COMP270
Windows/Object Oriented Programming with Lab

INSTRUCTOR: Tom Wheeler (Office in Room 208) 941-0430 x5211
twheeler@kc.devry.edu (DeVry e-mail address)
http://www.kc.devry.edu/homepages/twheeler

CREDIT HOURS: 5.0
WEEKLY CONTACT HOURS: 4 Lecture, 2 Laboratory

TEXT: Gosselin, Don: Microsoft Visual C++ .NET (Current Edition)
OPTIONAL TEXT: Lafore, Robert: Object-Oriented Programming in C++ 6 (4/e, Sams)
Note: The optional text is carried over from COMP217, but is not required.

Major Topics Covered In Course:

I. Visual C++ IDE: Navigating the VC Integrated Development Environment.
II. C Concepts: Data, objects, types, conversions, flow control, functions, pointers
III. Classes: Classes, instances, constructors, destructors, private/public/protected members
IV. Advanced Classes: Destructors, copy constructors, operator overloading, inheritance.
V. Introduction to the SDK and Windows Programming: MFC application structure, application architectures, simple application design.

This course provides an introduction to objected-oriented software design using Microsoft Visual C++. Topics include classes, objects, constructors, destructors, dynamic memory allocation, virtual classes, virtual functions, and overloading. The course begins with an overview of C concepts, and provides an introduction to Win32 programming using the Microsoft Foundation Classes (MFC).

ATTENDANCE

Daily class attendance is required. *You are responsible for the material presented in all class sessions, regardless of your presence or absence*. Absence of more than 8 class sessions is cause for dismissal from the course, with a grade of F. You are expected to be on time for every class meeting.

HOMEWORK

Homework is due at the beginning of class (xx00 UTC). *Late homework is not accepted unless mitigating circumstances are present*. If this is the case, bring documentation (court papers, note on doctor's letterhead, etc.) Homework carries the weight of one major exam (100 points) in the course. Failure to do homework will do severe damage to your grade. (UTC=Universal Coordinated Time, or Standard World Time.)

Homework will be kept in a 3-tab folder, with the latest assignment in front. Your name and the course number must appear on the front of the folder in clearly-readable form.

GRADING

There are 2 major exams, an unspecified number of quizzes given at random intervals, various homework assignments, and a final examination given in the 15th week of the course.Your grade will be determined as follows:

2 Major Exams @ 100 points each:	200 points (Sum of two scores)
Quizzes/Homework:	100 points (A percentage)
Laboratory Reports:	100 points (A percentage)
Final Exam (Comprehensive):	150 points
	550 points total for course

Note: There is no drop test, and makeup exams are not given. Alternate exam arrangements must be made at least 3 business days in advance of the scheduled examination. All examinations will be announced at least 1 week prior to administration. All students must take the final exam.

DETERMINATION OF LETTER GRADE FOR THIS COURSE

90 - 100 % = A 80 - 89 % = B 70 - 79 % = C 60 - 69 % = D <60 % = F

PLAGIARISM AND OTHER FORMS OF CHEATING

Copying the work of another, and claiming it to be your own is plagiarism. This includes (but is not limited to) copying others homework, copying from a lab manual or textbook, or collusion. The minimum penalty for cheating in any form is a grade of zero for the element involved; in some cases, failure of the course and/or expulsion from the Institute will also result. *All cases of misconduct will be documented and forwarded to Student Services for disciplinary consideration*. The DeVry Student Handbook contains complete information on this topic.

MISCELLANEOUS INFORMATION

EMERGENCY PROCEDURES - Each classroom has a plaque (located near the door) with instructions for evacuation in the event of an emergency. The instructor will remain in charge of your class group should the situation arise.

FOOD or DRINK are not allowed in the classrooms and labs at DeVry.

HOMEWORK / READING ASSIGNMENTS

NOTE: All assigned readings are from the Gosselin text.

TOPIC or TOPICS	CHAPTER
I. Visual C++ IDE:	1
II. C Concepts:	2,3,4
III. Classes, memory mgmt, object mgmt:	5,6,7
IV. Inheritance, Windows Programming:	8,9,10,11,12,13

Note: You may read the text either before or after the associated class presentations, as you desire. You may need to utilize other text resources (such as from the campus library). Also, for the Windows topics, the site http://msdn.microsoft.com contains a very comprehensive library of most of the "documented" Win32API calls, plus MFC classes.

Course Terminal Objectives:

1. Given a problem statement describing a simple engineering application, such as a very simple climate control system, identify an initial set of object candidates and their possible behaviors representing the problem domain
2. Given a problem statement describing a simple engineering application, such as a very simple climate control system, develop a Graphical User Interface design.
3. Given an initial set of object candidates and their possible behaviors for an engineering problem, construct an object model to identify the appropriate associations among the object candidates.
4. Given an application such as a simplified climate control system, examine the requirements document to correctly assemble a complete list of classes (objects), attributes, and methods.
5. Given an initial object model representing a very simple climate control system, construct an initial dynamic model.
6. Given an initial object model with its dynamic model representing a very simple climate control system, write appropriate algorithms to implement the individual object behaviors.
7. Given a completed initial object model representing the graphical user interface to a climate control system, use Visual C++ and Microsoft Foundation classes to implement the interface.
8. Given a completed initial object model with its dynamic model representing the system behind the user interface, develop the software modules using C++ object-oriented programming language to implement the remainder of the system.
9. Given an application such as a simplified climate control system, analyze the requirements document to create a complete list of primary use-cases representing the series of possible transactions between the actors (users) in the environment and proposed system.
10. Given an application such as a simplified climate control system, analyze the requirements document and use-cases from step 8 to develop a test plan.
11. Given the code to implement an application such as a simplified climate control system, use Visual C++ debugging tools to find both compiler and run-time errors.
12. Document achievement of at least one Lab or Lecture Class TCO with a formal written report and/or an oral presentation in accordance with prescribed guidelines.

Every class is to some extent a unique interactive experience, which may cause some variance within the stated objectives, in either content or level. Individual faculty, based on their experience and expertise, are encouraged to add objectives, as they deem appropriate, and to communicate these directly to the class. The outcomes of the course will depend on the design of the course, the quality of instruction, and the motivation and capabilities of the students, including time available for studying and the effectiveness of the effort.