Course Description
This course explores the principles of distributed computing systems, emphasizing fundamental issues underlying the design of such systems: communication, coordination, synchronization, and fault-tolerance. We will study key algorithms and theoretical results and explore how these foundations play out in modern systems and applications like cloud computing, edge computing, and peer-to-peer systems.

Below is a list of course topics and a preliminary schedule. The instructor may change the order and the contents depending on students’ backgrounds and other considerations.

Lecture 1: Introduction to Distributed Systems
Lectures 2-4: Clocks and the ordering of events in distributed systems
Lectures 5-6: The Replicated Log and Dictionary Problem
Lectures 7-10: Distributed Mutual Exclusion
Lectures 11-14: Distributed Agreement
Lectures 15-16: Leader Election
Lecture 17-18: Commit Protocols
Lectures 19-20: Byzantine Agreement
Lectures 21-23: Global Snapshots
Lectures 24-25: Broadcast Algorithms
Lectures 26-27: Replication and Amazon Dynamo
Lecture 28: Blockchain

Pre-requisites
- CSCI-2300: Introduction to Algorithms
- CSCI-4210: Operating Systems

Learning Outcomes
Upon successful completion of this CSCI 4510 and 6510, a student is able to:
- Understand and apply different models and abstractions for distributed systems
- Describe and analyze key algorithms for distributed systems
- Identify fundamental limitations and impossibility results for distributed systems
- Implement distributed algorithms in real-world distributed computing platforms
- Understand and identify applications of distributed algorithms in real-world systems

In addition, on successful completion of CSCI 6510, a student is able to:
- Read and analyze research papers on distributed systems and algorithms
- Develop and analyze novel distributed algorithms

Textbook
There is no required textbook for this course. All course material will be presented in the lectures, and conference and journal papers related to this material will be posted on the course web site.

This text may describe different algorithm variations than those presented in class. You are responsible for learning the algorithms versions and content presented in lecture.

**Grading**

Grades will be based on the following:

- Exams : 50%
- Programming Projects: 50%
  - Project 1: 5%
  - Project 2: 10%
  - Project 3: 20%
  - Project 4: 15%

Exam grades will be posted in Gradescope. Project grading will be done using Submitty.

The following chart will be used to assign course letter grades. Grades will be rounded to the nearest integer. The cutoff points may be lowered, and a different curve may be used for CSCI 4510 and CSCI 6510.

<table>
<thead>
<tr>
<th></th>
<th>B+: 87 – 89</th>
<th>C+: 77 – 79</th>
<th>D+: 67 – 69</th>
<th>F: 0 – 59</th>
</tr>
</thead>
<tbody>
<tr>
<td>A-:</td>
<td>90 – 92</td>
<td>B-: 80 – 82</td>
<td>C-: 70 – 72</td>
<td></td>
</tr>
</tbody>
</table>

**Exams**

There will be five online exams, each worth 8% of the course grade. Each exam will consist of 3 to 5 short-answer. You will have two hours for each exam, and you will be given a 12-hour window from which to select your two hours. The exam dates are posted on the course website. The exams will be managed through Gradescope. Your exam solutions must be typed, with the answer for each question starting on its own page. You may include images of digital or hand drawings in your solutions.

Requests for exam regrades must be made within 7 days of their return. Makeup exams are generally not allowed, but please contact the professor if you encounter extenuating circumstances.

After each exam, a subset of students will be selected at random for a post-exam follow up meeting. In this short WebEx meeting, you will explain the solution to one of your exam problems (chosen by the professor) to the professor or TA. The professor or TA will contact you to schedule these meetings. These meetings are worth 10% of the course grade.

**Programming Projects**

There will be four programming projects in which you will implement applications using algorithms studied in class. There may be different project requirements for students enrolled in CSCI-4510 and CSCI-6510.

**Students with Special Needs**

Federal law requires all colleges and universities to provide specified types of assistance to students with disabilities. If you have such special assistance, please obtain an authorizing memo from Disability Services for Students. Information about a student's special needs will be treated as confidential. Please submit a copy of your authorizing memo to the professor at least two weeks in advance of any affected assignment. Failure to do so may result in a lack of special accommodations.
Academic Integrity

For exams, no collaboration is allowed. During an exam window: do not discuss the exam with anyone; do not post questions online about the exam; do not read questions or answers that may be posted online about the exam. You may use your own notes, books, research papers, and online resources as references when completing the exam. You must write all solutions in your own words. If you use on a source outside of course materials, provide a citation. Do not copy text from another source, even if you cite the source.

For programming assignments, discussion is allowed, but you must write all of your own code.

Violation of the policies for projects or exams will be considered a breach of academic integrity, and the student will be subject to penalties outlined in The Rensselaer Handbook of Student Rights and Responsibilities, including "an academic (grade) penalty administered by the professor and/or disciplinary action through the Rensselaer judicial process described in this handbook."