USC Viterbi School of Engineering

Professional C++

ITP 435 (3 Units)
Fall 2018

Objective	This course provides students with the advanced knowledge they will need to succeed as a professional C++ developer. By semester's end, students will: 1. Become familiar with advanced C++ language idioms. 2. Gain exposure to common libraries used professionally today. 3. Understand how to write efficient and high-quality C++ code.
Concepts	Code Generation. Memory layout. Templates. STL. Optimization. Exceptions. RTTI. Design Patterns. Metaprogramming. Lambda Expressions. Boost. Custom Memory Allocators. C++11. Compilers.
Prerequisites	CSCI 104 or ITP 365x
Instructor	Sanjay Madhav
Contact	Students in the course should post their questions on Piazza. *Email:* madhav@usc.edu (Only for non-course questions or prospective students).
Office Hours	Monday/Wednesday 4:00-6:00PM in OHE 530H
Lecture	Section 32012: Monday and Wednesday, 2:00 – 3:20PM in KAP 160 Section 32042: Tuesday and Thursday, 2:00 – 3:20PM in KAP 160
Course Structure	The topics covered during class meetings will be applied to the seven programming assignments spread out through the semester. All programming assignments must be completed *individually*. There are two exams that are comprehensive of all topics covered.
Textbooks	**Required**: *Effective C++ (Third Edition)*. Scott Meyers. ISBN-10: 0321334876. **Recommended:** *Effective Modern C++*. Scott Meyers. ISBN-10: 1491903996.
Grading	The course is graded with the following weights: Programming Assignments (7% each) 49% Midterm Exam 21% Final Exam 25% Class Participation 5% TOTAL POSSIBLE 100%

Grading Scale

Letter grades will be assigned according to the following scale:

93%+	A
90-92%	A-
87-89%	B+
83-86%	B
80-82%	B-
77-79%	C+
73-76%	C
70-72%	C-
69	D+
67-68	D
66	D-
65 and below	F

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.

Policies

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 proper documentation. Make-up exams are only allowed under extraordinary circumstances.

Late Assignments: Late assignments will be accepted one day late for a 15% penalty, two days late for a 30% penalty, and three days late for a 45% penalty. An assignment submitted later than this will be given a grade of 0, unless there is an extraordinary and documented reason as to why it was late.

Students will be able to setup their own PC or Mac for use in the class, as all software is free either in general or specifically for students enrolled in Viterbi courses. All projects natively build on both PC and Mac, assuming the appropriate software is installed. Linux should work as well, but no technical support will be provided for students who wish to use Linux.

Alternatively, ITP offers Open Lab use for all students enrolled in ITP classes. These open labs are held beginning the second week of classes through the last week of classes. Please contact your instructor for specific times and days for the current semester.

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 Section 11, *Behavior Violating University Standards* https://scampus.usc.edu/1100-behavior-violating-university-standards-and-appropriate-sanctions/. 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/. Discrimination, sexual assault, and harassment are not tolerated by the university. You are encouraged to report any incidents to the *Office of Equity and Diversity* http://equity.usc.edu/ or to the *Department of Public Safety* http://capsnet.usc.edu/department/department-public-safety/online-forms/contact-us. This is important for the safety whole USC community. Another member of the university community – such as a friend, classmate, advisor, or faculty member – can help initiate the report, or can initiate the report on behalf of another person. *The Center for Women and Men* http://www.usc.edu/student-affairs/cwm/ provides 24/7 confidential support, and the sexual assault resource center webpage sarc.usc.edu describes reporting options and other resources. **Support Systems** A number of USC's schools provide support for students who need help with scholarly writing. Check with your advisor or program staff to find out more. Students whose primary language is not English should check with the *American Language Institute* http://dornsife.usc.edu/ali, which sponsors courses and workshops specifically for international graduate students. *The Office of Disability Services and Programs* http://sait.usc.edu/academicsupport/centerprograms/dsp/home_index.html provides certification for students with disabilities and helps arrange the relevant accommodations. If an officially declared emergency makes travel to campus infeasible, *USC Emergency Information http://emergency.usc.edu/* will provide safety and other updates, including ways in which instruction will be continued by means of blackboard, teleconferencing, and other technology.
A Further Note on Plagiarism	In this class, 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 reported to SJACS with the 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 you copying the code yourself.

Course Outline (Order of topics may be rearranged during the semester)		
Week	**Topic(s)**	**Reading/PA**
1	Introduction; Tools and Testing	
	Modernizing Your C++ Code	*Effective*: Intro & #1-4; 7; 9-12; 20, 27
2	Lambdas/Functional Programming	*Modern*: #2, 5, 6
	Genetic Algorithms	
3	***No class (Labor Day)***	**PA1 Due 9/5 @ 11:59PM**
	Sizeof; Virtual tables	*Effective*: #5, 6, 26, 30;
4	Is-a vs. Has-a; Preprocesser	*Effective: #32-40;*
	Bioinformatics and Dynamic Programming	
5	Writing Optimized and Secure Code	*Effective*: #30-31;
	Design Patterns	**PA2 Due 9/19 @ 11:59PM**
6	Smart Pointers	*Modern*: #18-21
	Basic Parallel Programming; Intel TBB	
7	Move Semantics	*Modern*: #23-26;
	Guest Lecture	**PA3 Due 10/3 @ 11:59PM**
8	Midterm Review	
	Midterm exam	
9	Exceptions and RTTI	
	Template Metaprogramming	**PA4 Due 10/17 @ 11:59PM**
10	Custom Memory Allocators	*Effective: #49-52*
	Uniform Initializers; Initializer Lists	*Modern*: #7
11	Secure Design, Development, and Test	
	Intro. to Compilers – Basics; Lexical Analysis	**PA5 Due 10/31 @ 11:59PM**
12	Intro. to Compilers – Syntax Analysis	
	Intro. to Compilers – Code Generation	
13	Boost Library; C++11 Concurrency; Testing	*Modern*: #25; *Effective*: #55;
	C++14, 17, and 20	**PA6 Due 11/16 @ 11:59PM**
14	Deep C	
	Thanksgiving Holiday (no class)	
15	Selected Talks from Cppcon2017	
	Final Review	**PA7 Due 11/30 @ 11:59PM**
	Final Exam according to final exam schedule For the M/W section – Friday, December 7 from 2 – 4PM For the T/Th section – Thursday, December 6 from 2 – 4PM	

Note that “Effective” refers to *Effective C++* while “Modern” refers to *Effective Modern C++*.