

CSCI 3010: Programming Project Workshop: Object-Oriented Programming

Instructor: Felix Muzny¹ (pronunciation: "Muse-knee"; pronouns: they/them and he/him)

Contact: muzny@colorado.edu

Office: ECOT 734 (engineering center tower)

Credit: 3 credits

Lecture times, section 001: Tuesdays & Thursdays 1 - 2:15pm, FLMG 33

Lecture times, section 002: Wednesdays & Fridays 9:30 - 10:45am, FLMG 33

CAs: Faisal Lalani, James McDonald

Graders: Nikhil Jain, Jay Patel

Office Hours: calendar on Canvas

Course Overview

This course is an intensive exercise in practicing writing code. We'll go from more structured assignments and exercises where you are given very specific guidelines as to how your programs should be structured to designing your own programs from the ground up. The main goal of this course is for you to emerge confident that you would know how to start from a blank page, produce a design for a piece of software, then translate that design into actual code.

We'll spend time working with version control, writing code reviews, integrating unit testing, producing prototypes, and dip our toes into the world of design patterns alongside a diversity of other programming topics.

Course Goals

1. Increase programming confidence and competency.
2. Understand how multi-file programs work and how to design them.
3. Understand larger considerations that factor into the design and development of software, such as UI design, testing, and language restrictions.

Topics

- Objects and object design in object-oriented programming
- Version control and code reviews
- Writing generalizable code
- Unit testing
- Design patterns in object oriented programming
- Designing and building Graphical User Interfaces (GUIs)
- Generalizing programming languages
- A few miscellaneous topics, such as cryptography and security

¹ Call them "Felix" or "Professor Muzny"

Classroom Environment & Expectations

CSCI 3010 is a class that requires you to code a lot, both in class and outside of class, in a wide variety of circumstances. The in class activities are specifically designed to support the programming exercises and homework that you'll be doing outside of class. Most students find that active participation and collaboration with classmates during lecture time is invaluable to their success. If you are someone who doesn't often speak up, I encourage you to challenge yourself to speak up more. If you are someone who often speaks up, I encourage you to step back and take on an active listener role more than you may be used to.

- **Preparation:** Completing the Tuesday programming exercises before the first class of the week and getting started on the Friday programming exercises before the second class of the week will leave you well prepared for the material that we cover.
- **Classroom environment:** It is unusually common in Computer science classes for some students to ask questions that are not really questions so much as opportunities to demonstrate knowledge of vocabulary or facts beyond the topic at hand. This can have a discouraging effect on other students who are not familiar with those terms, causing them to worry that they are less prepared to do well in the class (this is rarely the case—knowing terms outside the scope of the course is not a good predictor of success). If you find yourself wanting to make such a question or comment, please come talk to me about the topic after class or during office hours—I'm always happy to discuss tangentially related topics at those times!
- **Attendance:** I expect all students to attend lecture and to actively participate in the group and individual work that we do.
- **Accommodation letters:** If you have an accommodations letter, please bring it to me at your earliest convenience so that I can make sure this class is meeting your needs.
- **Name and pronouns:** If your name and pronouns are not in alignment with those listed on our class roster, please let me know either in person or via email so that I can ensure you are correctly addressed in this class.
- **Class expenses:** If obtaining any material for use in our class presents a financial hardship for you, please let me know and I will work with you to locate the resources that you need to succeed in this class.
- **Feedback:** Please don't hesitate to reach out to me if any aspect of this course or class community could be improved.

Late Policy

All homework may be turned in up to 3 days (72 hours) late for a 20% penalty. If homework is due on Friday at 6pm, it may be turned in as late as Monday at 6pm. If a student would have received a 95% had they turned their homework in on time, a late submission will earn them a 75% instead.

In class activities must be completed in class and you must be present in class to receive credit for in-class activities. Submission items will stay open until 11:59 the day of class should you wish to continue working, but you are expected to turn in what you have at the end of class.

Programming exercises may not be turned in late.

Make-Up Policy

If you are unable to class on a particular day, it is your responsibility to contact the instructor beforehand. If you contact the instructor at least 48 hours in advance, it may be possible for you to make-up the work for partial or full credit depending on the planned activity for the day. Other extensions will only be given in cases of medical and family emergencies.

You must acquire instructor approval to attend the other section prior to the earlier of the two class times.

Example: A student is enrolled in 3010-002 and has a dentist appointment during class on Wednesday. They contact Felix on Monday to request attending 3010-001 on Tuesday instead. They get approval, attend Tuesday class, and earn credit for their work.

Collaboration Policy

You are expected to do your work yourself or with your partner when indicated.

Here are three big-picture points to remember when collaborating with your classmates:

- **Strategies:** You may talk with your classmates about *general strategies* but you may not talk about *specific solutions*.
- **Explaining concepts:** You may talk with your classmates about how certain techniques work *in general* but not how to write any part (or sub-part) of the solution needed for the homework.
- **A good rule of thumb:** don't show your assignments to other people; don't look at other people's assignments; don't write code together unless the assignment explicitly states that you may work in pairs. This includes working through solutions on whiteboards as well as telling your friend verbally what you have written.

The finer-grained details:

- **Do not search for a solution online:** You may not actively search for a solution to the problem from the internet. This includes posting to sources like StackExchange, Reddit, Chegg, etc.
 - **StackExchange Clarification:** Searching for basic techniques in C++ is fine. If you want to post and ask "How do convert a float to an integer" that's fine. What you **cannot** do is post are things like "Here's the function my prof gave me to write. I need to convert this temperature in celcius to fahrenheit".
- **Plagiarism:** assignments **and code** that you turn in should be written entirely on your own. You should not need to consult sources beyond the class notes, posted lecture notes, examples, and resources, and c++ documentation and pattern examples.
- **Tutors:** you should always consult the TAs and LAs for this course if you need extra help. They are here specifically to help you! You should never have anyone else write code for you. This includes tutors, friends, strangers, friends of friends, or anyone who is not you.
 - **A second note on tutors:** our TAs are here for you. They are not paid to do your assignments, but to help you learn. If you are using a tutor that is not helping you learn, **they are not helping you**.
 - **A third note on tutors:** if you turn in work that you are unable to explain, this will result in an honor code violation and a loss of credit.
- **When in doubt, ask:** If you have doubts about this policy or would like to discuss specific cases, please ask the instructor.

Collaboration Policy violations will result in both a 0 on the assignment in question and a final grade reduction of one letter grade.

Grading & Assignments

Category	Due Dates & Points	Grade Percentage
Programming Exercises	Due Tuesdays at 12pm and Fridays at 6pm Most exercises are worth 20 points. There are a total of 280 possible points in this category. Points over 250 are extra credit.	25%
Homework	Due Fridays at 6pm 600 points total.	60%
In Class Activities	Every class session. 5 points each.* * Exception—last week of class for student presentations. Credit will be given but will not follow the 5 points/class scheme. 150 points total	15%

This course doesn't have any midterms or final exams.

Individual Projects vs. Homeworks

This course is unique in that it offers the option to students to design their own, 10-week long project to work on.

When Homework 2 is due, each student has the option to either submit the standard homework 2 or to submit a project proposal.

Students who submit proposals will be required to meet with the instructor to discuss scope, appropriateness, and project-specific topics. Projects will either be approved or not at this point. Approval depends largely upon: a) scope and appropriateness to the course and b) programming and other skills (such as time management) that students have demonstrated so far.

If your project is not approved, you will return to completing the standard set of homework assignments. As a guideline, in the first iteration of this course, about $\frac{2}{3}$ of students in the course completed the standard set of homework assignments.

More information regarding individual projects will be released during the third week of class. If you have questions about whether or not a specific project might be appropriate, you are encouraged to come by Felix's office hours!

Calendar

Week (Monday date)	Topics	Assignments due this week
8/26	C++, Object-oriented programming review	
9/2	Privacy, overloading, const, constructors, and enums	PE 1, PE 2
9/9	Terminal, bash, PS1, git integration and distributed version control	PE 3, PE 4
9/16	Continuous integration, UML diagrams, unit testing	PE 5, HW 1
9/23	Templating and generalizability, inheritance	PE 6
9/30	Static, design patterns	PE7, HW 2
10/7	Design patterns, technical interviews	PE 8
10/14	Design patterns, low-fidelity prototyping	HW 3
10/21	User testing, implementing GUIs	PE 9, PE 10
10/28	Implementing GUIs	PE 11
11/4	TBD, final projects	HW 4
11/11	Backends, apis, and servers	PE 12, PE 13
11/18	TBD, projects workshop (user testing)	Final Project Checkpoint
11/25	Fall Break (no classes)	
12/2	Cryptography, security	PE 14
12/9	Final Project Presentations	
Interview Grading Final Project		

Accommodation for Disabilities

If you qualify for accommodations because of a disability, please submit your accommodation letter from Disability Services to your faculty member in a timely manner so that your needs can be addressed. Disability Services determines accommodations based on documented disabilities in the academic environment. Information on requesting accommodations is located on the [Disability Services website](#). Contact Disability Services at 303-492-8671 or dsinfo@colorado.edu for further assistance. If you have a temporary medical condition or injury, see [Temporary Medical Conditions](#) under the Students tab on the Disability Services website.

Classroom Behavior

Students and faculty each have responsibility for maintaining an appropriate learning environment. Those who fail to adhere to such behavioral standards may be subject to discipline. Professional courtesy and sensitivity are especially important with respect to individuals and topics dealing with race, color, national origin, sex, pregnancy, age, disability, creed, religion, sexual orientation, gender identity, gender expression, veteran status, political affiliation or political philosophy. Class rosters are provided to the instructor with the student's legal name. I will gladly honor your request to address you by an alternate name or gender pronoun. Please advise me of this preference early in the semester so that I may make appropriate changes to my records. For more information, see the policies on [classroom behavior](#) and the [Student Code of Conduct](#).

Honor Code

All students enrolled in a University of Colorado Boulder course are responsible for knowing and adhering to the Honor Code. Violations of the policy may include: plagiarism, cheating, fabrication, lying, bribery, threat, unauthorized access to academic materials, clicker fraud, submitting the same or similar work in more than one course without permission from all course instructors involved, and aiding academic dishonesty. All incidents of academic misconduct will be reported to the Honor Code (honor@colorado.edu); 303-492-5550). Students who are found responsible for violating the academic integrity policy will be subject to nonacademic sanctions from the Honor Code as well as academic sanctions from the faculty member. Additional information regarding the Honor Code academic integrity policy can be found at the Honor Code Office website.

Sexual Misconduct, Discrimination, Harassment and/or Related Retaliation

The University of Colorado Boulder (CU Boulder) is committed to fostering a positive and welcoming learning, working, and living environment. CU Boulder will not tolerate acts of sexual misconduct intimate partner abuse (including dating or domestic violence), stalking, protected-class discrimination or harassment by members of our community. Individuals who believe they have been subject to misconduct or retaliatory actions for reporting a concern should contact the Office of Institutional Equity and Compliance (OIEC) at 303-492-2127 or cureport@colorado.edu. Information about the OIEC, university policies, [anonymous reporting](#), and the campus resources can be found on the [OIEC website](#).

Please know that faculty and instructors have a responsibility to inform OIEC when made aware of incidents of sexual misconduct, discrimination, harassment and/or related retaliation, to ensure that individuals impacted receive information about options for reporting and support resources.

Religious Holidays

Campus policy regarding religious observances requires that faculty make every effort to deal reasonably and fairly with all students who, because of religious obligations, have conflicts with scheduled exams, assignments or required attendance. In this class, contact the course staff at least 7 days in advance to reschedule a test. Contact both your TAs and the TAs of the lab that you wish to attend at least 24 hours before the earlier of the two to reschedule a lab.

See the [campus policy regarding religious observances](#) for full details.