COMP370
SOFTWARE DESIGN WITH OOP/C++

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: 4.0
WEEKLY CONTACT HOURS: 3 Lecture, 2 Laboratory

TEXT: Lafore, Robert: Object-Oriented Programming in C++ 6 (4th Edition, Sams)

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: (Time permitting, not required by DeVry terminal objectives): 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 time permitting, ends with an introduction to 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. If you will not be able to make it to class on time, please call the instructor in advance to make arrangements.

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.

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: Page references are from the assigned textbook unless otherwise noted.

TOPIC or TOPICS	CHAPTER and PAGE NUMBERS
I. Visual C++ IDE:	C, pp. 863-870
II. C Concepts:	1-5, pp. 1-214
III. Classes:	6, pp. 215-262
IV. Advanced Classes:	7-13, pp. 263-633
V. Introduction to Win32:	VC Online SDK Documentation

Note: You will need to carefully read the text. The instructor's order of topic coverage is not exactly the same as the textbook. You may need to utilize other text resources (such as from the campus library).

COMP370
Midterm Grade Estimation Worksheet
Professor Wheeler

The following table can be used to estimate your standing in my course as of midterm. *You will be required to fill out the data in this table, and bring it to our midterm class meeting.*

There are **no** midterm grades delivered by the DeVry system; by filling in the data in this table, you will compute your own midterm grade in my course. You are responsible for keeping track of the information on this worksheet.

The midterm grade is estimated by a total point system. 300 points are possible, with 100 points coming from each of the following: **Laboratory Percentage**, **Homework Percentage**, and **Exam #1 Percentage**.

Calculating Laboratory Percentage:

In the table below, the number of points possible for each lab is given. Write in your scores (from your returned papers).

Lab 1	Lab 2	Lab 3	Total Score	Percent Score
100	100	100		

Your midterm lab percent score is the sum of all the lab scores entered, divided by the number of labs assigned (3).

A: Midterm Lab Percent Score: _____

Calculating Homework Percentage:

Add up the number of points you've earned on each assignment so far. Divide by the number of possible points (30), and multiply by 100 to get the *midterm homework percentage:*

Assignment	HW 1	HW 2	HW 3	Total Points Earned	Midterm Homework Percentage
Possible Points	10	10	10		
Your Score					

B: Your midterm homework percentage = _____
C: Your score on exam 1: _____

Calculation of Midterm Grade in the Course:

Add up the three sources of percentage points (items A, B, and C), and divide by 3. The result is the percent grade you've earned at midterm.

Your class midterm grade = (A + B + C) / 3 = _____

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 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 using a "CASE" documentation tool.
3. Given an initial object model representing a very simple climate control system, construct an initial dynamic model.
4. 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.
5. Given a completed initial object model with its dynamic model representing the system, develop the software modules using C++ object-oriented programming language to implement the system.
6. 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.
7. Given an application such as a simplified climate control system, examine the requirements document to correctly assemble a complete lists of classes (objects), attributes, and methods.
8. Given a complete correct lists of classes, attributes, methods, associations, and use-casescenarios, draw an object model, object class diagram and event trace diagram for the above proposed system.
9. Given the solution to the problem described in Terminal Objective 8, examine the completed lists of primary use-cases, classes, attributes, associations, and preliminary list of methods to construct a diagram of message layer and dynamic model representing the system's behavior.
10. Given a completed static object model with its dynamic models representing the expanded system, create the software modules using C++ and verify if they meet the system behavior requirements.
11. Given a problem statement describing an engineering database application, such as a sensors database system, create a list of object candidates, each candidate containing an initial set of possible attributes (states) and behaviors representing the problem domain.
12. Given an object container maintaining a set of heterogeneous list of objectives as described in Terminal Objective #11, design a set of algorithms to implement the various object behaviors including polymorphism, operator overloading and friends.
13. Given a completed object model class diagram and a set of function algorithm designs for a sensors database system as described in Terminal Objectives 11 and 12, create menu-driven software program using C++ to implement the systems. Verify that the software meets the functional (behavior) system requirements.

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.