San José State University
Department of Computer Science
CS 144 Advanced C++ Programming

Instructor: Soon Tee Teoh
Email: *soontee.teoh@sjsu.edu*
Office Hours: By appointment only. I am typically available Mon/Wed/Fri 5 pm till midnight. Email me 24 hours in advance to schedule a meeting.
Office Location: https://sjsu.zoom.us/j/132407857

Class Days/Time: Tue/Thu 6:00 - 7:15 pm
Classroom: https://sjsu.zoom.us/j/132407857
Prerequisites: CS 46B and CS 49C (with a grade of C- or better in each), or equivalent knowledge of object-oriented programming and C, or instructor consent.

Course Format
This course will be taught primarily via classroom presentations. All lectures will be recorded and posted on the "Pages" tab on Canvas. Students are expected to be present during the scheduled exams and any homework demonstrations during the class period.

Canvas Course Website
Course materials, syllabus, assignments, grading criteria, exams, and other information will be posted on the Canvas Learning Management System course login website at http://sjsu.instructure.com.
You are responsible for regularly checking these websites to learn of any updates. You can find Canvas video tutorials and documentations at http://ges.sjsu.edu/canvas-students.

Course Catalog Description
Advanced features of C++, including operator overloading, memory management, templates, exceptions, multiple inheritance, RTTI, namespaces, tools.

Course Learning Outcomes (CLO)
Upon successful completion of this course, you will be able to:
CLO 1: Apply object-oriented features of C++, including polymorphism and recursion.
CLO 2: Apply advanced features of C++, including operator overloading, memory management, templates, the Standard Template Library (STL), exceptions, multiple inheritance, runtime type identification (RTTI), namespaces, etc.
CLO 3: Apply modern features of C++, including lambda expressions, smart pointers, move semantics, etc.
CLO 4: Develop interactive GUI-based applications in C++ that use inversion of control and callback functions as event handlers.

CLO 5: Understand the concepts of multithreaded programming.

CLO 6: Use high-level software development tools, including an integrated development environment (IDE), compilers, linkers, and source-level debuggers to implement and debug C++ applications.

CLO 7: Write efficient programs in C++ that adhere to good design principles while avoiding pitfalls of the language.

CLO 8: Document program design with Unified Modeling Language (UML) diagrams.

Academic Integrity

You may study together and discuss the assignments, but what you turn in must be your individual work. Copying code from another student's program or sharing your program code are equally serious violations of academic integrity. Never use code you find on the web, unless you have the instructor's permission, and then you must give proper attribution in your comments. This is similar to giving attribution to a quote that you use in a term paper. Assignment submissions will be checked for plagiarism using Moss from the Department of Computer Science at Stanford University. See http://theory.stanford.edu/~aiken/moss/.

Violators of academic integrity will suffer severe sanctions, including academic probation. Students who are on academic probation are not eligible for work as instructional assistants in the university or for internships at local companies.

Recommended Texts

C++ How to Program, 10th edition, Paul J. Deitel and Harvey Deitel, Pearson International, 2017, ISBN:978-9332585737

Effective Modern C++ 42 Specific Ways to Improve Your Use of C++11 and C++ 14, Scott Meyers, O'Reilly Media, 2014, ISBN:978-1491903995

The C++ Standard Library A Tutorial and Reference, 2nd edition, Nicolai M. Josuttis Addison-Wesley Professional, ISBN:2012 978-0321623218

Software to Install

Use any IDE (Integrated Development Environment). Examples for Windows include Eclipse or CodeBlocks. For the initial weeks, just a C++ compiler will suffice, such as Cygwin64 for Windows.

Course requirements and assignments

There will be homework assignments and exams.

<u>Assignments</u>

There will be multiple (about 10) programming assignments throughout the semester. Each assignment will be worth a specified maximum number of points. Assignments can be turned in within 48 hours late for 20% deduction. After 48 hours, no submission is allowed (it will get a 0 score).

<u>Exams</u>

All exams are open-book. You can refer to all class material. However, you may not communicate with any other person, or search solutions on the Internet. The exams will test

understanding (not memorization) of the material taught during the semester. Instant messaging, e-mails, texting, tweeting, file sharing, or any other forms of communication with anyone else during the exams will be strictly forbidden. There can be no make-up exams unless there is a documented medical emergency.

The university's syllabus policies:
• University Syllabus Policy S16-9 at http://www.sjsu.edu/senate/docs/S16-9.pdf
• Office of Graduate and Undergraduate Programs' Syllabus Information web page at http://www.sjsu.edu/gup/syllabusinfo/.
"Success in this course is based on the expectation that students will spend, for each unit of credit, a minimum of 45 hours over the length of the course (normally 3 hours per unit per week with 1 of the hours used for lecture) for instruction or preparation/studying or course related activities including but not limited to internships, labs, clinical practica. Other course structures will have equivalent workload expectations as described in the syllabus."

Grading Information
Your final class grade will be weighted as follows:
Programming Assignments: 40%
4 Exams: 15% each

Final score to letter grade conversion is as follows:

Overall Score	Grade
100 - 96	A
95.99 - 92	A-
91.99 - 88	B+
87.99 - 84	B
83.99 - 80	B-
79.99 - 77	C+
76.99 - 74	C
73.99 - 70	C-
69.99 - 0	F

Classroom Protocol
It is very important for each student to attend classes and to participate. Cell phones in silent mode, please. No use of electronic devices (phones, tablets, laptop computers etc.) in class, except to take notes.

University Policies
Per University Policy S16-9, university-wide policy information relevant to all courses, such as academic integrity, accommodations, etc. will be available on Office of Graduate and

Undergraduate Programs' Syllabus Information web page at
http://www.sjsu.edu/gup/syllabusinfo/.

Course Schedule

Week	Tue	Thu
0		1/27/2022
		I/O, classes, private, public, pass by reference
1	2/1/2022	2/3/2022
	function and operator overloading, const	static arrays
2	2/8/2022	2/10/2022
	pointers, dynamic arrays	friend, copy constructor
3	2/15/2022	2/17/2022
	Review	Exam 1
4	2/22/2022	2/24/2022
	linked list	template
5	3/1/2022	3/3/2022
	vector, map	inheritance
6	3/8/2022	3/10/2022
	Review	Exam 2
7	3/15/2022	3/17/2022
	file i/o	polymorphism
8	3/22/2022	3/24/2022
	multiple inheritance	static
9	3/29/2022	3/31/2022

		Spring Break	Spring Break
10	4/5/2022		4/7/2022
		run-time type information (RTTI)	Homework Demo
11	4/12/2022		4/14/2022
		IDE, GUI callback	Review
12	4/19/2022		4/21/2022
		Exam 3	exception handling
13	4/26/2022		4/28/2022
		multi-threaded programming	lambda expressions
14	5/3/2022		5/5/2022
		smart pointers	recursion, namespace
15	5/10/2022		5/12/2022
		Review	Exam 4