

COP 3330 – Object-Oriented Programming – Summer 2011 Syllabus

Course Prerequisites: COP 3223

Class Meets: Monday & Wednesday from 4:00 - 5:50 pm in HEC 125

Instructor: Dr. Mark Llewellyn

Office: HEC 236 **Office Hours:** Monday: 2:00 – 4:00pm
Tuesday 12:00– 2:00pm
Wednesday: 2:00 – 4:00pm

Phone: 407-823-2790 (voice mail available)

E-mail: markl@cs.ucf.edu

Course Web Site: <http://www.cs.ucf.edu/courses/cop3330/sum2011>

Course Objective:

This course is designed to provide a fundamental understanding of the object-oriented programming paradigm using the Java language.

Texts: The following text is required:

An Introduction to Object-Oriented Programming with Java, Fifth Edition, C. Thomas Wu, 2010, McGraw-Hill, ISBN: 978-0-07-352330-9
I'll supplement the text with on-line PowerPoint presentations for each topic.

Grading:

Three exams will be given, a two regular exams and a final exam (somewhat cumulative). Exams are given once – be there as there are no dropped test scores. There will be between four and six programming assignments. Programs submission will be through WebCourses, guidelines will be given later. The programming assignments are to be individual efforts.

Programming assignments (total)	25%
Exam #1 (on or about June 8 th)	25%
Exam #2 (on or about July 11 th).....	25%
Final Exam (regular class time – August 3 rd last day of class).....	25%

Grading Scale:

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

Plus/minus grading will not be used in this course.

Some Important Dates:

No class: [Monday May 31st – Memorial Day](#)
[Monday July 4th – Independence Day](#)
[Last Day to Withdraw: Friday June 24th](#)
Final Exam: [Wednesday August 3rd](#)

Topics To Be Covered:

1. Introduction to OOP and UML.
2. Objects and Classes – instance methods, constructors, parameter binding, static modifier, method overloading.
3. Exceptions and I/O streams.
4. Inheritance – extending classes, class hierarchies, constructors, interfaces, extending interfaces, designing classes.
5. Polymorphism
6. GUI, containers, components, layout managers. (event-driven programming)
7. Design by abstraction, interfaces.
8. Multi-threaded applications

This is a general list of topics only and is subject to the needs of the class. It will be altered without notice, but will generally follow the same progression. At the end of each class I will tell you what we will be discussing during the next class period.

Open Lab Times:

There are no regularly scheduled labs for this course. The TAs for the course will have office hours to answer any programming related questions that you may have and to assist you with the programming of your course assignments. We'll have some time

TA Information:

The TA for the course will be announced later and their contact information will be available on the course webpage.