Degree Applicable | Glendale Community College
March
2012

COURSE OUTLINE

COMPUTER SCIENCE/INFORMATION SYSTEMS 137
Visual C++ - Object Oriented Programming and Advanced Topics

I. Catalog Statement

Computer Science/Information Systems 137 prepares the student for C++ programming in the workplace and other real world environments. The course will focus on the advanced object-oriented programming concepts needed for today's programs as well as other advanced concepts such as templates and generics, files and streams, and operator overloading. In addition, the course will focus on using the Visual C++ IDE (Integrated Development Environment) and will present some techniques for creating basic Windows-based programs in Visual C++. The course will provide students an opportunity to work on projects involving graphics and game programming with the Ogre 3D graphics engine and on other advanced projects of the student's choosing within the scope of the course.

Units – 4.0
Lecture/Demonstration Hours – 4.0

Prerequisites: Computer Science/Information Systems 135.

II. Course Entry Expectations

Skills Level Ranges: Reading 4; Writing 4; Listening/Speaking 4; Math 3.

III. Course Exit Standards

Upon successful completion of the required coursework, the student will be able to:

1. demonstrate complete understanding of object oriented programming concepts and the application of those concepts to the C++ language;
2. develop a practical understanding of techniques used in the areas of objected oriented programming, graphics, and creating a user interface;
3. apply programming skills to the types of programs commonly encountered in professional
4. create software development projects.

IV. Course Content — Total Contact Hours = 64

A. Introduction to Visual C++ 2008 — 3 hours
   1. Using the Interface
   2. Creating a Basic Window
   3. Background and Basics of using .NET and Visual C++

B. Object Oriented Programming – A Deeper Look — 6 hours
   1. Const (Constant) Objects and const Member Functions
   2. Composition: Objects as Members of Classes
   3. Friend Functions and friend Classes
   4. Dynamic Memory Management with Operators new and delete
   5. Static Class Members
   6. Data Abstraction and Information Hiding

C. Operator Overloading; String and Array Objects 6 hours
D. Object Oriented Programming - Inheritance 7 hours
1. Base Classes and Derived Classes
2. protected Members
3. Relationship between Base Classes and Derived Classes
4. Constructors and Destructors in Derived Classes
E. Object Oriented Programming - Polymorphism 7 hours
1. Fundamentals and Examples
2. Virtual Functions
F. Templates and Generics 7 hours
G. Files and Streams 7 hours
H. Standard Template Library 7 hours
I. Using the Ogre 3D Graphics Library for Game Programming 7 hours
1. Basics of Game Programming
2. Introduction to and Use of the Library
J. Basics of Windows GUI Programming 7 hours

V. Methods of Presentation

The following instructional methodologies may be used in the course:
1. lecture;
2. demonstration;
3. student presentations;

VI. Assignments and Methods of Evaluation

1. midterm examination.
2. final examination.
3. group and individual programming projects (e.g. move a sprite under program and algorithmic control within a window).
4. peer critique of student-originated project.

VII. Textbook

Deitel & Associates, Inc., C++: How to Program, Current Edition.
Upper Saddle River, New Jersey: Prentice Hall; 2010.
12th Grade Textbook Reading Level. ISBN: 0-13-611726-1

VIII. Student Learning Objectives

Upon successful completion, the student will be able to:

1. learn both file stream and event scheduled operating system interfacing and the methods for designing programs that will successfully operate in one or both environments;
2. develop a practical understanding of the techniques used in the areas of data base management, graphics, and creating a user interface;
3. apply programming skills to the types of problems commonly encountered in professional software development projects including intermediate and advanced object oriented programming problems.