>)
Office hours: 10:15 - 11:30 a.m. MWF (or by arrangement)

Teaching Assistant

There is no teaching assistant assigned to this class.

Communication Policy

Students may contact the instructor by telephone or email or meet him during his office hours. He will attempt to respond to email and telephone messages within 24 hours during the work week. Emails or telephone calls arriving outside the 8:00 a.m. to 5:00 p.m., Monday-Friday workday may be deferred until the next workday.

Methods of Instruction

The instructional methods used in this course include readings from the textbook and other course materials, classroom lectures and discussions, assignments (programming projects, written exercises, and quizzes), and in-class exams.

Course Materials

The Engr 664 course website is at <http://www.cs.olemiss.edu/~hcc/engr/>.

Textbook (required): Jeff Magee and Jeff Kramer. *Concurrency: State Models and Java Programs, Second Edition*, Wiley, 2006. ISBN-13: 978-0-470-09356-6.

Hardware

The course does not require any special hardware. Students may use the Department's Adler lab facilities, servers, or their own desktop or laptop systems

to complete the programming assignments.

Software

To complete the course, students need the following software available on the computer at which they will do their work:

- Labelled Transition System Analysis (LTSA) toolset from the textbook's companion website
- Java 8 and other programming languages required by assignments.

Technical Support

Contact the instructor concerning any issues related to installation and use of the software or if access is needed to Department of Computer and Information Science computing facilities.

Course Topics

Basic concepts:

1. Concurrency concepts
2. Processes and threads
3. Concurrent execution
4. Shared objects and mutual exclusion
5. Monitors and condition synchronization
6. Deadlock
7. Safety and liveness properties
8. Model-based design

Advanced concepts (selected from the following):

9. Dynamic systems
10. Message passing
11. Concurrent architectures
12. Timed systems
13. Program verification
14. Logical properties
15. Other topics of interest to students and instructor

Course Expectations and Attendance Policy

The instructor expects each student to attend class and participate during each scheduled class period and to complete the required out-of-class assignments by

the stated deadlines.

Assignments

- All students are expected to study the relevant portions of the textbook and other course materials in conjunction with our class discussions (i.e., before coming to class). Explicit reading assignments will not always be given. If in doubt on what you need to read, please ask the instructor.
- Each student is expected to complete all required assignments.
- Several programming assignments will be assigned during the semester. We will likely use Java for these assignments. There may also be written assignments or short quizzes. (The instructor anticipates approximately five major assignments, but may add smaller assignments to replace or augment the major assignments as appropriate.)
- Unless otherwise stated in the assignment description, an assignment is to be carried out by each individual student without inappropriate collaboration with others. See the section on Academic Integrity.
- Most (but possibly not all) assignments will be submitted electronically using the Blackboard system.
- In preparing and submitting homework assignments make sure that:
 - your name, the course number or name, the assignment identifier, and individual exercises are clearly indicated in the content of the file or on the paper. (If it is a group assignment, give the group identifier and the names of all members.)
 - for any handwritten portions, you write legibly on only one side of the paper in a black or blue pen or dark pencil. Do NOT use red or green ink!
 - for any assignments submitted on paper, your pages are stapled together in the upper left corner when viewed from the front.
- All students are expected to complete their homework assignments by their due dates. If an assignment is submitted late, a penalty of 10 percent of that assignment's grade will be assessed for each class day it is late. A homework paper will not be accepted after graded papers have been returned, after a solution has been distributed, or after the final examination.
- If an assignment is marked "optional", it can replace an earlier assignment. However, all students are encouraged to work on these assignments; these optional activities can help the student master the concepts and skills and prepare for the proctored exams.

Examinations

- There will be three examinations – two mid-term exams and a final exam.
- The lowest grade of the three will be dropped and each of the other two grades will contribute 30 percent toward the course grade.
- The mid-term examinations will be given in late September and mid-November. The final examination will be given during the final examination period published by the Registrar's office (Monday, 6 December, 4:00 p.m.).
- Please do not ask to take the final examination earlier than the time set for the entire class.
- Each exam will cover all topics studied to that point. The final exam is comprehensive over all material covered during the semester.
- Each exam may consist of a combination of in-class and take-home components.
- If you cannot take an examination at the scheduled time because of an illness or other special circumstances, please notify the instructor in advance. Without advance notification, it may not be possible to give a make-up examination.

Optional Activities

The instructor may suggest various optional resources and activities. These activities are not graded, but they can help a student broaden and deepen his or her understanding of the programming language concepts.

Grading

- We will attempt to grade submitted assignments within approximately one week.
- 60% of the final grade comes from the grades on the examinations. The remaining 40% comes from grades on the assignments (programming project, written exercises, and quizzes).
- The ranges for the letter grades are 90-100% for A, 80-89% for B, 70-79% for C, 60-69% for D, and below 60% for F.
- +/- grades will be used near the boundaries at the discretion of the instructor. The actual assignment of +/- grades depends upon the overall distribution of grades for the class.

University of Mississippi Policies

Students with Disabilities

University policy provides for reasonable accommodations to be made for students with verified disabilities on an individualized and flexible basis as specified under Section 504 of the Rehabilitation Act of 1973 and the Americans with Disabilities Act of 1990 (ADA).

Students with disabilities who believe they may benefit from classroom or other accommodations should contact the *Office of Student Disability Services* for information at 234 Martindale, 662-915-7128 (Phone), 662-915-7907 (TTY Phone), or sds@olemiss.edu (Email).

Any student requiring accommodation under this policy should present the instructor with the required documentation early in the semester and make arrangements in advance with the instructor.

Copyrighted Materials

Materials used in connection with this course may be subject to copyright protection under Title 17 of the United States Code. Under certain Fair Use circumstances specified by law, copies may be made for private study, scholarship, or research.

Students should not share electronic copies of copyrighted materials with unauthorized users. Violations of copyright laws could subject individuals to federal and state civil penalties and criminal liability as well as disciplinary action under University policies.

Appropriate Use of Information Technology

The Information Technology (IT) Appropriate Use Policy sets forth the privileges of and restrictions on students, faculty, staff, and other users with respect to the computing and telecommunications systems offered by the University of Mississippi (UM). This policy is designed to protect the University community from illegal or damaging actions by individuals, either knowingly or unknowingly. Inappropriate use exposes the University to risks, including virus attacks, compromise of network systems and services, and legal issues. This policy directly addresses copyright issues related to illegal downloads and peer-to-peer file sharing.

For questions about the Appropriate Use Policy, send an email to aup@olemiss.edu.

Academic Integrity

The University of Mississippi is dedicated to supporting and sustaining a safe and scholarly community of learning dedicated to nurturing excellence inside and outside of the classroom. Each student has a duty to become familiar with University values and standards reflected in University policies, and each student has a duty to honor University values and standards reflected in University policies. These policies are outlined in the *M Book*. For a complete listing of policies, please visit the University Policy Directory.

As a student in Engr 664, you are expected to conduct yourself in a professional and ethical manner according to the policies, procedures, and expectations of the Department of Computer and Information Science, School of Engineering, Graduate School, University of Mississippi, and discipline of computer science.

The University's academic discipline procedure will be followed in this course. Violations of academic integrity may result in anything from failure on an assignment to expulsion from the course, depending on the severity of the violation.

Verification of Student Attendance

The University must abide by Federal guidelines to verify the participation of students. For all course types, including thesis, internships, labs, online courses, etc., the instructor must verify your participation based on some type of participation. In this course, the instructor will verify the physical attendance of each student and report it during the first two weeks and otherwise as required.

Student Privacy Policy

The University of Mississippi protects the privacy of all students, including online and distance learning students, through adherence to the Family Educational Rights and Privacy Act of 1974 (FERPA) through compliance with other institutional policies and procedures governing the management and security of protected information of faculty, staff, and students, and by outlining the expectations of privacy for the university community as regards to electronic information. See the *Student Information and Privacy Policy* for more information.