

C/C++ Programming IV:
Advanced Programming with Objects
Syllabus



Course Number: CSE-40478
Section ID: 152488
Quarter: Winter 2021
Course Dates: 1/12/2021 – 3/13/2021

Instructor Information

Name: Raymond Mitchell III
Email: ray@raymondmitchell.com

Communication Policy

You may contact me by email (ray@raymondmitchell.com) or via the Canvas email feature. My goal is to **respond within 24-48 hours**. If you do not receive a response during this period, please follow-up to ensure that your email was received.

Please communicate with me early and often if you are experiencing challenges in completing course assignments on time.

Course Description (Goals and Objectives)

Learn to write safe C++ code. Understand how design decisions can impact the safety and usability of the code you write. This course focuses on writing real-world (non-academic) C++ code. Learn to code defensively to ensure your programs handle error situations gracefully. Homework assignments reinforce the concepts covered each week. Projects provide experience with larger systems of interacting objects. Course highlights:

- Defensive coding
- Interface design
- Considerations when designing custom data types

- Design decisions behind the C++ standard library
- Custom containers, iterators, and algorithms
- Expanding the C++ memory model
- Compile-time optimization
- Multiple inheritance

Course Materials and Textbooks

The course materials and textbooks listed here can be acquired through the UCSD bookstore.

Required Texts

- C/C++ IV Lecture Notes, Raymond Mitchell III, ISBN 9780100729872

Recommended Texts

- Exceptional C++: 47 Engineering Puzzles, Programming Problems, and Solutions, 1st Edition, Sutter, ISBN 9780201615623

Course Prerequisites

CSE-40477 C/C++ Programming III or equivalent knowledge and experience.

Student Learning Outcomes

By the end of this course, students will be able to:

- Write safe C++03-compliant programs capable of running in the real world
- Identify and resolve safety issues in existing code
- Think defensively when writing code
- How and when to apply multi-inheritance to a solution
- Have a solid foundation for learning the features introduced by newer versions of C++, and understand why such features were added
- Write software tests to verify the behavior of a program

Course Schedule

Session	Topic & Reading	Assignments w/due dates
1	Generic programming <ul style="list-style-type: none"> Mitchell section 1 Sutter items 1-7 (ch. 1) 	Homework #1 (due 1/19/2021 11:59pm)
2	Exception safety <ul style="list-style-type: none"> Mitchell section 2 Sutter items 8-19 (ch. 2) 	Homework #2 (due 1/26/2021 11:59pm)
3	Class design <ul style="list-style-type: none"> Mitchell section 3 Sutter items 20-25 (ch. 3) 	Homework #3 (due 2/2/2021 11:59pm) Project #1 (due 2/16/2021 11:59pm)
4	Compilers, names, interfaces <ul style="list-style-type: none"> Mitchell section 4 Sutter items 26-34 (ch. 4 & 5) 	Homework #4 (due 2/9/2021 11:59pm)
5	Memory management <ul style="list-style-type: none"> Mitchell section 5 Sutter items 35-37 (ch. 6) 	Homework #5 (due 2/16/2021 11:59pm)
6	Traps, pitfalls, anti-idioms <ul style="list-style-type: none"> Mitchell section 6 Sutter items 38-41 (ch. 7) 	Homework #6 (due 2/23/2021 11:59pm) Project #2 (due 3/9/2021 11:59pm)
7	Functors & iterators <ul style="list-style-type: none"> Mitchell section 7 Sutter (none) 	Homework #7 (due 3/2/2021 11:59pm)
8	Miscellaneous topics <ul style="list-style-type: none"> Mitchell section 8 Sutter items 42-46 (ch. 8) 	Homework #8 (due 3/9/2021 11:59pm)
9	Control flow, multiple inheritance <ul style="list-style-type: none"> Mitchell section 9 Sutter (none) 	None

Grading and Assignment Information

Grading Scale

Your final course grade is based on the percentage of points you have earned.

<i>Grades</i>	
A	85-100%
B	65-84%
C	50-64%
F	<50%

Weighted Grading Criteria

Your course grade is determined by the grades you receive on the homework and projects. The course grade is calculated as follows:

- Homework – 1/2 of course grade
- Project #1 – 1/4 of course grade
- Project #2 – 1/4 of course grade

All homework assignments are weighted equally, and you may drop the lowest grade. This means your homework grade is determined by taking the average of your top 7 homework assignment grades.

Grading Policies

This course can be taken as part of the C/C++ Programming certificate. In order for the class to count towards your certificate it must be taken for a letter grade or as pass/no pass. Classes that are taken as NFC cannot count towards a certificate. You can change your grading option any time BEFORE the last day of class through [My Extension](#).

Late Policy

Assignments are due on the dates listed in the Course Schedule and are submitted via canvas. Late homework will be accepted **only on the day after the due date (by midnight)** for only 50% max credit, unless e-mail confirmation from the instructor prior to the start of the class allows for another arrangement.

Expect and plan for contingencies and technical problems (they WILL happen!). Remember that you can drop the lowest assignment so plan accordingly.

Assignments

Assignments are submitted via canvas.

For **homework** you should submit a single PDF file. This file should contain your source code and a screenshot of your program's output.

The PDF file you submit should be named as follows:

- <your name>HW<homework #>.pdf (e.g. "RayMitchellHW1.pdf")

For **projects** you should submit a single zip file. This file should contain your source code files. Only the source code files you write should be included. Each project will indicate which files you should include.

The zip file you submit should be named as follows:

- <your name>PR<project #>.zip (e.g. "RayMitchellPR1.zip")

Assignment Feedback

Assignment grades and solutions will be posted within 1 week of the assignment due date.