ECE 751: Distributed Computing  
University of Waterloo  
Fall 2020

Meeting Times  
Lectures: Tuesday/Thursday at 5:30pm, held virtually using bongo

Note 1: The video and audio streams of live lectures will be recorded and shared with the class via LEARN. These recordings will include questions from the audience. If you do not wish to have your image or voice recorded, then kindly ask your questions using the live chat feature.

Note 2: In the event that the bongo virtual classroom becomes unavailable due to a technical problem, the contingency plan for lectures will be to switch to Prof. Golab’s WebEx meeting room, which is accessible via the following URL: https://uwaterloo.webex.com/meet/wgolab

Teaching Team  
Instructor: Prof. Wojciech Golab, wgolab@uwaterloo.ca

Note: Please include “ECE 751” in the subject line of any email you send to the teaching team.

Course Description  
Distribution has become essential to scaling computation on modern hardware. ECE 751 will introduce students to software techniques for distributed computing with a focus on two fundamental technical concerns: parallelism and fault-tolerance. These topics will be studied in the context of open-source software systems for communication, coordination, and data analytics.

Prerequisites and Antirequisites  
Prerequisites:
- Undergraduate-level training in operating systems, databases, and computer networks.
- Solid Java programming and debugging skills, including concurrency and I/O.
- Working knowledge of computer networking fundamentals, particularly TCP/IP and DNS.
- Familiarity with version control software (preferably Git, alternatively SVN or CVS).

Antirequisites:
- ECE 750 Topic 5 and Topic 10, ECE 454, CS 454

Note: The instructor reserves the right to override antirequisites on a case-by-case basis, and adjust the evaluation structure. For example, a student who completed ECE 454 may take ECE 751 with the instructor’s permission and complete a larger course project in place of some or all of the course assignments.
Learning Objectives
Upon successful completion of the course, students should be able to:

- Understand the architectures, protocols and algorithms underlying modern distributed systems.
- Understand fundamental techniques for making distributed systems scalable and dependable, and reason about design trade-offs.
- Use a remote procedure call (RPC) framework to implement a simple distributed service.
- Use a scalable data processing framework to solve fundamental analytics problems.
- Use a fault-tolerant coordination service for configuration management, synchronization, and failure detection in a distributed software system.
- Use a stream processing engine to perform real-time analytics.
- Discuss the relative merits of distributed versus centralized systems.

Learning Materials


This textbook is available online: https://www.distributed-systems.net/

Courseware: We will use LEARN (https://learn.uwaterloo.ca) to distribute lecture notes and homework materials. We will use Piazza (https://piazza.com/uwaterloo.ca/fall2020/ece751) for discussion.

Evaluation Structure

Coursework will include the following components:

<table>
<thead>
<tr>
<th>Component</th>
<th>Weight (%)</th>
<th>Notes</th>
</tr>
</thead>
<tbody>
<tr>
<td>Assignments</td>
<td>50%</td>
<td>(multiple, varying weights)</td>
</tr>
<tr>
<td>Quizzes</td>
<td>10%</td>
<td>(online)</td>
</tr>
<tr>
<td>Research Project</td>
<td>40%</td>
<td>(includes one or more in-class presentations)</td>
</tr>
</tbody>
</table>

Note 1: The instructor reserves the right to use alternative evaluation structures in special circumstances. For example, adjustments may be made to address antirequisite courses, as noted earlier, and accommodations may be offered to students affected directly or indirectly by COVID-19.

Note 2: The quizzes will be open book in the sense that you are permitted to refer to the lecture materials provided by the instructor and the course textbook. You may also refer to online materials available on the Internet, such as API docs and tutorials. You may not collaborate with another individual (e.g., via phone, chat, or a forum like Stack Overflow) while completing the quiz, and you may not post the quiz questions or answers publicly.

Note 3: Unless otherwise approved by the course instructor, each student will complete the two smaller programming assignments (Assignment 0 worth 10% and Assignment 4 worth 4%) for 14% of the course grade, as well as two of the three large programming assignments (Assignment 1, Assignment 2, and Assignment 3, each worth 18%) for 36% of the course grade. If a student who is following the default evaluation structure completes all three of the large programming assignments, the highest two grades will be recorded.
**Group Work**

Unless stated explicitly by the course instructor, all coursework will be completed individually. In cases where group collaboration is permitted, coursework will be completed in small groups (1-3 students). Each group member individually is responsible for learning all the material corresponding to a group deliverable, and may be required to answer technical questions posed by members of the teaching team. Copying material from other groups, from other courses, or from online sources is forbidden except for materials authorized explicitly by the course instructor. Unauthorized collaboration and copying of material constitute academic offences under Policy 71.

**Lateness and Absence**

Coursework deliverables will be penalized −20% if submitted up to 24 hours late, −40% if submitted more than 24 hours and up to 48 hours late, and will not be accepted at all after 48 hours. Lateness penalties will not be prorated (e.g., a deliverable submitted 1 hour late will be penalized by the same amount as a deliverable submitted 23 hours late). The instructor reserves the right to waive the lateness penalty for submissions made within a short grace period (usually about 5 minutes) after the scheduled submission deadline, or in the event of a major disruption such as a campus-wide power outage.

**Submission and Pickup of Assignments**

Assignment deliverables will be submitted electronically using a drop box in LEARN. Feedback from graders may be provided on paper, electronically by e-mail, or electronically using LEARN.

**Grading and Regrading**

All coursework items will be graded on the basis of completeness, correctness, and clarity. Students are encouraged to ask questions related to grading during lecture.

Where appropriate, students are expected to test their own solutions to problems using appropriate input data. In the event that a student's solution differs substantially from the solution(s) accepted by the teaching team, the student may be asked to provide sample inputs and outputs to evaluate the correctness of their solution.

Students who feel that they have been graded unfairly may request that a coursework item be re-graded, in which case the entire item (i.e., assignment or exam) will be re-graded. As a result, the grade may increase, decrease, or remain unchanged. The teaching team may refuse to re-grade coursework if there is substantial concern that the solution was modified or otherwise tampered with after submission. A penalty will be applied for all re-grades, except in cases of a buggy grading script.
General Policies and Information

Academic integrity: In order to maintain a culture of academic integrity, members of the University of Waterloo community are expected to promote honesty, trust, fairness, respect and responsibility. [Check the Office of Academic Integrity for more information.]

Grievance: A student who believes that a decision affecting some aspect of his/her university life has been unfair or unreasonable may have grounds for initiating a grievance. Read Policy 70, Student Petitions and Grievances, Section 4. When in doubt, please be certain to contact the department’s administrative assistant who will provide further assistance.

Discipline: A student is expected to know what constitutes academic integrity to avoid committing an academic offence, and to take responsibility for his/her actions. [Check the Office of Academic Integrity for more information.] A student who is unsure whether an action constitutes an offence, or who needs help in learning how to avoid offences (e.g., plagiarism, cheating) or about “rules” for group work/collaboration should seek guidance from the course instructor, academic advisor, or the undergraduate associate dean. For information on categories of offences and types of penalties, students should refer to Policy 71, Student Discipline. For typical penalties, check Guidelines for the Assessment of Penalties.

Appeals: A decision made or penalty imposed under Policy 70, Student Petitions and Grievances (other than a petition) or Policy 71, Student Discipline may be appealed if there is a ground. A student who believes he/she has a ground for an appeal should refer to Policy 72, Student Appeals.

Note for students with disabilities: AccessAbility Services, located in Needles Hall, Room 1132, collaborates with all academic departments to arrange appropriate accommodations for students with disabilities without compromising the academic integrity of the curriculum. If you require academic accommodations to lessen the impact of your disability, please register with AccessAbility Services at the beginning of each academic term.

Turnitin.com and alternatives: Plagiarism detection software (e.g., Turnitin or MOSS) may be used to screen deliverables in this course. In the first week of the term, details will be provided on request about the arrangements for the use of plagiarism checkers and alternatives in this course.