George Mason University

DEPARTMENT OF COMPUTER SCIENCE
CS 475 Concurrent & Distributed Systems
Spring 2025

3:00 - 4:15 pm, ENGR 1109

Professor Sanjeev Setia
703-993-4098
setia at gmu.edu

OFFICE HOURS
Wednesday 1:30-3:00 pm, Room 5305, Nguyen (Engineering) Building

PREREQUISITES
CS 367 (Computer Systems & Programming)

This is a programming intensive course. You need to be comfortable with programming in C and Java to be able to do the programming assignments. (You will also need to learn to program in Go). It is not a good idea to take this class if your course/work schedule is already quite full.

DESCRIPTION
The class focusses on issues that arise in the design and implementation of concurrent and distributed applications. Course work will involve several programming assignments.

TOPICS
The following topics will be covered:

- Processes & Threads
- Concurrent Programming (Mutual Exclusion, Synchronization Mechanisms, Locking Strategies)
- Exploiting thread-level parallelism
- Network Fundamentals
- RPC/RMI
- Distributed Application Architectures
- Coordination in Distributed Systems
- Consistency and Replication in Distributed Systems
- Fault Tolerance in Distributed Systems

READINGS
Unfortunately there is no single textbook that covers all the material that will be discussed in this class. The following books are strongly recommended (but not required):

- Randal Bryant & David O'Halloran. Computer Systems: A Programmer's Perspective,

Prentice Hall, 3rd edition (2015)
- Remzi Arpaci-Dusseau & Andrea Arpaci-Dusseau Operating Systems: Three Easy Pieces
- Martin Steen & Andrew Tannenbaum Distributed Systems 4th edition (2023)

GRADING

There will be four or five programming assignments.. The software required for these projects is available on the computers in the CEC Labs but you can also do the projects on your own computer. (The programming assignments involve programming in C, Go, and Java on a UNIX/ Linux platform). The assignments will be graded on zeus (CEC Labs) so please make sure to test them on zeus before submission.

All programming assignments need to be done individually by each student

- 40% of the course grade will be based on the programming assignments.
- 10% of the grade will be based on participation in class discussions
- 20% Midterm Exam
- 30% Final Exam

The final exam will be comprehensive in nature, i.e., it will cover all the topics discussed during the semester. All exams are closed book/closed notes.

Overall course grades will be generally based on the following scheme (but we may change the thresholds for various grades taking into account the performance of the class as a whole):
- A+ (>= 95%) A (>= 90%) A– (>= 85%)
- B+ (>= 80%) B (>= 75%) B– (>= 70%)
- C+ (>= 66%) C (>= 63%) C– (>= 60%)
- D (>= 50%)
- F (< 50%)

To pass the class, the weighted average of your midterm and final exams scores will need to exceed 20% of the overall course grade, irrespective of your performance on the programming assignments and in-class activities. Since the midterm and final exams are collectively worth 50% of the overall course grade, this implies that the average weighted score of your midterm and final exams should be above 40%.

LATENESS
The late submission policy is as follows: you have **two slip days** that you can use during the course of the semester, but at most one slip day per assignment. An assignment is considered late by one day irrespective of whether you miss the deadline by 1 minute or up to 24 hours.

GTA
TBA

CLASS NOTES
Class notes, slides, handouts, etc. will be available on the class Canvas page. Students should use Piazza for online discussions.

ABET COURSE OUTCOMES

- Demonstrate an ability to design and implement concurrent programs
- Demonstrate an understanding of the fundamental concepts in synchronizing concurrent processes and threads by using locks, monitors and channels
- Demonstrate an ability to design and implement distributed programs using current middleware technologies
- Demonstrate an understanding of the fundamental concepts underlying coordination, consistency and replication, and fault-tolerance in distributed system

HONOR CODE

GMU is an Honor Code university; please see the Office for Academic Integrity for a full description of the code and the honor committee process, and the Computer Science Department's Honor Code Policies regarding programming assignments. The principle of academic integrity is taken very seriously and violations are treated gravely. What does academic integrity mean in this course? Essentially this: when you are responsible for a task, you will perform that task. When you rely on someone else's work in an aspect of the performance of that task, you will give full credit in the proper, accepted form. When in doubt (of any kind) please ask for guidance and clarification.

DIVERSITY

This class seeks to create a learning environment that fosters respect for people across identities. We welcome and value individuals and their differences, including gender expression and identity, race, economic status, sex, sexuality, ethnicity, national origin, first language, religion, age and ability. We encourage all members of the learning environment to engage with the material personally, but to also be open to exploring and learning from experiences different than their own.

This class abides by the GMU anti-racism statement.

NOUNS AND PRONOUNS

If you wish, please share your name and gender pronouns with me and how best to address you in class and via email. I use he/him/his for myself. You may address me as Prof. Setia in email and verbally.

COMMON POLICIES AFFECTING ALL COURSES

Please review the Mason policies affecting all courses.

OTHER USEFUL CAMPUS RESOURCES

- Writing Center: https://writingcenter.gmu.edu; +1-703-993-1200; Robinson Hall A114
- University Libraries: Ask a Librarian
- Student Support and Advocacy Center (SSAC): https://ssac.gmu.edu/; +1-703-380-1434; Student Union Building (SUB) I, Suite 3200
- Counseling and Psychological Services (CAPS): https://caps.gmu.edu/; +1-703-993-2380

- University Policies: The [University Catalog](), is the central resource for university policies affecting student, faculty, and staff conduct in university academic affairs. Other policies are available at [https://universitypolicy.gmu.edu/](). All members of the university community are responsible for knowing and following established policies.
- [GMU Academic Calendar]()