

COP 3330

Object Oriented Programming

Fall 2006 Syllabus

Instructor Information

Position	Name	E-mail	Office	Office hours
Course Instructor	Bob Myers	myers@cs.fsu.edu	Love 205-A	TR 12:15 - 1:45 PM

Course Requirements:

Prerequisite Course

- COP 3014 (Programming I)

Co/Prerequisite Course

- COP 3344 (Introduction to UNIX)

Course Website:

- <http://www.cs.fsu.edu/~myers/cop3330>

Text: Starting Out with C++, Standard Version, Tony Gaddis, 4th Edition, Scott/Jones Publishers, 2005 Update

(this is the same text that is used for COP 3014; we will continue with C++, starting with chapter 13)

Course Objectives:

Upon completion of the course, the student will:

- Demonstrate the use of arrays and pointers in the solution of programming problems using C++
- Understand the Object Oriented Programming concept and be able to discuss the differences between procedural and object oriented languages.
- Demonstrate the ability to create and use classes within the C++ programming language

- Demonstrate the ability to create, compile, and execute programs within the Microsoft Visual C++ environment and the UNIX environment, using the Object-Oriented design model.
- Be able to program using important C++ techniques, such as composition of objects, operator overloads, dynamic memory allocation, inheritance and polymorphism, file I/O, exception handling, templates, preprocessor directives, and basic data structures.

Topics:

Dynamic Memory Allocation
Object-Oriented Concepts
Use of Existing Classes, File I/O
The UNIX Environment, Preprocessor Directives
Composition of Objects
Operator Overloading
Friends
Inheritance
Polymorphism
Exception Handling
Vectors
Templates
Basic Data Structures
The Standard Template Library

Grading Policy:

The final course grade will be computed as follows:

Programs / Assignments	30 %
Test 1	20 %
Test 2	20 %
Final Exam	30 %

In addition to the scale listed above, in order to earn a C- or better in the course, a student is **required** to achieve a test average of C- or better. If the test average is below this level, the highest possible course grade is a D. The test average can be computed with the following formula:

$$\text{TestAvg} = ((\text{Test1} * 20) + (\text{Test2} * 20) + (\text{FinalExam} * 30)) / 70$$

Letter Grade Scale:

Letter	Numerical Average
A	92.00 – 100
A-	90.00 - 91.99
B+	88.00 - 89.99
B	82.00 - 87.99
B-	80.00 - 81.99
C+	78.00 - 79.99
C	72.00 - 77.99
C-	69.00 - 71.99
D	62.00 - 68.99
D-	60.00 - 61.99
F	0.00 - 59.99

Tests:

There will be three tests -- two term tests and a final exam. On-campus students will be required to bring and show your Student ID on test days. The final exam will be cumulative.

Recitation / Quizzes

Periodic quizzes may be given, in lecture or in recitation class, to help students gauge their progress in the class, and to gauge attendance, if needed. Attendance and participation is expected, both in lecture and recitation class. There may also be some hand-in exercises done in recitation class. Any attendance grades, quiz grades, or graded work from recitation will count in the assignment average.

Programming Assignments

- There will be a variety of homeworks and programming projects assigned. Some will be small and easy to complete in one sitting. Others will be larger programming projects. Assignment specifications will be posted on the web page.
- **Turn in all assignments on time!** Unexcused late assignments will be accepted one day after the due date, with the deduction of a letter grade (10%). Assignments more than a day late (unexcused) will not be accepted.
- **Compiling** -- Programs that do not compile are very tedious to grade, and they show a lack of testing, which is a large part of programming. There will be an automatic 5% point penalty for each compile error in a student's code that has to

be fixed in the grading process. (This means that program submissions with compile errors will likely earn very little, if any, credit). Make sure your code compiles before you submit it!!!

Web References:

The course web page is your friend -- check it frequently! It will be continually updated with essential course materials, such as assignments, examples, and notes outlines. It will also include other helpful supplements, such as instructions for using the compilers, a FAQ (Frequently Asked Questions) page, suggested exercises, and other useful help materials. It is your responsibility to check the web page often for posted materials.

Miscellaneous Policies:

1. Students in the class should have a computer account from the Computer Science Department (sign up for one if you don't already have one), and this can and should be used to store project files and access one of the compilers used in the course.
2. Please turn OFF all cellular phones, beepers, etc. in the classroom.

Academic Honor Code:

It is your responsibility to read, understand, and conform to the [Academic Honor Code](#) as set forth in the FSU General Bulletin and the Student Handbook. In addition to this information, please be aware of the following:

- Students are expected to do their **own** work on any classwork or test submitted for a grade (unless designated as a group assignment).
 - It is NOT appropriate to work on assignments with other students or to give or receive solutions to or from anyone before an assignment is due and handed in (by all parties).
 - Discussing solutions and techniques on assignments with other students **after** the assignment has been graded and handed back is okay, and encouraged.
 - When you turn in work with your name on it, you are representing that work as **your own**. If your submission matches that of another student, this is considered a **violation** of the Academic Honor Code.
- If a **group** project is given, then names of all group members would appear on the single program submission. This is appropriate
- Examples found in the course textbook may be used in programs, as long as the source is cited. This is appropriate, as some hand-in assignments may be based on program examples found in the book or contain other code that is provided to you in the assignment specification

- A first violation of the honor code will result, at **minimum** (but not limited to), a penalty of a 0 grade on the assignment or test involved, along with a reduced letter grade in the course.
- Any second violation of the honor code will result in an automatic F in the course, and possible proceedings before the Honor Court.

Accommodation of Disabilities:

Students with disabilities needing special accommodations should register with and provide documentation to the Student Disability Resource Center (SDRC), and they should bring a letter from the SDRC to the instructor indicating what accommodations are needed. Any notice of special accommodations should be given at least a week in advance.

Students taking exams at the SDRC office are expected to take exams at the regularly scheduled time. Any exception to this will only be granted with a valid documented reason and must be approved by the instructor a week before the exam.