

## CSCI 605: Advanced Object Oriented Programming Concepts Spring '24 (2245), section 1 (in-person, hybrid)

### Course Information

**Instructor:** Eduardo Lima

Email: [lima@cs.rit.edu](mailto:lima@cs.rit.edu)

Phone: Send me an email or discord message.

Office: [Zoom office](#) / 70-3515

Office Hours: <https://cs.rit.edu/~lima>

**Class Meeting Times:** check SIS

**Location:** Check SIS (in-person), click navbar/zoom link (online)

**Hybrid students:** use the zoom link in mycourses navbar.

**Course Description:** This course focuses on identifying advanced object-oriented programming concepts and implementing them in the context of specific problems. This course covers advanced concepts such as event-driven programming, design patterns, distributed and concurrent programming, and the use, design and implementation of applications. Assignments (both in class and as homework) requiring a solution to a problem and an implementation in code are an integral part of the course. Note: This course serves as a bridge course for graduate students and cannot be taken by undergraduate students without permission from the CS Undergraduate Program Coordinator.

**Course Learning Outcomes:** Students will demonstrate fluency in high-level object-oriented programming languages, environments, and tools for computing. Assessment: Homework and Exams.

**Prerequisites:** Introductory Programming and Algorithms course

**Required Textbook and Other Course Materials:** None

**Lecture Notes:** There is a [LINK](#) in mycourses that take you to a Google Drive folder with all lecture notes. I will NOT post source code files in mycourses.

**Lecture Source Code and Memo files:** I will keep updating this public repository with the source code used during lectures. I recommend you clone and regularly get updates (pull it). Check mycourses for the link. A list of topics expected to be covered is available in this [spreadsheet](#).

**Questions and Answers:** I will use a discord server for answering questions outside of the classroom. It is a good idea to join the server as soon as possible. You can always email me, but unless strictly required, I recommend you use the discord server as our main discussion forum. Link is in mycourses.

## Grading

Course grades will be based on the following:

- Midterm Exam 1 25%
- Midterm Exam 2 25%
- Weekly Homeworks 25%
- Final Exam 25%

Final grades will be the weighted average of the scores, based on the percentages above, and not curved (A=93+, A-=90-93, B+=87-90, B=83-87, etc.). No extra credit. You need a B to pass this course.

**Make-ups:** Make-ups for graded activities may be arranged if your absence is caused **by illness or personal emergency**. A written explanation (including supporting documentation) must be submitted to your instructor; if accepted, an alternative will be arranged. Make-up arrangements must be requested **prior** to the exam or activity.

**Late Policy:** Any homework assignment may be submitted up to 8 hours late with a late penalty of 20%. After that, no credit will be given. You must use mycourses dropbox for your submissions, email will not be accepted.

**Grading Sessions:** You will work in pairs for the homework assignments. Each team will meet with their designated grader every week to discuss their solution. All team members have to be able to explain all solutions. Although you will work as a team, you will be evaluated individually—the grader chooses whom to ask.

Note: You will receive zero points if you are late to your grading session or if you can not explain your solution.

First Week: make sure to form a 2-person team and sign-up for a grading slot using a spreadsheet. Link to the spreadsheet is in mycourses.

**Coding Style:** Students must follow the [CS style recommendation](#) for Java in this course. Here is an [Example](#).

**Required Software:** [How to install Java, JavaFX, IntelliJ](#)

JDK - <https://www.oracle.com/java/technologies/javase-downloads.html>

JavaFX - <https://gluonhq.com/products/javafx/> (only if needed)

IDE - <https://www.jetbrains.com/idea/> (I will use IntelliJ, but you are free to use any IDE)

## Course Schedule (Subject to Change)

| <u>Week</u> | <u>Class Dates</u> | <u>Topic</u>                                   | <u>Assignments</u> | <u>check mycourse<br/>for due dates</u> |
|-------------|--------------------|--|--------------------|---|
| 1           |                    | OOP, Java Basics: Data types, Control flow     |                    |   |
| 2           |                    | Classes  | Homework1          |   |
| 3           |                    | Interfaces, Inheritance, Abstract Classes      | Homework2          |   |
| 4           |                    | <i>Patterns</i>                                | Homework3          |   |
| 5           |                    | <b>Exam1</b> , Generics                        |                    |   |
| 6           |                    | Collections Framework                          | Homework4          |   |
| 7           |                    | Basic IO, Exceptions                           | Homework5          |   |
| 8           |                    | <i>Java FX</i>                                 | Homework6          |   |
|             |                    | BREAK  |                    |   |
| 9           |                    | Threads Intro                                  | Homework7          |   |
| 10          |                    | <b>Exam 2</b> , Threads: Synchronization, Join |                    |   |
| 11          |                    | Threads: Wait, Notify/All                      | Homework8          |   |
| 12          |                    | Networking: Tcp, Udp, Multithreading           | Homework9          |   |
| 13          |                    | Threads: Producer/Consumer                     | Homework10         |   |
| 14          |                    | Lambda, Stream API (time allowing)             |                    |   |
| 15          |                    | Review, <b>Final Exam</b>                      |                    |   |

### Miscellaneous

**Mental Health:** The academic demands in this course and your other classes can be understandably difficult. It is normal to feel anxious about your academic ability, especially when unexpected life events emerge. I want to invite you to connect with me about any difficulties you have in this course as soon as possible. Your success is important. It is important to get additional assistance if needed before the challenges become too much.

**Reasonable Accommodations:** RIT is committed to providing reasonable accommodations to students with disabilities. If you would like to request accommodations such as special seating or testing modifications due to a disability, please contact the Disability Services Office. It is located in the Student Alumni Union, Room 1150; the Web site is [www.rit.edu/dso](http://www.rit.edu/dso). After you receive accommodation approval, it is imperative that you see me during office hours so that we can work out whatever arrangement is necessary.

**Academic Integrity:** As an institution of higher learning, RIT expects students to behave honestly and ethically at all times, especially when submitting work for evaluation in conjunction with any course or degree requirement. The Department of Computing Security encourages all students to become familiar with the RIT Honor Code and with RIT's Academic Integrity Policy. Students may discuss assignments with others including classmates, tutors and SLI instructors. After any such discussions, students must discard all written notes/pictures/etc. Submitting **any** work written by others or as an unsanctioned team is considered an act of academic dishonesty. Team-developed work also must be created solely by the team members and not copied from others or other sources. Work copied from Github or other similar sources will be subject to prosecution for breach of academic integrity. Any version control repository used must be private.

**Publishing your work:** Any version control repository used must be private unless explicitly specified by the instructor.

**Generative AI use:** You can not use generative AI code generation unless explicitly specified by the instructor.

**Copyright:** Certain materials used in this course are protected by copyright and may not be copied or distributed by students. You can find more information at: [http://www.rit.edu/academicaffairs/policiesmanual/sectionC/C3\\_2.html](http://www.rit.edu/academicaffairs/policiesmanual/sectionC/C3_2.html).

**Emergency / Public Safety:** Under emergency circumstances, we may have to alter course requirements, assignment deadlines, attendance expectations, and grading procedures. The University may also have to alter the academic calendar. Visit [RIT's Emergency landpage](#) for more information.

**Academic Continuity:** in case of challenges such as the instructor being sick, alternative learning delivery will be used, e.g. pre-recorded lectures, zoom lectures. The instructor will always communicate with the students via email, mycourses announcements or discord.

**Final Note:** *The instructor retains the right to change anything, but will make a good faith effort to notify you in advance.*