EECS3342: System Specification and Refinement

Section Z – Winter 2022

Last Updated: January 10, 2022

Course Syllabus

1 Course Policies 2

2 Academic Integrity 3

3 Instructors 4

4 eClass Site 4

5 Study Materials 4

6 Available Help Resources 4

7 Prerequisites 4

8 Course Description 5

9 Course Learning Outcomes (CLOs) 5

10 Grading Scheme 6

11 Lab Exercises: No Submissions Required 6

12 Expected Weekly Workload 6

13 Mapping Raw Marks to Letter Grades 7

14 Semester Calendar 8

15 Scheduled Q&A and Lab Time 9

16 (Tentative) Weekly Lecture Topics 10

1 Course Policies

To ensure a smooth, fair, and effective online delivery of this course:

1. **Plagiarism**: When submitting your **programming test**, you claim that it is **solely** your work. It is considered as **an violation of academic integrity** if you copy or share **any** parts of your work (e.g., code, notes) during **any** stage of your development. The instructor and TAs **will** examine **all** submitted code, and suspicious submissions will be reported *immediately* to Lassonde as *a breach of academic integrity*. **We do not tolerate academic dishonesty**, so please be fully responsible for your learning.

2. **Online Submission/Assessment**: **Stringent deadlines** are imposed on all **scheduled written & programming tests**. An **exam** is scheduled online (via eClass) with **stringent timing requirements** (start time, duration, and end time to be announced by the registrar office).

   All announced deadlines are in the Eastern Time Zone (Toronto time). Students on a different time zone must figure out the corresponding local time.

   Students are responsible for **taking proactive steps and/or seeking assistance** well in advance to ensure that their technical setup (e.g., stable internet connection, a computer which does not freeze sporadically) allows them to complete and submit each assessment item (written test, programming test, exam) in time.

   Rationales for this policy are to: **urge** students with technical issues to take steps or seek assistance to fix/improve them (otherwise, how can they benefit from the online setting in the first place?); and **discourage** students trying to take an unfair advantage (e.g., a student ignorant of the submission deadline or starting late may claim technical failure to have an extension, a student who has already seen the exam questions may claim network/computer failure in order to gain extra time or a deferred exam).

   When it comes to assessments, your instructor's priorities are fairness and academic integrity.

3. **No Team Work**: All **written & programming tests** are to be developed and completed **individually** (i.e., **team work is forbidden**). This is meant for avoiding students having difficulties finding a suitable teammate and disputes between teammates (e.g., non-responsiveness, overdue progress, last-minute notice of withdrawal): the online nature of this course would only exacerbate these problems.

4. **Late Enrolment**: Students who are not yet officially registered should assume an eventual successful enrolment into the course and are responsible for: **1)** contacting the section instructor **within Week 1** for course information (e.g., lecture materials, lab access and deadlines); and **2)** studying lecture videos, attending Q&A sessions, and submitting tests in time.

   No deadline extensions or deferred tests will be accommodated.

2 Academic Integrity

Written Tests & Programming Tests

– All written and programming tests, as well as the final exam, are to be completed ***individually***: no group work is allowed.

TAs will perform thorough checks on **all** programming test submissions: convincingly suspicious submissions will be reported to the Lassonde Student Service for a ***formal investigation*** immediately.

– It is considered ***a breach of academic honesty*** if:

- You collaborate with someone on completing a written or programming test during any stage of your development.
- After you have attempted the written or programming test and before that test is closed, share your test questions with someone.

Reporting Cases

Enforcing the policy of academic honesty not only maintains the ***standard*** of the course, but also ensures ***fairness*** among all students in the class. If you have sufficient reasons to believe that cases of violation are present, let the instructor know and confidentiality will be maintained.

3 Instructors

- Chen-Wei (Jackie) Wang [Section B & Section E]
  - Contact: jackie@eecs.yorku.ca (http://www.eecs.yorku.ca/~jackie/)
  - Virtual Office: https://yorku.zoom.us/my/jackie.loves.oxford
  - Office Hours: 14:00 – 15:00 (EST), Mon, Tue, Thu; or by Appointments.

4 eClass Site

- There is a single eClass site for Section Z:

  https://eclass.yorku.ca/course/view.php?id=49205

5 Study Materials

- The main study materials will be made available on the lectures page:

  https://www.eecs.yorku.ca/~jackie/teaching/lectures/index.html#EECS3342_W22

- Modeling in Event-B (2010), Cambridge University Press (http://www.event-b.org/)

  Jean-Raymond Abrial

  By agreement with the author, a draft of the book is available for the private use of EECS students at York, and through a valid PPY username/password on the course eClass site. This book draft is copyrighted by the author and may not be distributed in any manner. By logging in and downloading this book draft, you agree to abide by all the copyright conditions. Note that there are errors and typos in the notes. The final text may be purchased by students.

6 Available Help Resources

- Your instructor's office hours
- Scheduled lab sessions starting on Week 2 (you can attend any of them to ask TA questions)
- Weekly Q&A sessions (held by the instrutor)

7 Prerequisites

- **General Prerequisites**: A cumulative grade point average (GPA) of 4.50 or better over all previously completed Major EECS courses. The GPA computation excludes all EECS courses that have a second digit 5, or are Co-Op/PEP courses.
- LE/EECS 2030 3.00 or LE/EECS 1030 3.00
- LE/EECS 2011 3.00
- SC/MATH 1090 3.00

8 Course Description

This course provides students with an understanding of how to use mathematics (set theory and predicate logic) to specify and design correct computer systems whether the systems are sequential, concurrent or embedded. The course stresses both the underlying theory as well as the ability to use industrial strength tools that can be applied in practice.

User requirements are formalized via an abstract mathematical model that is amenable to formal reasoning long before any programming activity is undertaken (e.g. as done in Event-B, Z and VDM). Successive models are like blueprints in traditional engineering disciplines and their mathematical nature allows us to reason about and predict their safety properties.

9 Course Learning Outcomes (CLOs)

Upon completion of the course, students are expected to be able to:

CLO1 Document requirements organizing them into appropriate categories such as environmental constraints versus functional properties (safety and progress).

CLO2 Construct high level, abstract mathematical models of a system (consisting of both the system and its environment) amenable to formal reasoning.

CLO3 Apply set theory and predicate logic to express functional and safety properties from the requirements as events, guards, system variants and invariants of a state-event model.

CLO4 Use models to reason about and predict their safety and progress properties.

CLO5 Plan and construct a sequence of refinements from abstract high-level specifications to implemented code.

CLO6 Prove that a concrete system refines an abstract model.

CLO7 Apply the method to a variety of systems such as sequential, concurrent and embedded systems.

CLO8 Use practical tools for constructing and reasoning about the models.

CLO9 Use Hoare Logic and Dijkstra weakest precondition calculus to derive correct designs.

10 Grading Scheme

Programming Test	10%
4 Written Tests (10% each)	40%
Exam (Cumulative)	50%

11 Lab Exercises: No Submissions Required

- Lab exercises are released to help you prepare for the written tests, which also cover the relevant lecture materials.
- No submissions are needed for labs: they are first released for you to attempt, and solutions will be made available to you.

12 Expected Weekly Workload

- Lassonde's recommendation is 3 – 4.5 hours per credit: *9 – 13.5 hours* for a 3.00 course.
- "In-Class" Hours:
  - Lecture Videos [≈ 3 hours]

    Optional: Schedule Labs, Q&A sessions, Office Hours
- "Out-of-Class" Hours:
  - Completing Lab Assignments, Studying for Lectures/Tests [6 to 10.5 hours]

13 Mapping Raw Marks to Letter Grades

According to the Common Grading Scheme for Undergraduate Faculties approved by Senate:

Letter Grade	Grade Point	Interpretation
A+	9	Exceptional
A	8	Excellent
B+	7	Very Good
B	6	Good
C+	5	Competent
C	4	Fairly Competent
D+	3	Passing
D	2	Marginally Passing
E	1	Marginally Failing
F	0	Failing

- For each grading unit, you will receive a **raw mark score** (not necessarily out of 100).
- The **weighted sum** of all grading units will be mapped to its letter grade.

  e.g., Say there are only two grading units: Exam (60%) and Midterm (40%).

  A student receiving 150 marks (out of 200) for Exam and 2 marks (out of 3) for Midterm has:

  Weighted sum: $\frac{150}{200} \times 60 + \frac{2}{3} \times 40 \approx 71.7$

  Letter grade: B

14 Semester Calendar

– Figure 1 summarizes the schedule of required work items:

- Pre-recorded lectures are released on Wednesdays.
- Optional Q&A sessions (for questions related lectures) are held during the scheduled class times: Thursdays (11:30 to 13:00, EST).
- Lab exercises are released on Tuesdays.
- Each written test lasts for **30 minutes** and takes place between 11:30 and 12:00 on the corresponding Tuesdays.
- Each programming test lasts for **90 minutes** and takes place between 11:30 and 13:00 on the corresponding Tuesdays.

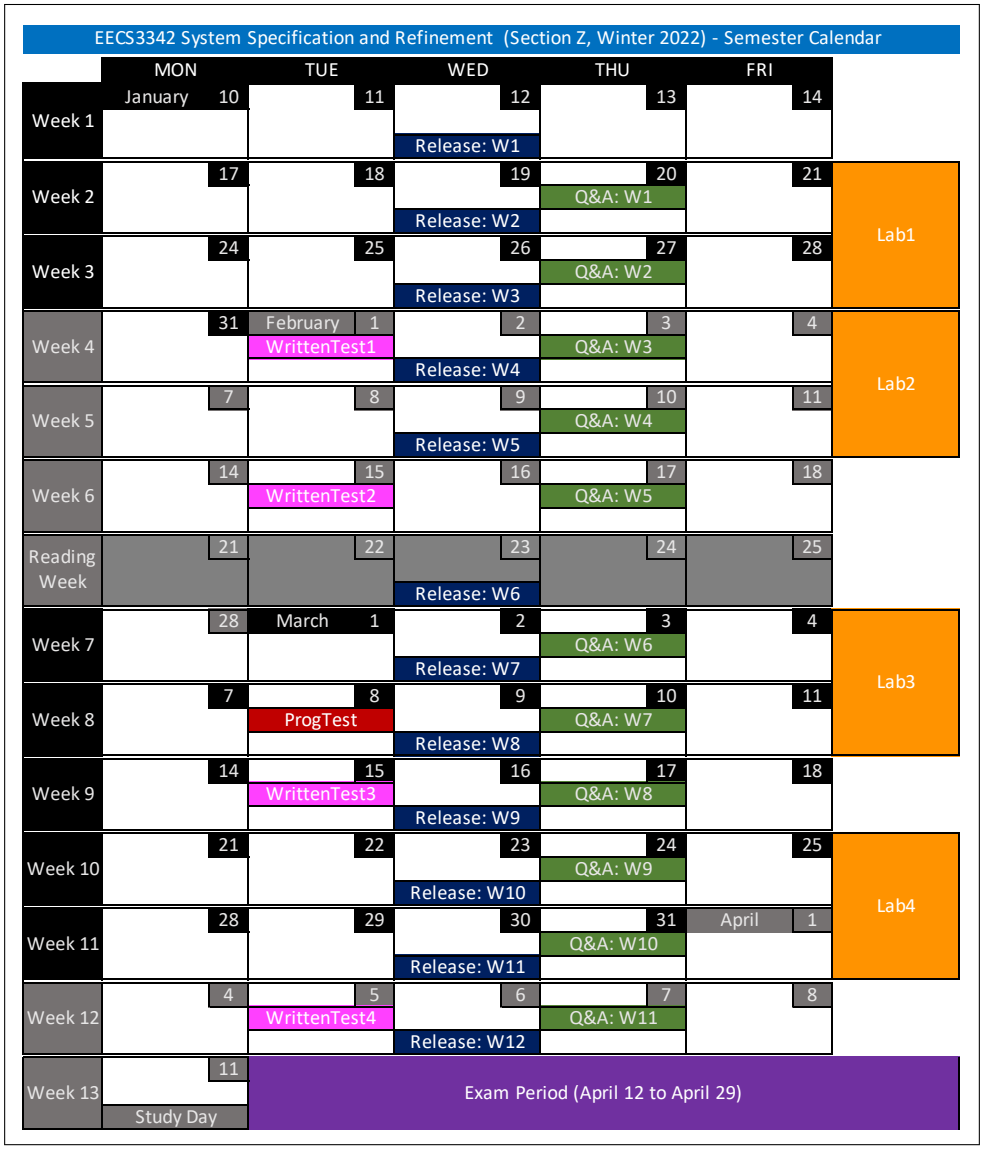

Figure 1: EECS3342-Z W22 Semester Calendar – Expected Work Items

15 Scheduled Q&A and Lab Time

– In the time table below, each cell denotes a 30-minutes interval. For examples:

- Cell 11:30 denotes the interval starting at 11:30 and ending at 12:00.
- The fact that the Q&A session on Thursday occupies 3 cells indicates that it lasts for 1.5 hours (starting at 11:30 and ending at 13:00).

	Monday	Tuesday	Wednesday	Thursday	Friday
8:30			EECS3342-Z Lab 3		
9:00					
9:30					
10:00					
10:30					
11:00					
11:30		Test when Scheduled		Q&A	
12:00					
12:30					
13:00		EECS3342-Z Lab 1		EECS3342-Z Lab 5	
13:30					
14:00	Office Hours			Office Hours	
14:30					

– The lecture time slot on Tuesdays is used to host written tests and programming tests. See the test schedule on Figure 1 (p8).

– The scheduled lecture time slot

- **11:30 – 13:00 on Thursdays**

is used to hold (optional, Zoom) Q&A sessions to answer your questions related to the lecture materials.

– All scheduled lab sessions are *optional*: you are welcome to attend **any** of them to ask questions related to lab assignments.

Remark. For both kinds of Q&A sessions, it is completely your decision on attending one, more, or none of them. However, I would **not** advise that you skip all of them, unless you are absolutely confident with the course materials.

16 (Tentative) Weekly Lecture Topics

Lecture videos are being actively recorded, so the order of topics below are subject to changes.

Week	Topics
1	• Introduction • Review on Math
2 – 5	• Reactive Systems: Bridge Controller
6	• Distributed Systems: FTP Protocol
Reading Week	
7 – 8	• Distributed Systems: Bounded Re-Transmission Protocol
9 – 10	• Development of Concurrent Programs
11 – 12	• Development of Sequential Programs (Refinement and Hoare Logic)