

# EECS4302: COMPILERS AND INTERPRETERS

Section A – Summer 2025

LAST UPDATED: MAY 6, 2025

SUBJECT TO CHANGES UNTIL: MAY 19, 2025

---

## COURSE SYLLABUS

|    |  |    |
|----|--|----|
| 1  | <u>COURSE POLICIES</u>                                   | 2  |
| 2  | <u>INSTRUCTOR</u>  | 3  |
| 3  | <u>VENUES</u>  | 3  |
| 4  | <u>ECLASS SITE</u>                                       | 3  |
| 5  | <u>STUDY MATERIALS</u>                                   | 3  |
| 6  | <u>PREREQUISITES</u>                                     | 4  |
| 7  | <u>COURSE DESCRIPTION</u>                                | 4  |
| 8  | <u>COURSE LEARNING OUTCOMES (CLOs)</u>                   | 5  |
| 9  | <u>GRADING SCHEME</u>                                    | 5  |
| 10 | <u>FINAL EXAM: CUMULATIVE &amp; SUBSTANTIAL</u>          | 5  |
| 11 | <u>MAPPING RAW MARKS TO LETTER GRADES</u>                | 5  |
| 12 | <u>EXPECTED WEEKLY WORKLOAD</u>                          | 6  |
| 13 | <u>ATTENDANCE OF CLASSES: ENCOURAGED &amp; REWARDING</u> | 6  |
| 14 | <u>SEMESTER CALENDAR</u>                                 | 9  |
| 15 | <u>COVERAGE OF QUIZZES</u>                               | 10 |
| 16 | <u>WEEKLY SCHEDULE</u>                                   | 10 |
| 17 | <u>(TENTATIVE) TOPICS TO BE COVERED</u>                  | 10 |

# 1 COURSE POLICIES

To ensure a smooth, fair, and effective in-person delivery of this course:

1. **Team Work Encouraged for Labs/Assignments**: You will receive **full** marks as long as submission attempts are made by the corresponding submission deadlines.

Your submitted labs/assignments, though awarded **full** marks automatically, will still be graded and given detailed feedback (i.e., compilation and testing results). **The grading results and feedback are intended to exemplify how the actual programming tests will be graded.** Therefore, it would be your best interest in submitting work representing your **true** and **best** attempt.

The rationales of this policy are that: 1) you can **rest assured that you will not lose any marks from labs (as long as you submit them by the deadlines)**; and 2) **you can just focus on the learning by seeking help from colleagues, TAs, and Jackie without worrying about violating the academic honesty policy.**

Please do not abuse this policy: **you are still 100% responsible for acquiring the intended understandings and skills from these labs.** Be advised that **later scheduled (written and programming) tests will be based on these labs**, so if you chose **not** to learn the materials responsibly (e.g., relying much on your colleagues, submitting incomplete work and only intending to look at solutions when they are made available), you risk **poor performance** in subsequent tests and the exam.

2. **No Team Work Allowed for Scheduled Tests**: All **written & programming tests** are to be completed **individually** (i.e., **team work is forbidden**).

When a scheduled test takes place between different lab sessions, until all test sessions conclude, it is considered a **a violation of academic integrity** if you communicate in any way, shape, or form with others about the test(s) already given.

3. **Plagiarism**: When submitting each of your **written tests** and **programming tests**, you claim that it is **solely** your work. It is considered as **a 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 may examine all submissions, and suspicious ones will be reported *immediately* to Lassonde as *a breach of academic integrity*. **We do not tolerate academic dishonesty.**

4. **MEETING LAB/TEST DEADLINES**: **Stringent deadlines** are imposed on all scheduled **written tests** (to be completed and submitted via eClass), as well as scheduled **programming tests** and **labs** (to be submitted via the *web submit* to the EECS server). An in-person **exam** will be scheduled by the registrar office to take place during the **exam period**. It is your responsibility for meeting all deadlines.

5. **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 assignments access and deadlines); and 2) attending lectures, submitting lab assignments, and taking scheduled tests in time.

**No lab deadline extensions or deferred tests will be accommodated.**

## 2 INSTRUCTOR

– Chen-Wei (JACKIE) Wang

- Contact: [jackie@eecs.yorku.ca](mailto:jackie@eecs.yorku.ca) (<https://www.eecs.yorku.ca/~jackie/>)

Jackie believes that **in-person** communication is the *most effective* for attending to your questions/concerns related course materials and grading. When you receive slow or no responses to your email inquiries, it is most likely an indication that Jackie is happy to help you during his **in-person** office hours and/or appointments.

- Office Hours:
  - \* 12:30 – 13:30, Tuesdays & Thursdays (**in-person**)
  - \* 13:00 – 14:00, Wednesdays
  - \* by appointments (Zoom or In-Person)

Campus Office: Lassonde Building, Room 2043 [ 19, D5 in the Keele campus ]

Virtual Office: <https://yorku.zoom.us/my/jackie.loves.oxford>

## 3 VENUES

17:30 – 19:00, Tuesdays & Thursdays

LSB 105 (Life Science Building) [ C4/90 on the Keele Campus ]

## 4 ECLASS SITE

<https://eclass.yorku.ca/course/view.php?id=132049>

## 5 STUDY MATERIALS

– There will be no textbooks for this course. Study your instructor's lecture materials:

- The lectures page:  
[https://www.eecs.yorku.ca/~jackie/teaching/lectures/index.html#EECS4302\\_S25](https://www.eecs.yorku.ca/~jackie/teaching/lectures/index.html#EECS4302_S25)

– If you wish to review OOP in Java, consider the study materials for:

- EECS2030 (Introductory OOP):  
[https://www.eecs.yorku.ca/~jackie/teaching/lectures/index.html#EECS1022\\_W21](https://www.eecs.yorku.ca/~jackie/teaching/lectures/index.html#EECS1022_W21)
- EECS2030 (Advanced OOP):  
[https://www.eecs.yorku.ca/~jackie/teaching/lectures/index.html#EECS2030\\_F21](https://www.eecs.yorku.ca/~jackie/teaching/lectures/index.html#EECS2030_F21)
- EECS2011 (Data Structures):  
[https://www.eecs.yorku.ca/~jackie/teaching/lectures/index.html#EECS2011\\_W22](https://www.eecs.yorku.ca/~jackie/teaching/lectures/index.html#EECS2011_W22)

– Here are some optional reference textbooks:

- TITLE: Engineering a Compiler  
AUTHOR: Keith D. Cooper & Linda Torczon  
PUBLISHER: Morgan Kaufmann Publishers Inc.  
EDITION: Second Edition (2012)  
ISBN: 978-0-12-088478-0

- Classic Text (the “dragon” book):  
TITLE: Compilers: Principles, Techniques, and Tools  
AUTHOR: Alfred V. Aho, Monica S. Lam, Ravi Sethi & Jeffery D. Ullman  
PUBLISHER: Addison-Wesley Longman Publishing Co., Inc.  
EDITION: Second Edition (2006)  
ISBN: 978-0-321-48681-3

## 6 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/EECS2001 3.00; LE/EECS2011 3.00

## 7 COURSE DESCRIPTION

This course covers principles and design techniques for compilers and interpreters, purpose and design of domain-specific languages (DSLs), compiler organization, compiler writing tools, scanning, parsing, semantic analysis, code generation, and if time permitted, run-time storage organization, memory management, and optimization.

This course is a hands-on introduction to the design and construction of compilers and interpreters. The main purpose is to have students acquire the techniques of defining their own languages and implementing tools to suit their own needs. At the end of the course, students will understand the architecture of compilers and interpreters, their major components, how the components interact, and the algorithms and tools that can be used to construct the components.

Students will be assigned a semester-long project divided into phases: e.g., designing the grammar of a domain-specific language, using tools to generate its scanners and parsers, and applying suitable software design patterns (e.g., composite, visitor) to build abstract syntax trees (ASTs) of input programs and to perform semantic operations (e.g., evaluation, type checking, code generations). Generated programs may take different forms: e.g., C code, database schemas/procedures, theorems/functions to a proof assistant, specification language to a model checker.

## 8 COURSE LEARNING OUTCOMES (CLOs)

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

**CLO1** Compare and contrast General-Purpose Languages (GPLs) and Domain-Specific Languages (DSLs).

**CLO2** Apply the theoretical understanding of, and use the relevant tools to generate, a lexical scanner.

**CLO3** Apply the theoretical understanding of, and use the relevant tools to generate, a grammatical parser.

**CLO4** Construct an abstract syntax tree & perform semantic operations on it.

**CLO5** Communicate the design and implementation of a DSL and its associated tools.

## 9 GRADING SCHEME

|                         |     |
|-------------------------|-----|
| 3 Assignments (2% each) | 6%  |
| Project                 | 19% |
| Programming Test        | 10% |
| Written Test            | 10% |
| Final Exam              | 55% |

## 10 FINAL EXAM: CUMULATIVE & SUBSTANTIAL

- Your final exam will be *cumulative*: it will cover all study materials.
  - It will be an opportunity for you to continually *synthesize* topics that are connected.
- Therefore, your final exam will be the most substantial grading component.
  - It assesses how competently you can apply the learned concepts and skills.
  - The best preparation is to constantly review and reflect on the covered topics.

## 11 MAPPING RAW MARKS TO LETTER GRADES

- 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.
  - Check the common **Grades and Grading Schemes**.
  - e.g., Say there are only two grading units: Exam (60%) and Lab1 (40%).  
Receiving 150 marks (out of 200) for Exam and 2 marks (out of 3) for Lab1 leads to a letter grade B (based on the weighted sum  $\frac{150}{200} \times 60 + \frac{2}{3} \times 40 \approx 71.7$ ).

## 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:
  - Weekly Lectures [  $\approx 3$  hours ]
  - Optional: Office Hours
- “Out-of-Class” Hours:
  - Completing Assignments, Studying for Lectures/Tests [ 6 to 10.5 hours ]
- You may find yourself needing more time to digest the materials and build the skills. The harder you work, the easier you may find in your future career endeavours.

## 13 ATTENDANCE OF CLASSES: ENCOURAGED & REWARDING

- There are 23 upcoming in-class lectures in total (2 classes  $\times$  12 weeks – first class).
  - Attending classes (in-time & focused) is an indispensable part of your learning.
  - Despite it being your responsibility, Jackie would encourage you to attend classes by the following rewarding scheme:
    - Attendance will be taken randomly (via iClicker) on  $X$  classes ( $10 \leq X \leq 23$ )  
 $\Rightarrow$  Attendance will be checked somewhere between every class and every other class.  
*In a class where attendance is taken, one more more checks may be conducted: your attendance of that class will not count if you miss any of the checks.*  
 For example, if you wait for the first check to occur and choose to leave right afterwards, your attendance will not count as you may miss the subsequent check(s). That is, *your attendance to a class will count only if you complete all checks.*
    - Each attendance check will be conducted briefly (e.g., for a few minutes) at some-time between **5 minutes** after class starts and **5 minutes** before class ends.
    - No makeup attendance will be considered if you missed a check because you, e.g.,
      - \* arrived late
      - \* left early
      - \* did not pay attention or was absent when the attendance check took place
    - Please always have the iClicker launched on your computer or mobile device:
      - \* There will be a sign-up sheet to accommodate the (extremely) rare occurrences of failed check-ins.  
 You will be accommodated to sign on a sheet for **a maximum of 2 classes**.  
 $\Rightarrow$  *You are solely responsible for resolving any technical issues that caused you to fail checking in via the installed iClicker.*
- e.g., see: <https://mhe.my.site.com/iclicker/s/article/How-to-Troubleshoot-Your-Connection-to-the-iClicker-Student-App>

- At the end of the semester, say you attended  $Y$  classes:

```

if       $Y \geq \lfloor 90\% \cdot X \rfloor \rightarrow 5\% \text{ bonus}$ 
elseif  $Y \geq \lfloor 80\% \cdot X \rfloor \rightarrow 4\% \text{ bonus}$ 
elseif  $Y \geq \lfloor 70\% \cdot X \rfloor \rightarrow 2\% \text{ bonus}$ 
elseif  $Y \geq \lfloor 60\% \cdot X \rfloor \rightarrow 1\% \text{ bonus}$ 
elseif  $Y \geq \lfloor 50\% \cdot X \rfloor \rightarrow .5\% \text{ bonus}$ 
elseif  $Y < \lfloor 50\% \cdot X \rfloor \rightarrow \text{no bonus}$ 

```

- For examples:

$X = 23$  (check at every class)     $X = 10$  (check at every other class)

```

if       $Y \geq 20.7 \rightarrow 5\% \text{ bonus}$ 
elseif  $Y \geq 18.4 \rightarrow 4\% \text{ bonus}$ 
elseif  $Y \geq 16.1 \rightarrow 2\% \text{ bonus}$ 
elseif  $Y \geq 13.8 \rightarrow 1\% \text{ bonus}$ 
elseif  $Y \geq 11.5 \rightarrow .5\% \text{ bonus}$ 
elseif  $Y < 11.5 \rightarrow \text{no bonus}$ 

```

```

if       $Y \geq 9 \rightarrow 5\% \text{ bonus}$ 
elseif  $Y \geq 8 \rightarrow 4\% \text{ bonus}$ 
elseif  $Y \geq 7 \rightarrow 2\% \text{ bonus}$ 
elseif  $Y \geq 6 \rightarrow 1\% \text{ bonus}$ 
elseif  $Y \geq 5 \rightarrow .5\% \text{ bonus}$ 
if       $Y < 5 \rightarrow \text{no bonus}$ 

```

- The above rewarding scheme **only** applies to in-class lectures.
- The allowable quota for you to miss classes, so as to get a particular bonus, already accommodates valid excuses (e.g., sick, family emergency).

Therefore:

- once the attendance-taking window expired, **no** late responses will be accepted;
- **no** reasons will be considered for missing attendance checks.

- The instructor reserves the right to **cancel** your bonus if, e.g.:
  - you just wait to be taken attendance and leave the class shortly after;
  - you attend classes but cause distractions (e.g., talking, using devices for irrelevant activities) to the instructor and/or to other students.

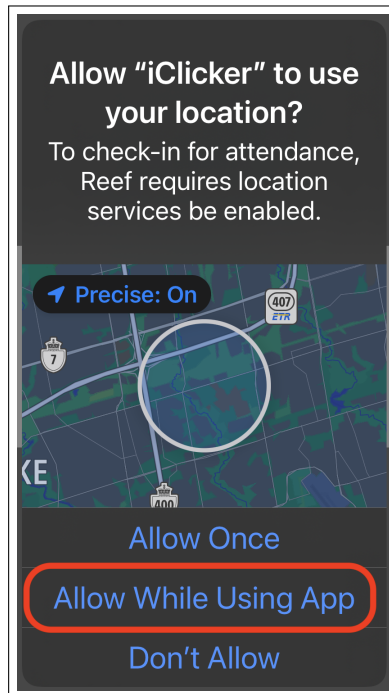
– What should I do to set up the iClicker for attendance checks?

- Refer to this starter guide (to install iClicker on your mobile device):

[https://lthelp.yorku.ca/polling-students/  
iclicker-student-app-quick-start-guide](https://lthelp.yorku.ca/polling-students/iclicker-student-app-quick-start-guide)

When creating an iClicker account, be sure to supply your **student number** and **...@my.yorku.ca** email (you are responsible for **not** receiving the bonus if an invalid student number or email is supplied).

- Ignore the first section “**For Courses using eClass integration**”.
- Follow these sections:
  - \* “**For Courses not using eClass integration**”
  - \* “**Add Your Instructor iClicker Course**”:  
EECS4302-A (S’25) - Compilers & Interpreters
  - \* “**Respond to Polls**”
- When launching iClicker, it is critical that you allow iClicker to use your location; otherwise you will not be able to join the course and take attendance.





## 14 SEMESTER CALENDAR

Figure 1 summarizes the schedule of required work items:

- Each (programming or written) test will take place during the scheduled lecture time.
- Specific details for each (programming or written) test will be announced in advance.

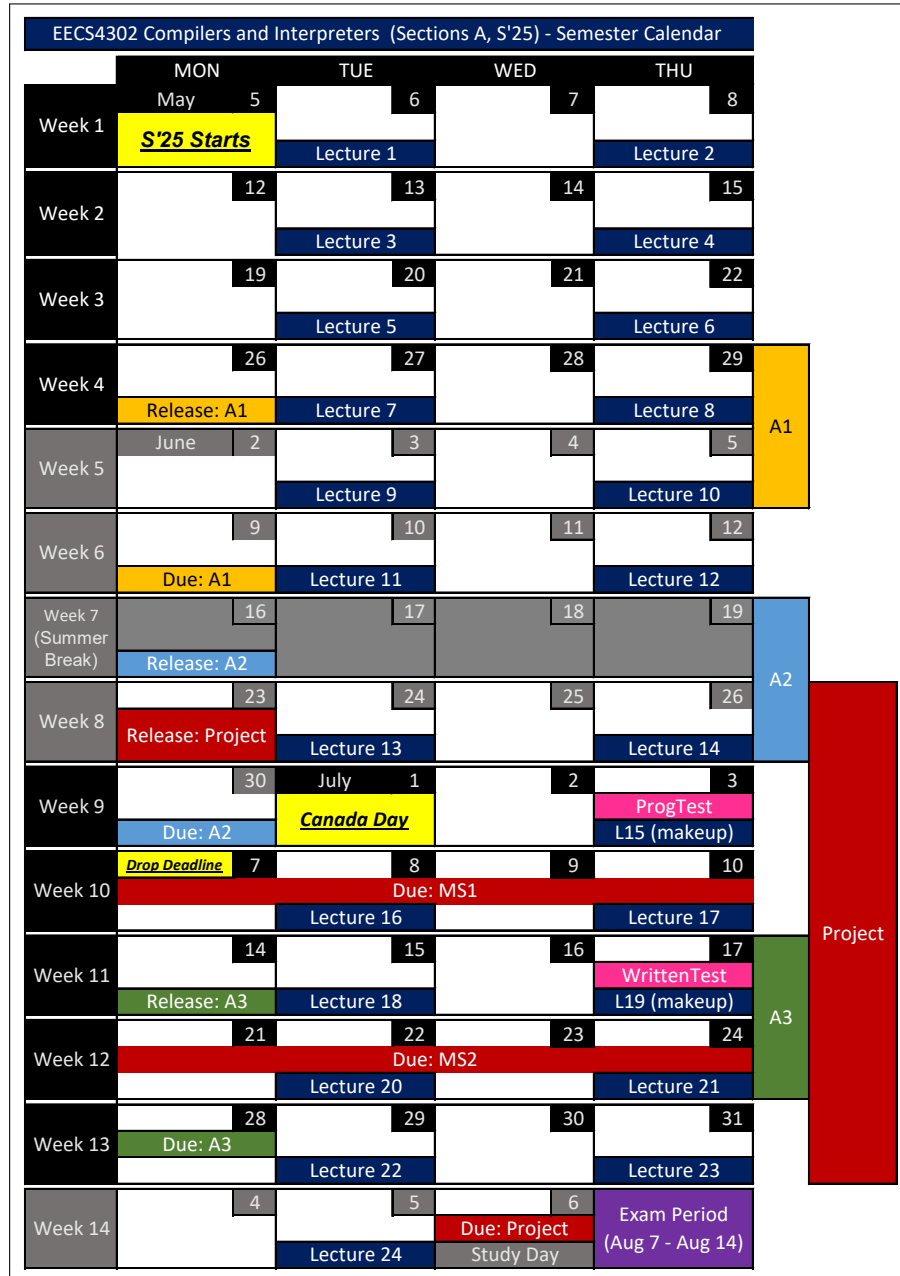


Figure 1: EECS4302-A S25 Semester Calendar – Expected Work Items

## 15 COVERAGE OF QUIZZES

Tentatively, referencing the semester calendar in Figure 1 (p9):

- Programming Test covers A2 and the ANTLR4 tutorial (to be assigned during the study week).
- Written Test covers Lectures 1 – 17

## 16 WEEKLY SCHEDULE

In the time table below, each cell denotes a 30-minutes interval.

- Cell 8:30 denotes the interval starting at 8:30 and ending at 9:00.
- For example, the in-person office hours (on Tuesdays and Thursdays) occupies 2 cells indicates that it lasts for 1 hour (starting at 12:30 and ending at 13:30).

|       | Monday | Tuesday                          | Wednesday            | Thursday                         |
|-------|--------|----------------------------------|----------------------|----------------------------------|
| 8:30  |        |                                  |                      |                                  |
| 9:00  |        |                                  |                      |                                  |
| 9:30  |        |                                  |                      |                                  |
| 10:00 |        |                                  |                      |                                  |
| 10:30 |        |                                  |                      |                                  |
| 11:00 |        |                                  |                      |                                  |
| 11:30 |        |                                  |                      |                                  |
| 12:00 |        |                                  |                      |                                  |
| 12:30 |        | Office Hours<br>LAS2043          |                      | Office Hours<br>LAS2043          |
| 13:00 |        |                                  | Office Hours<br>Zoom |                                  |
| 13:30 |        |                                  |                      |                                  |
| 14:00 |        |                                  |                      |                                  |
| 14:30 |        |                                  |                      |                                  |
| 15:00 |        |                                  |                      |                                  |
| 15:30 |        |                                  |                      |                                  |
| 16:00 |        |                                  |                      |                                  |
| 16:30 |        |                                  |                      |                                  |
| 17:00 |        |                                  |                      |                                  |
| 17:30 |        | EECS4302 A<br>Lecture<br>LSB 105 |                      | EECS4302 A<br>Lecture<br>LSB 105 |
| 18:00 |        |                                  |                      |                                  |
| 18:30 |        |                                  |                      |                                  |

## 17 (TENTATIVE) TOPICS TO BE COVERED

- Overview of Compilation
- Lexical Analysis (Scanners)
- Syntax Analysis (Parsers)
- Use of a Parser Generator (ANTLR)
- OOP Design Patterns: Composite & Visitor
- Context-Sensitive Analysis
- Intermediate Representation
- Optimization