CS G179: C++ PROGRAMMING, ADVANCED

Item	Value
Top Code	070710 - Computer Programming
Units	3 Total Units
Hours	90 Total Hours (Lecture Hours 36; Lab Hours 54)
Total Outside of Class Hours	0
Course Credit Status	Credit: Degree Applicable (D)
Material Fee	No
Basic Skills	Not Basic Skills (N)
Repeatable	No
Grading Policy	Standard Letter (S), • Pass/No Pass (B)

Course Description

This course covers advanced features of software development using the C++ language. Topics covered will include input/output streams, file input and output, exception handling, Standard Template Library (STL) including String class, sequential, and associative containers. Understanding function objects, STL algorithms, adaptive containers, bitset class, and smart pointers will also be discussed in lectures and practiced through lab projects. ADVISORY: CS G175. Transfer Credit: CSU; UC.

Course Level Student Learning Outcome(s)

1. Course Outcomes
2. Create data manipulation programs using STL containers.
3. Produce programs that implement principles of Object Oriented Programming (OOP) of inheritance and polymorphism.
4. Employ error isolation using both built-in and custom exception classes

Course Objectives

- 1. Discuss advanced Object Oriented Programming concepts in C++.
- 2. Create and use modules including containers from the Standard Template Library(STL).
- 3. Discuss and use iterators and algorithms to find, remove, reverse, and edit elements in containers.
- 4. Create and implement a method to work with bit flags to organize and manipulate bitwise information.
- 5. Implement exception handling.
- 6. Use software development methodologies and debugging techniques.

Lecture Content

A. Input / Output Streams.B. File Input / Output.C. Exception handling.D. Intro to the Standard Template Library (STL).E. Iterators and Algorithms.F. STL String Class.G. STL Dynamic Array classes (vector and deque classes).H. Doubly linked list (STL list class).I. Efficient searches using STL associative containers (set, multiset, map and multimap).J. Supplying Custom Sort Predicates (unary and binary).K. Concept and usage function objects.L. Adaptive Containers (STL stack, queue and priority_queue classes).M. Organize and manipulate bitwise information using STL bitset and vector classes.N. Smart Pointers.

Lab Content

A. Weekly assignments will be given such as: given a set of requirements for a small business or scientific problem, need code using the various classes being studied in their solution.B. Create a project with the correct file structure.C. Break the programs into appropriate classes.D. Design a simple user interface to satisfy the user interactions.E. Code all the necessary expressions, branches, loops, functions, and classes.F. Add the appropriate error handling routines.

Method(s) of Instruction

- Lecture (02)
- DE Live Online Lecture (02S)
- DE Online Lecture (02X)
- Lab (04)
- DE Live Online Lab (04S)
- DE Online Lab (04X)

Reading Assignments

Required reading: TextbookWebsitesOptional library research paper promoting study in current advanced programming

Writing Assignments

Weekly lab assignments will be posted on the class web site. Lab solutions shall be submitted in person by the student in the lab during lab hours. Students will be required to write documentation under the form of comments for their projects. Detailed explanations may be requested by the instructor.

Out-of-class Assignments

.

Demonstration of Critical Thinking

Students will be demonstrating their laboratory projects. Optional research in current advanced programming or other related topics selected by the student and approved by the instructor.

Required Writing, Problem Solving, Skills Demonstration

Weekly lab assignments will be posted on the class web site. Lab solutions shall be submitted in person by the student in the lab during lab hours. Students will be required to write documentation under the form of comments for their projects. Detailed explanations may be requested by the instructor.

Eligible Disciplines

Computer science: Masters degree in computer science or computer engineering OR bachelors degree in either of the above AND masters degree in mathematics, cybernetics, business administration, accounting or engineering OR bachelors degree in engineering AND masters degree in cybernetics, engineering mathematics, or business administration OR bachelors degree in mathematics AND masters degree in cybernetics, engineering mathematics, or business administration OR bachelors degree in any of the above AND a masters degree in information science, computer information systems, or information systems OR the equivalent. Note: Courses in the use of computer programs for application to a particular discipline may be classified, for the minimum qualification purposes, under the discipline of the application. Masters degree required.

Textbooks Resources

1. Required Jesse Liberty, Siddhartha Rao and Bradley Jones. Teach Yourself C++ in One Hour a Day, 6th ed. Sams Publishing, 2009