Computer Science 355
Advanced Object Oriented Programming
3:00 – 4:20 M/W Stuart Hall 308

Instructor: Dr. Stephen Hughes
shughes@coe.edu
399-8231

Office Hours: 315 Stuart Hall
Mon/Wed 11:00 – 12:00
Tues/Thur 2:00 – 3:00
By Appointment or Open Door.

COURSE DESCRIPTION

This course is designed to help you understand the design and mechanics of developing programs within an object-oriented environment. Specifically, the topics in this course will fall into three major themes:

1. *Object Oriented Design and Programming:* We will explore some of the major features and functionality afforded by OO languages. Emphasis will be placed on analyzing real-world situations to effectively design solutions using an OO approach.
2. *Contemporary Programming:* In addition to the core functionality of a programming language, many languages also support extensions to facilitate implementation of mechanisms that are commonly found in modern day systems. The ability to quickly integrate subsets of these extensions into your working repertoire is an essential skill for computer scientists. We will explore some of these specifically, including Graphical User Interfaces (GUIs), Exceptions, Threads, and Network Connectivity.
3. *Software Development Tools and Processes:* Design skills and a command of language syntax will only take you so far. Modern software developers augment theses skills with numerous, powerful tools that help them produce quality software. Among the tools that we will examine are: Integrated Development Environments (IDEs), Unit Testing, Documentation techniques, and Collaboration/Versioning tools.

Prerequisite: Computer Science II (CS-225).

LEARNING OUTCOMES

Upon completion of this course, students will be able to:

- Describe the main concepts of the OO model such as object identity, type constructors, encapsulation, inheritance, and polymorphism.
- Design and implement a coherent class
- Describe a solution in terms of a package of classes with loose coupling between components.
- Identify the relative strengths and weaknesses among multiple designs or implementations for a problem.
- Use object-oriented encapsulation mechanisms such as interfaces and private members.
- Explain why an event-driven programming style is natural in domains where programs react to external actions.
- Explain and use GUI programming concepts; create applications with modern graphical user interfaces.
- Construct and debug programs using the standard libraries available with a chosen programming language.
- Construct, execute and debug programs using a modern IDE and associated tools such as unit testing tools and visual debuggers.
- Explain the importance of Model-View controller to interface programming.
- Recognize commonly encountered patterns of how computations are organized and apply these patterns to software design.

COURSE MATERIALS

Textbooks[***]:

The Object Oriented Thought Process 3/E Weisfeld, Addison-Wesley 978-0672330162

(Recommended) Java 9 Pocket Guide, Liguori, O'Reilly, 978-1491900864
OR Some other trade-press book on Java Programming
OR Reliable access to online materials about Java (There are lots of these)

You should consider having a (4GB min) thumb drive that you can use for storing your coursework for this class

STUDENT ASSESSMENT

Your grades are considered confidential in accordance with FERPA (See page 47 of the Coe College Catalog or online at: http://www.coe.edu/academics/registrar/ferpa).

Your final grade will be determined by the following assessments:

- **15%** **Exam 1:** Week of Sept 28.
- **15%** **Exam 2:** Week of Nov 23 (± a week).
- **15%** **OO Design Project**: Presentation during Final Exam slot: Dec 19, 8:00-10:00
- **55%** **Assignments:** There will be approximately 10 programming assignments over the course of the semester.

Letter grades will be assigned based on the following scale.

	87 ≤ B+ < 90	77 ≤ C+ < 80	67 ≤ D+ < 70	
93 ≤ A	83 ≤ B < 87	73 ≤ C < 77	63 ≤ D < 67	F < 60
90 ≤ A- < 93	80 ≤ B- < 83	70 ≤ C- < 73	60 ≤ D- < 63	

COURSE POLICIES

Expected Workload
This class is expected to consume at least 150 hours of student work over the course of the term. To meet this expectation, you will need to work outside of our scheduled meeting time. You should plan to dedicate a minimum of 10 hours per week to this class.

Attendance
I do not factor attendance directly into your grade. However, I believe that class attendance is vital to your success in this course; conversations held in class illuminate the class materials and should not be missed. Material covered during missed sessions is the responsibility of the student. Graded in-class activities, including quizzes, will not be available for make-up without prior approval or extreme circumstances.

Late Work
All assignments are expected to be submitted on time. However, I understand that events sometimes conspire against us and I am generally open to negotiating due dates. If your work is going to be late, you must:

- Contact me at least 24 hours in advance to ***negotiate*** a new deadline.
- Be prepared to show evidence of progress (i.e. a draft of the work that you have completed)
- Indicate what barriers exist to completing the work on time (i.e. time mgmt., conceptual problems, etc).

Work that is submitted late without prior approval will not receive full credit; work submitted beyond two weeks of the deadline will not be accepted for credit.

[***] These books are available on the Safari Books online for ACM members --- other electronic copies can be found.

Office Hours
Office hours are an opportunity for you to clarify details you may have missed in class, discuss general computer science issues, or to have a profound conversation about marmots. *It is time that is reserved for you*; I may appear busy, but you are not interrupting me – unless another student has arrived first. If you come to office hours with a problem on the assignment, you should come prepared to answer questions, as well as ask them. If you have questions regarding code, you also should come prepared with access to an electronic version of your work.

Academic Integrity
Honesty and integrity are qualities we value in ourselves and in others. You are expected to be fully aware of your responsibility to maintain the highest degree of integrity in all of your work. It is accepted that you have read and understood the standards for academic integrity outlined on page 41 of the Coe College Catalog (online at: http://www.coe.edu/academics/dean/academicintegrity), and will abide by these standards for this course.

I believe that you can learn a lot from your peers, both in the class and in the broader community. Therefore, I *strongly encourage* collaboration with both. However, do not mistake this as a license to cheat. It is one thing to *learn* from and with your peers; it is another to pass their work off as your own. With respect to writing code for this class:

- You are expected to document any collaboration that takes place.
- Absolutely no electronic transfer of code between students is permitted.
- Any code that you "find" on the Internet must be cited, with an active link to that code.
- While you are encouraged to engage in conversations in online forums, under no circumstances are you permitted to solicit other individuals to complete your work for you.

Ultimately, YOU are responsible for all aspects of your submissions. Failure to be able to explain and defend your submission to my satisfaction will be treated as a violation of academic integrity.

Students with Disabilities
Coe College will make reasonable accommodations for persons with documented disabilities. If you have a disability which may have some impact on your work in this course, please contact the Learning Commons' Student Disability and Academic Services Coordinator. All arrangements for accommodations must be handled through the Learning Commons; I am not able to offer individual accommodations without documentation from the Student Disability and Academic Services Coordinator.

End of Course
This course officially ends with the scheduled Final Exam session. No work for this class will be accepted beyond that point