CSCE 240: Advanced Programming Techniques

Instructor: Brian Hipp
Email: bhipp@cse.sc.edu
Office: Storey Innovation Center 2245
Phone: 803.777.8103
Office Hours: MW 2pm-3pm, TR 1pm-2pm, and by appointment

Course Details
 Semester: Spring 2023
 Section: 002
 Meeting Time: TR 8:30am – 9:45am
 Meeting Location: Storey Innovation Center 1400
 Credit Hours: 3
 Bulletin Description: Pointers; memory management; advanced programming language structures: operator overloading, iterators, multiple inheritance, polymorphism, templates, virtual functions; Unix programming environment.
 Prerequisite(s): Grade of D or better in CSCE 215, grade of C or better in CSCE 146.

Learning Outcomes
 • Independently design and implement C++ programs in a Unix environment
 • Demonstrate mastery of pointers, iterators, memory management including object creation and destruction, and parameter passing in C++
 • Demonstrate mastery of object-oriented programming concepts including: inheritance, polymorphism, operator overloading, template functions and classes, and the use of STL containers
 • Develop object-oriented models using UML

Textbook
 Savitch, Walter (2016). *Absolute C++* (6th ed.). Pearson. ISBN: 9780133970982 (print) 9780133970838 (eText)

 All readings/materials comply with copyright/fair use policies.

Course Overview
 This course is synchronous and in-person. Attendance is mandatory as described in the Course Delivery section below.

 Students will complete readings from the course text, submit programming assignments (based on the reading assignments and lectures), and complete three projects/exams.

 Students should expect to receive feedback on program and project submissions within five business days of the submission deadline.

Technology and Software
 This course will extensively make use of blackboard.sc.edu. Students should check Blackboard regularly for class materials and assignments.

 All software required in the course is free for download. Students are expected to be familiar with the Unix operating system (familiarity is a prerequisite for this course via CSCE 215)

Topics Covered

1. Unix programming environment: Unix tools, C preprocessor, make, shell, I/O redirection, debugging
2. Unit testing
3. Pointers: pointer manipulation, functions and function pointers, virtual functions.
4. Basic class management: constructors, destructors, data hiding, container classes.
5. Memory management: object creation and destruction, memory leaks
6. Advanced C++ features: operator overloading, iteration, special containers, inheritance, code reuse, multiple inheritance, virtual functions, polymorphism, templates, template libraries
7. Introduction to UML and object oriented modeling: usecase models, object identification, specifying static behavior, activity diagrams, collaboration diagrams and sequence diagrams, specifying relationships: generalization/specialization, aggregation, associations including multiplicity and roles, dynamic behavior using state diagrams
8. Introduction to source control and distributed source control
9. Software development patterns

Course Activities

Projects/Exams: There are three projects in this course covering topics from

1. Chapters 1, 2, 3, 4 and 11.1 – *February 7th*
2. Chapters 5, 6, 7, 8, and 10 – *March 23rd*
3. Chapters 11.2, 14, 15, and 16 – *May 2nd*

Projects will be treated as exams and plagiarism will be not be tolerated. Projects are meant to test a student's current ability level. Any attempt to seek help with a project should be treated as an Academic Integrity violation.

Programming Assignments: There are six programming assignments in this course. Students are encouraged to ask the instructor for guidance. Program assignments must be completed by the student submitting the assignment without help from individuals other than the instructor. Programming assignments will be posted at least one week prior to their due date.

Program Expectations

- Programs must run correctly on one of the reference Linux machines in the SWGN 1D43 or SWGN 3D22 computer labs. Your program running on your personal computer is not considered valid. It must run on an official computer chosen by the "client" (the instructors). Note, these are the same machines used in the prerequisite course.
- Programs that do not compile will not receive execution credit.
- Programs that crash during execution will not receive execution credit.
- Programs that go into infinite loops during execution will receive no execution credit.
- Programs that fail to have your name and copyright information in the header documentation of ALL files (header and implementation code) will receive zero credit.
- Programs are expected to process any input file that meets the same format description as the sample data provided. For grading purposes, programs will be executed with different data with the same format.
- Programs must be styled correctly and documented. We will be following the Google Style guidelines found at https://google.github.io/styleguide/cppguide.html. You may download the Google's free cpplint Python3 application to ensure that you are correctly following the style guidelines.
- Submissions will be graded roughly as follows: 80% for correct execution, 20% for correct organization, style, and documentation.
- All submissions from all sections will be submitted to the MOSS website at Stanford for plagiarism detection purposes.

Grade Computation
Programming Assignments: 50%
Exam / Project 1: 15%
Exam / Project 2: 15%
Exam / Project 3: 20%

Grading Scale

A	90%-100%
B+	85%-89%
B	80%-84%
C+	75%-79%
C	70%-74%
D+	65%-69%
D	60%-64%
F	0%-59%

Student Disability Resource Center
The Student Disability Resource Center (SDRC) empowers students to manage challenges and limitations imposed by disabilities. Students with disabilities are encouraged to contact me to discuss the logistics of any accommodations needed to fulfill course requirements (within the first week of the semester). In order to receive reasonable accommodations from me, you must be registered with the Student Disability Resource Center (contact information is listed below). Any student with a documented disability should contact the SDRC to make arrangements for appropriate accommodations.

Office Location: 1705 College Street, Close-Hipp Suite 102, Columbia, SC 29208
Phone: 803.777.6142
Fax: 803.777.6741
Email: sadrc@mailbox.sc.edu
Web: https://sc.edu/about/offices_and_divisions/student_disability_resource_center/index.php

Academic Integrity
Students should visit the office of Student Conduct and Academic Integrity (https://www.sa.sc.edu/academicintegrity) to review the Code of Conduct and Honor Code. Academic misconduct will not be tolerated and will result in a failing grade on the assignment and/or in the course.

Course Delivery
This is an in-person synchronous course. Attendance is mandatory. However, students suffering from any contagious or unknown illness should not attend class, and should instead go to campus health and see a health professional. In addition to helping contain the spread of sickness, campus health can provide you with documentation for your absence or absences.

If you miss class, you are responsible for obtaining the notes from a classmate, and completing any required assignments in Blackboard.