San José State University
Computer Science Department
Advanced C++ Programming (CS 144, Section 01)
Fall 2021

COURSE AND CONTACT INFORMATION

Instructor:	Dominic Abucejo
Office Location:	Online Zoom meeting
Telephone:	N/A
Email:	dominic.abucejo@sjsu.edu
Office Hours:	Tuesday/Thursday, email for an appointment slot (from 6:00pm to 7pm PST) Zoom Details: • https://sjsu.zoom.us/j/89234181745?pwd=T3ZYeE43aGFpeVdJTlN2UmJnbVZvUT09 • Password: F80cUbQC
Class Days/Time:	Mon/Wed 7:30 pm – 8:45pm PST (Pacific Standard Time)
Classroom:	Online Synchronous Zoom meetings on Canvas
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 DESCRIPTION

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

COURSE FORMAT

CS 144 is an online hands-on programming course. Classes will be taught online via Zoom Video Call Conferencing. In class programming practice, quizzes, surveys, as well as topic presentation will be provided (not in any specific order). Assignments, quizzes, projects, and exams will be provided via Canvas. All classes will be recorded and recordings will be made available for reference. Student participation during class is key to the successful completion of the course.

Course materials such as syllabus, handouts, notes, assignment instructions, reading assignments, etc. are posted on the Canvas Website at http://sjsu.instructure.com. Students are responsible for regularly checking for updates on canvas.

C.L.O. (Course Learning Outcomes)

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.

REQUIRED TEXTS/READINGS

Textbook

✦ Title: C++ How to Program, 10th edition
Author: Paul Deitel, Harvey M. Deitel
Publisher: Pearson
ISBN-13: 9780134448961
@ Pearson (www.pearson.com) or Amazon

Other Readings

Additional course readings, code examples, exercises, etc. will be assigned and will be provided by the instructor during the course.

Other technology requirements / equipment / material

Wired or Wireless Laptop/Workstation (should be used for all classes). Students are responsible for ensuring that they have access to reliable Wi-Fi during tests. If students are unable to have reliable Wi-Fi, they must inform the instructor, as soon as possible or at the latest one week before the test date to determine an alternative.

COURSE REQUIREMENTS AND ASSIGNMENTS

Course requirements, reading materials, hands-on coding activities, and assignments contribute to and are aligned with course learning outcomes. 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 three hours per unit per week) for instruction, preparation/studying, or course-related activities, including but not limited to reading and assignments.

The final grade is calculated based on the percentage of the total points for all the Course Requirement and Assignments listed below:

- Homework Assignments: individual, regularly assigned, will include written problem assignments, and perhaps some online exercises. Solutions will be not posted. Assignments are submitted via Canvas for grading. Students must submit only their own work by the posted due date. (No late assignment submissions)
- Reading assignments: Reading assignments will regularly be for the next class.
- Quizzes: At least 2 quizzes per week will be issued via Canvas. Quizzes will be 5 to 10 minutes in total duration with one to three questions.
- Midterm: There will be one written Midterm exam during the semester. Makeup exams will only be given in cases of illness (with signed documentation from a medical facility – original copy). Exams are closed book, closed notes and closed communications.
- Project: There will be a programming project. Information on the project, including topics and deadlines, will be given later in the course.
- Final exam: The final has a fixed date. Makeup exams will only be given in cases of illness (with signed documentation from a medical facility – original copy). Exams are closed book, closed notes and closed communications. The final exam is cumulative.

THE UNIVERSITY'S SYLLABUS POLICIES:

- University Syllabus Policy S16-9 at http://www.sjsu.edu/senate/docs/S16-9.pdf.
- Student Expectations, Academic Integrity description at the Office of Undergraduate Education web page: http://www.sjsu.edu/gup/syllabusinfo/

Final Examination or Evaluation

The exam will test understanding (not mainly memorization) of the material taught during the semester. The final exam is cumulative and closed book. The examination is expected to have problem analysis and problem-solving style questions to answer.

GRADING INFORMATION

The final grade is calculated based on the percentage of the total points for all the Course Requirement and Assignments listed below:

Homework Assignments	40%
Quizzes	10%
Midterm	15%

Project	15%
Final Exam	20%

Grade	Percentage
A +	97.50 to 100%
A	92.50 to 97.49%
A -	90.00 to 92.49%
B +	87.50 to 89.99 %
B	82.50 to 87.49%
B -	80.00 to 82.49%
C +	77.50 to 79.99%
C	72.50 to 77.49%
C -	70.00 to 72.49%
D +	67.50 to 69.99%
D	62.50 to 67.49%
D -	60.00 to 62.49%
F	Below 60.00%

RECORDING OF ZOOM CLASSES

- This course or portions of this course (i.e., lectures, discussions, student presentations) will be recorded for instructional or educational purposes. The recordings will only be shared with students enrolled in the class through Canvas. The recordings will be deleted at the end of the semester.

- If, however, you would prefer to remain anonymous during these recordings, then please speak with the instructor about possible accommodations (e.g., temporarily turning off identifying information from the Zoom session, including student name and picture, prior to recording).

- Students are not allowed to record without instructor permission Students are prohibited from recording class activities (including class lectures, office hours, advising sessions, etc.), distributing class recordings, or posting class recordings. Materials created by the instructor for the course (syllabi, lectures and lecture notes, presentations, etc.) are copyrighted by the instructor. This university policy (S12-7) is in place to protect the privacy of students in the course, as well as to maintain academic integrity through reducing the instances of cheating. Students who record, distribute, or post these materials will be referred to the Student Conduct and Ethical Development office. Unauthorized recording may violate university and state law. It is the responsibility of students that require special accommodations or assistive technology due to a disability to notify the instructor.

- Any student that needs accommodations or assistive technology due to a disability should work with the Accessible Education Center (AEC), and the instructor.

UNIVERSITY POLICIES

Per University Policy S16-9 (http://www.sjsu.edu/senate/docs/S16-9.pdf), relevant university policy concerning all courses, such as student responsibilities, academic integrity, accommodations, dropping and adding, consent for recording of class, etc. and available student services (e.g. learning assistance, counseling, and other resources) are listed on Syllabus Information web page (http://www.sjsu.edu/gup/syllabusinfo), which is hosted by the Office of Undergraduate Education. Make sure to visit this page to review and be aware of these university policies and resources.

NOTE: Any form of cheating is prohibited during the duration of this course. Consequences will range from an university academic integrity reporting of the participating student(s) to the university and/or the participating student(s) will receive an automatic failure for the assignment/test/exam/quiz/project and up to an automatic failure (overall/final grade) for the course.

FALL 2021 (TENTATIVE COURSE TOPIC SCHEDULE)

This schedule is subject to change with fair notice. Updates will be given in class and the schedule available on the course website will be updated.

Course Tentative Schedule

Week	Date	Topics, Readings, Assignments, Deadlines
1	August 23, 2021	*Course Information, C++ basics*
1	August 25, 2021	*Flow of control, Simple input and output (I/O)*
2	August 30, 2021	*Procedural abstraction*
2	September 1, 2021	*Functions*

3	***September 6, 2021***	***Holiday (Labor Day)***
3	September 8, 2021	*I/O streams, Introduction to classes and objects*
4	September 13, 2021	*Arrays, Strings, Vectors*
4	September 15, 2021	*Pointers*
5	September 20, 2021	*Dynamic arrays*
5	September 22, 2021	*Structures, Classes, Public and private members, Constructors and destructors*
6	September 27, 2021	*Friend functions, Abstract data types (ADT)*
6	September 29, 2021	*Analysis precedes design, Where do classes come from?, UML class and sequence diagrams*
7	October 4, 2021	*Separate compilation, Namespaces, Inlining*
7	October 6, 2021	*Class hierarchies, Inheritance*
8	October 11, 2021	*Overriding and overloading functions, Operator overloading*
8	October 13, 2021	Review
9	**October 18, 2021**	***Midterm***
9	October 20, 2021	*Cohesion and consistency, The Liskov Substitution Principle, The Law of Demeter and the Principle of Least Knowledge, The Open-Closed Principle*
10	October 25, 2021	*A class design example, Accessors and mutators, Immutable classes*
10	October 27, 2021	*Copy constructors, The assignment operator, The “Big Three”, A “safe” array type*
11	November 1, 2021	*Linked lists, Stacks, Queues*
11	November 3, 2021	*Polymorphism, Virtual destructors, Abstract classes and interfaces, Multiple inheritance*
12	November 8, 2021	*Runtime type identification (RTTI), The Principle of Coding to the Interface, The Principle of Favoring Delegation over Inheritance*
12	November 10, 2021	*Recursion, Binary search, Mergesort*
13	November 15, 2021	*Exception handling, Template functions and classes*
13	November 17, 2021	*Standard Template Library (STL), STL Containers, STL vectors, STL linked lists, STL iterators, STL sorting*

14	November 22, 2021	*The Model-View-Controller architecture, Interactive programming with a graphical user interface (GUI), Introduction to wxWidgets, Inversion of control*
14	***November 24, 2021***	***No Class (Non Instruction Day)***
15	November 29, 2021	*Callback functions, Events and event handlers, Lambda expressions, The auto keyword, The decltype pseudo-function*
15	December 1, 2021	*Constructor and destructor calls, How does an STL vector grow?, Why did my program crash?, Shallow vs. deep copy, Pointers vs. references, Raw pointers vs. unique and shared smart pointers, Move semantics*
16	December 6, 2021	*Introduction to multi-threaded programming, Critical regions, mutexes, and semaphores,* Review
Final Exam	**December 8, 2021 (Wednesday)**	**19:45-10:00 PST (Pacific Standard Time)**