Tompkins Cortland Community College

Master Course Syllabus

Course Discipline and Number: CIS 223 **Year: 2024-2025**
Course Title: Programming in C++ Advanced Topics **Credit Hours: 3**

Attendance Policy: *To maintain good grades, regular attendance in class is necessary. Absence from class is considered a serious matter and absence never excuses a student from class work. It is the responsibility of all instructors to distribute reasonable attendance policies in writing during the first week of class. Students are required to comply with the attendance policy set by each of their instructors. Students are not penalized if they are unable to attend classes or participate in exams on particular days because of religious beliefs, in accordance with Chapter 161, Section 224-a of the Education Law of the State of New York. Students who plan to be absent from classroom activity for religious reasons should discuss the absence in advance with their instructors. See college catalog for more information.*

Services for Students with Disabilities: *It is the College's policy to provide, on an individual basis, appropriate academic adjustments for students with disabilities, which may affect their ability to fully participate in program or course activities or to meet course requirements. Students with disabilities should contact the Coordinator of Access and Equity Services, to discuss their particular need for accommodations. All course materials are available in alternate formats upon request.*

Course Description

This course is an advanced study of the C++ programming language, focusing on the object-oriented aspects of the language. Topics include classes and objects, inheritance, composition, overloading, and data abstraction. Prerequisites: C or better grade in CIS 213; MATH 095 and RDNG 099 if required by placement testing; ENGL 099 or prior completion or concurrent enrollment in ESL 103 if required by placement testing. 3 Cr. (3 Lec.) Fall semester.

Course Context/Audience

This is a required course for the Computer Information Systems major.

Basic Skills/Entry Level Expectations

Writing: W2 Student should have completed ENGL 099 (if needed). The course requires short written responses and/or short papers without documentation, particularly personal reflection or narrative.

Math: M4 Completed MATH 095(if needed) - Course requires the use of basic mathematical skills plus basic algebra skills.

Reading: R2 Before taking this course, students must have a C or better in RDNG 099 or assessment indicating that RDNG 099 was not required.

Course Goals

This course has two primary goals:

1. To introduce the student to object-oriented programming using C++, and
2. To reinforce strong problem solving, algorithm development, and program documentation skills.

Course Objectives/Topics

Objective/Topic	**# Hours**
The student will be able to create a complete C++ class definition and implementation for a given problem.	6 Hours
The student will be able to create appropriate constructors and destructors for a given class.	3 Hours
The student will be able to separate a class definition into multiple files, using header files and name spaces as appropriate.	3 Hours
The student will be able to use composition to create complex objects.	6 Hours
The student will be able to overload standard operators to work with user-defined objects.	6 Hours
The student will be able to use inheritance to create objects.	6 Hours

The student will be able to create class templates.	3 Hours
The student will be able to work with the String class.	3 Hours
The student will be able to understand the concepts of polymorphism and virtual functions in C++.	3 Hours
The student will be able to describe the importance and uses of data abstraction and data hiding in an object-oriented programming environment.	6 Hours

General Education Goals - Critical Thinking & Social/Global Awareness

CRITICAL THINKING OUTCOMES	**HOW DOES THE COURSE ADDRESS THE OUTCOMES** (Include required or recommended instructional resources, strategies, learning activities, assignments, etc., that must or could be used to address the goal/outcomes)
Students will be able to ➢ develop meaningful questions to address problems or issues. ➢ gather, interpret, and evaluate relevant sources of information. ➢ reach informed conclusions and solutions. ➢ consider analytically the viewpoints of self and others.	Students will be presented with Object Oriented terminology and methodology as a design paradigm. These concepts will frame all development challenges. The instructor should include discussion of terms and techniques followed by hands-on lab exercises to allow for practice and development of individual styles. Students will be introduced to online knowledge bases and shown techniques for implementing previously written modules. The instructor should require demonstration of problem solving approaches and preparation prior to developing a solution. Lab exercises will allow students to practice and develop problem solving skills. Students will be walked through demonstrations from start to finish that encompass all areas of the software development life cycle. Various solutions to problems will be presented and discussed in class. Students should be encouraged to share viewpoints and approaches to problem solving. In class lab collaborations as well as encouraging students to share approaches during lectures should be incorporated into the course.
SOCIAL/GLOBAL AWARENESS OUTCOMES	**HOW DOES THE COURSE ADDRESS THE OUTCOMES** (Include required or recommended instructional resources, strategies, learning activities, assignments, etc., that must or could be used to address the goal/outcomes)
➢ Students will begin to understand how their lives are shaped by the complex world in which they live.	Topics, including software internationalization should be presented and discussed. Current language encoding schemes employed in modern software engineering should be presented. There should be discussions of necessary software and hardware requirements for a language portable system.
➢ Students will understand that their actions have social, economic and environmental consequences	Students are encouraged to collaborate during lab exercises

Instructional Methods

Presentation of new programming constructs is best done in the context of actual programming problems. Walk-through of existing code are very helpful as well.

Methods of Assessment/Evaluation

Method	**% Course Grade**
Exams	60%
Programming projects/homework	40%

Text(s)

C++ How to Program, Harvey Deitel and Paul Dietel, Latest Edition, © 1998 Prentice Hall.

Bibliography

C++ Programming in Easy Steps, McGrath, Mike. © 2005: Computer Step.

C++ Programming Language, Stroustrup, Bjarne. 3rd edition, © 2000: Addison-Wesley Professional.

Effective C++, Meyers, Scott. 3rd edition, © 2005: Addison-Wesley Professional.

Other Learning Resources

Audiovisual No resources specified
Electronic No resources specified
Other No resources specified