

CSC305: Course Syllabus

Learning Environment

My teaching philosophy focuses on providing the best possible learning environment, creating multiple ways of learning, and designing all the additional tools necessary for students to understand and appreciate a subject. All members of this class are expected to treat each other with respect and kindness; together, we succeed.

Cal Poly believes in community building. My goal as an instructor is to help you in every way I can to provide you with the best learning experience. If you have any questions or concerns, please do not hesitate to seek help from me. Participation in this course is highly welcomed and encouraged.

Personal Well-being

If you are feeling overwhelmed with course work and personal obligations, note that I completely understand. In these situations, your coursework may suffer and you may fall behind on some course deadlines. Your personal well-being is extremely important to me. Kindly reach out to me in these situations so that we can work out a solution to help you attain your goals.

If you or someone you know is having difficulty accessing enough healthy food or stable housing, please know that our campus offers a number of free and effective resources. Visit <http://www.basicneeds.calpoly.edu/> for more information.

Course Learning Objectives

After completion of this course, you will be able to:

- Discuss the features of a standard OO language
- Use a standard debugger
- Use Makefiles or a standard IDE
- Develop and properly organize multi-source file projects
- Use modularity in building a project
- Use module-level testing
- Use a language-based assertion facility
- Use and design generic classes
- Use, at an intermediate level, a language-standard class library
- Describe single inheritance, including abstract base classes

- Describe class members and methods
- Use polymorphic methods and interface types
- Implement and evaluate for use a selection of object-oriented design patterns
- Use Application Frameworks
- Use functional programming and write lambda expressions
- Individually write and test mid-sized object-oriented software modules of a professional quality, using a standard OO programming language, and using the concepts listed above
- Use concurrency libraries and language constructs including techniques for synchronization

Text Books

To support our study we will reference "**Design Patterns: Elements of Reusable Object-Oriented Software**" by Gamma et al. and "**Effective Java**" (3rd Edition) by Joshua Bloch. These are excellent resources that will serve you well beyond the course (these books are available through the **Kennedy Library**) (<https://lib.calpoly.edu/>)

Note that the primary source of learning in this class will be the slides covered in the lecture classes that will be made available after each class on Canvas.

Activities

1. Questions/Queries/Comments

Make sure that you attend lectures to understand the concepts presented in this course. The lectures are for your benefit, and you should use this time to ask questions, to discuss general approaches to the projects, to clarify the material.

2. In-Lecture Code/Design Critiques

Most lectures will include one or multiple code or design samples to be critiqued. These can be done either individually or in small groups. Your participation in the live in-lecture discussion is welcome and encouraged. These critiques are not assessed for correctness but rather to see if your response contains careful consideration of the assignment problem.

3. Lecture Quizzes

There will be multiple low-risk formative assessment quizzes to assess comprehension of lecture material and its applicability to the projects. These are generally posted at the end of each week and will be due at the beginning of the next week.

4. Labs

Each week there will be a lab activity that is designed to develop and apply the skills discussed in lecture. These labs will help in better understanding and implementation of the lecture material and will support the course project. You can discuss and work on these lab activities with your classmates.

5. Project Assignments

There is one project with multiple milestones listed as individual assignments in the schedule. These projects are meant to supplement the skills and knowledge acquired in the lectures and labs. Note that project assignments are typically larger in scale as compared to lab assignments. Kindly don't wait until the last moment if you are stuck on any of these project assignments; ask for help immediately (e.g., in lab or office hours, or even email or Canvas).

6. Final Project Analysis

In lieu of a traditional final exam, you will submit a reflection on and analysis of the use of design principles within your project. More details will be provided near the end of the quarter, but the general idea is that this mimics, to a degree, a (scaled down) project handoff or the on-boarding of a new team member. The analysis itself will be submitted as a paper.

Grading

The course grading breakdown is as follows. Final course grades will be assigned roughly according to a standard scale, but with consideration for performance by the class as a whole.

Activity	% Of Total
Labs	10
Quizzes	15
Project Assignment-1	15
Project Assignment-2	15
Project Assignment-3	15
Project Assignment-4	15
Final Project Analysis	15

Course Policies

The following course policies have been established to help you succeed in this course and are shared here for your reference.

1. Communication Policy

My communication policy includes regular in-person communication before and after lecture, during lab, and in passing in the halls of Building 14 or any other common area. For on-line communication, there are multiple ways of reaching out, including Canvas discussion forums and email. Please take advantage of these methods to reach out for assistance so that I can help you.

Canvas Discussion Forum: I regularly monitor the Canvas Discussion forum for questions about labs, project assignments and the course material. General programming and concept questions should be made public so that all students can benefit from and contribute to responses. Questions with code specific to a solution should be made private. Of course, treat each other with kindness; and respect; I will not tolerate inappropriate posts.

email: You can also email (jmukherj@calpoly.edu) me with any questions, concerns, or issues that you may have. Though the Canvas forum provides the benefit for shared discussion of course material and is generally preferred for that reason, I understand that email is a more appropriate method for some communications.

You can contact me any day of the week and at any time. I will respond to each email and Canvas post within 24 hours, but often much more quickly. Please do not hesitate to contact me but give me adequate time for a response. Even though I will likely respond very quickly, you should not rely on my ability to do so very close to a deadline.

2. Late Work Policy

It is important to turn in your assignments on time. In case you can not do so due to an emergency or any unforeseeable circumstances, kindly contact me at your earliest.

You may submit your code for late project and lab assignments for a maximum grade of 70% of the project/lab credit. The due date for late submissions will be 7 days after the original due date. After 7 days of the original due date, late submissions will no longer be accepted.

3. Code/Design Critique Policy

These are regular activities that demonstrate active participation in the course. These activities are designed to help students understand the course material through an effective, engaging way. Missing out on these will not affect your grade in a major way, but it will concern me. Expect to hear from me in case you are not actively participating in the class activities.

4. Quiz Policy

You will have multiple days to complete each quiz (Posted on Canvas). The goal is for you to assess your understanding of the material leading into project assignments. ***Late submissions are not accepted for quizzes.***

5. Lab Policy

Each lab is assigned at the beginning of the week (available on Monday mornings) and due at the end of the next week. Note that although the labs are to be submitted individually, the labs have to be demonstrated to the instructor in the lab classes, either individually or as a group. ***Note that failing to complete any of these two activities (in-person demonstration and code submission) by the given lab submission deadline will result in your lab assignment ungraded.***

6. Project Assignment Policy

Each project assignment is designed to help you in implementing a component of your course project by yourself. The project submission deadlines have been assigned keeping in mind the project complexity.

7. Collaboration Policy

Talk, share ideas, help each other.

- Code/Design critiques are to be done in lecture classes where you can discuss in a team, but you are still required to respond individually. Please note that I am not looking for the correct answer in these activities, but trying to encourage teamwork and critical thinking.
- Lab activities allow for full collaboration between members of the class, but bear in mind that these are meant to help each of us develop our individual skills.
- Project assignments are individual efforts. You can talk to your classmates about the assignments, and you can share ideas, but make sure that you are submitting your own work and that you are learning from the experience.

Fully individual assessments

- Quizzes are meant to assess your understanding of the lecture material and must be attempted by only the individual.
- The final analysis must be completed individually without assistance from others.

With everything being said, I think you all recognize that we have been going through an exceptional time in the recent past. While attempting to return back to normal this Fall 2022 quarter, remember that life, similar to programming code, often throws exceptions back at you. Do not hesitate to contact me for any such reason that may be unique to your situation. Always remember: *“Together, we succeed.”*

8. Course Syllabus

Week	Date	Lecture Topic
1	19 Sep	Course Overview and Basic Introduction
	21 Sep	OO Practices and Principles
	23 Sep	OO Practices and Principles
2	26 Sep	OO Practices and Principles
	28 Sep	OO Practices and Principles
	30 Sep	OO Practices and Principles

3	3 Oct	Assignment-1
	5 Oct	Testing tools and Strategies
	7 Oct	Testing tools and Strategies
4	10 Oct	Generalizations- generics
	12 Oct	Generalizations- generics
	14 Oct	Generalizations- generics
5	17 Oct	Design Patterns Assignment-1 Submission Deadline on 16 Oct Assignment-2
	19 Oct	Design Patterns
	21 Oct	Design Patterns
6	24 Oct	Design Patterns
	26 Oct	Design Patterns
	28 Oct	Design Patterns
7	31 Oct	Design Patterns Assignment-2 Submission Deadline on 30 Oct Assignment-3
	2 Nov	Design Patterns
	4 Nov	Design Patterns
8	7 Nov	Generalization- Lambda Expressions
	9 Nov	Generalization- Lambda Expressions

	11 Nov	Holiday
9	14 Nov	Generalization- Lambda Expressions Assignment-3 Submission Deadline on 13 Nov Assignment-4
	16 Nov	Concurrency
	18 Nov	Concurrency
Thanksgiving Week		
10	28 Nov	Concurrency
	30 Nov	Concurrency
	2 Dec	Final Analysis Work Day Assignment-4 Submission Deadline