



School of Engineering
*Information
Technology Program*

ITP 435 – Professional C++

Units: 4

Spring 2021

Location: Online (it is unlikely we will have any in-class meetings)

Instructors: Sanjay Madhav: 2:00-3:50pm Mon/Wed

Arash Saifhashemi: 4:00-5:50pm Mon/Wed

Office: Online

Office Hours: TBD

Contact Info: All general course/assignments questions should be asked on Piazza (every student will receive an invitation at the start of the semester).

Personal questions and questions from prospective students should be directed via email to madhav@usc.edu or saifhash@usc.edu.

Teaching Assistants: TBD

Office: TBD

Office Hours: TBD

Contact Info: Via Piazza.

Course Description

This course teaches students how to use C++ as a professional developer in industry would. We will explore several different areas and applications where C++ sees significant use. We will learn applications of advanced concepts including lambda expressions, templates, secure coding, parallel programming, writing performant code, CMake, and continuous integration.

Learning Objectives

- Write C++ code for programming assignments in several different real-life applications
- Refine student's ability to design and write high-quality C++ code
- Learn how to improve the performance of C++ code
- Learn how to apply new programming paradigms (such as functional and data-oriented programming)
- Learn new ways to apply previously-known C++ language constructs
- Learn how the C++ language has evolved in the C++11, 14, and 17 standards, and how to utilize these new features

Prerequisite(s): CSCI 104L *or* ITP 365

Course Notes

Lecture slides and assignments will all be posted on Blackboard. Course discussions will occur on Piazza. Assignments will be submitted through GitHub.

Technological Proficiency and Hardware/Software Required

Students should have access to their own computer running either Windows, MacOS, or Linux, and should be familiar with the basic operation of their computer.

Required Readings and Supplementary Materials

Bancila, Marius. *Modern C++ Programming Cookbook*. Packt. 2017. ISBN-10: 1786465183.

Meyers, Scott. *Effective C++ (Third Edition)*. Addison-Wesley. 2005. ISBN-10: 0321334876.

Meyers, Scott. *Effective Modern C++*. O'Reilly Media. 2015. ISBN-10: 1491903996.

Additional readings such as excerpts from other books or online articles will be provided on Blackboard.

Description and Assessment of Assignments

There are seven different programming assignments in this course, and students will have approximately two weeks per assignment. Each assignment combines specific C++ skills with an industrial application of C++. Students are expected to complete these programming assignments **individually**. Each assignment's instructions include a grading rubric for that assignment.

Here's an **example** of what the seven programming assignments have been in prior semesters. They may vary this semester:

1. RLE compression
2. Genetic Algorithms and Functional Programming
3. DNA global sequence alignment
4. Parallelization
5. Virtual Machine
6. Basic Compiler
7. REST server

Exams

There is a midterm and final exam. The final exam is cumulative.

Grading Breakdown

Item	% of Grade
Assignments (8% each)	56
Midterm Exam	22
Final Exam	22
Total	100

Grading Scale

Course final grades are determined by the following scale

A	93-100
A-	90-92
B+	87-89
B	83-86
B-	80-82
C+	77-79
C	73-76
C-	70-72
D+	69
D	67-69
D-	66
F	65 and below

Half percentage points will be rounded up to the next whole percentage. So for instance, 89.5% is an A-, but 89.4% is a B+.

Depending on the overall class average at the end of the semester, the above grading scale may be relaxed. Extra credit is generally not offered.

Assignment Submission Policy

Programming assignments must be submitted to student's GitHub repositories by 11:59PM of the deadline date or will be considered late. Programming assignments that do not compile on the Travis CI continuous integration system will receive a 0. Information about Travis CI is provided in the first week of class.

Grading Timeline

Students will receive grades on programming assignments within one week after the due date.

Late Policy

Programming assignments will be accepted up to three days late, with a 15% deduction per day late. This means an assignment late by one day can receive a grade no higher than 85%, two days no higher than 70%, and three days no higher than 45%.

Extensions are only provided in the event of a *documented* reason satisfactory to the instructor, such as an illness or family emergency.

Make-up Policy for Exams

To make up for a missed exam, the student must provide a satisfactory reason (as determined by the instructor) along with documentation. Make-up exams are only allowed under extraordinary circumstances.

Grading Issues

Students will have two weeks after graded feedback is given to contest scores (e.g. assignments and exams). After two weeks, scores will not be changed.

Plagiarism and Individual Work Policy

In this class, programming assignments are expected to represent the individual effort of each student. All programming assignment submissions will be compared with current, previous, and future students' submissions using MOSS, which is a code plagiarism identification program. If your code significantly matches another student's submission, you will be referred to SJACS with a recommended penalty of an F in the course.

It is okay to discuss solutions to specific problems with other students, but it is not okay to look through another student's code. It does not matter if this code is online or from a student you know, it is cheating. Do not share your code with anyone else in this or a future section of the course, as allowing someone else to copy your code carries the same penalty as copying the code yourself.

Course Material Policy

Do not reproduce, distribute, or post any lecture material, assignments, assignment solutions, or exams publicly without written consent of the instructor. You may take notes and make copies of course materials for your own use. You may not post course materials on sites like CourseHero. Doing so is a copyright violation and in some cases may also be an academic integrity violation that will be dealt with accordingly

Course Schedule

(This is a rough outline of the topics we plan to cover. A handful of topics especially towards the end of the semester may change between now and the start of the semester).

Date	Lecture Topics	Readings	Due Dates
1/18	MLK Day (No class)		
1/20	Introduction; Tools and Testing	Mancila: pp. 1-15, 38-45	
1/25	Modernizing Your C++ Code	<i>Effective</i> : 1-4, 7; 9-12; 20, 27	
1/27	Lambdas and C++ Functional Programming	<i>Modern</i> : 2, 5, 6; Mancila: pp. 124-136, 149-161	
2/1	Applications: Genetic Algorithms	"Genetic Algorithms" from <i>AI/AMA</i>	
2/3	Sizeof and Virtual Tables	<i>Effective</i> : 5, 6, 26, 30	PA1 2/3 @ 11:59PM
2/8	Writing Optimized Code	<i>Effective</i> : 30-31	
2/10	Move Semantics	<i>Modern</i> : 23-26	
2/15	President's Day (No class)		
2/17	Parallel Programming in C++	Mancila: pp. 354-370	PA2 2/17 @ 11:59PM
2/22	Multithreading in C++		
2/24	C++ Smart Pointers	<i>Modern</i> : 18-21	
3/1	Applications: Bioinformatics	"Beginners Guide to DNA Seq..."	
3/3	Midterm Review		PA3 3/3 @ 11:59PM
3/8	Midterm Exam - The midterm exam will be online, we will give details in class		
3/10	Exceptions and RTTI	Mancila: pp. 413-429	
3/15	Design Patterns	"Introduction" from <i>Design Patterns</i>	
3/17	Template Metaprogramming Basics	Mancila: pp. 175-189	PA4 3/17 @ 11:59PM
3/22	More Template Metaprogramming		
3/24	Compiler Basics: Lexical Analysis		
3/29	Compiler Basics: Syntax Analysis		
3/31	Compiler Basics: Code Gen		PA5 3/31 @ 11:59PM
4/5	Uniform Initializers and Related	<i>Modern</i> : 7; Mancila: pp. 15-21	
4/7	Wellness Day (No class)		
4/12	CMake and Bazel		
4/14	Google Benchmark, Test, and Mock		PA6 4/16 @ 11:59PM
4/19	Custom Memory Allocators	<i>Effective</i> : 49-52	
4/21	C++ 14/17/20 and Beyond		
4/27	Selected Talks from Cppcon		
4/28	Final Exam Review		PA7 4/29 @ 11:59PM
FINAL	Final Exam – The final exam will be online, we will give details in class		

"Effective" refers to Effective C++ while "Modern" refers to Effective Modern C++. For these texts, the numbers are not page numbers, but rather the item numbers referenced in the table of contents.

Assigned readings in quotes will be available either directly or via link on Blackboard.

Statement on Academic Conduct and Support Systems

Academic Conduct

Plagiarism – presenting someone else’s ideas as your own, either verbatim or recast in your own words – is a serious academic offense with serious consequences. Please familiarize yourself with the discussion of plagiarism in *SCampus* in Part B, Section 11, “Behavior Violating University Standards” <https://policy.usc.edu/scampus-part-b/>. Other forms of academic dishonesty are equally unacceptable. See additional information in *SCampus* and university policies on scientific misconduct, <http://policy.usc.edu/scientific-misconduct>.

Support Systems

Student Counseling Services (SCS) - (213) 740-7711 – 24/7 on call

Free and confidential mental health treatment for students, including short-term psychotherapy, group counseling, stress fitness workshops, and crisis intervention. <https://engemannshc.usc.edu/counseling/>

National Suicide Prevention Lifeline - 1-800-273-8255

Provides free and confidential emotional support to people in suicidal crisis or emotional distress 24 hours a day, 7 days a week. <http://www.suicidepreventionlifeline.org>

Relationship and Sexual Violence Prevention Services (RSVP) - (213) 740-4900 - 24/7 on call

Free and confidential therapy services, workshops, and training for situations related to gender-based harm. <https://engemannshc.usc.edu/rsvp/>

Sexual Assault Resource Center

For more information about how to get help or help a survivor, rights, reporting options, and additional resources, visit the website: <http://sarc.usc.edu/>

Office of Equity and Diversity (OED)/Title IX Compliance – (213) 740-5086

Works with faculty, staff, visitors, applicants, and students around issues of protected class. <https://equity.usc.edu/>

Bias Assessment Response and Support

Incidents of bias, hate crimes and microaggressions need to be reported allowing for appropriate investigation and response. <https://studentaffairs.usc.edu/bias-assessment-response-support/>

The Office of Disability Services and Programs

Provides certification for students with disabilities and helps arrange relevant accommodations. <http://dsp.usc.edu>

Student Support and Advocacy – (213) 821-4710

Assists students and families in resolving complex issues adversely affecting their success as a student EX: personal, financial, and academic. <https://studentaffairs.usc.edu/ssa/>

Diversity at USC

Information on events, programs and training, the Diversity Task Force (including representatives for each school), chronology, participation, and various resources for students. <https://diversity.usc.edu/>

USC Emergency Information

Provides safety and other updates, including ways in which instruction will be continued if an officially declared emergency makes travel to campus infeasible, <http://emergency.usc.edu>

USC Department of Public Safety – 213-740-4321 (UPC) and 323-442-1000 (HSC) for 24-hour emergency assistance or to report a crime.

Provides overall safety to USC community. <http://dps.usc.edu>