COSC1020 Object Oriented Software Design

Course Description
“Object-Oriented Software Design” provides students with an introduction to object-oriented design and object-oriented programming using Java. Topics include fundamentals of java programming languages, introduction to object oriented analysis and design, and Computer-aided software engineering (CASE) tools.

Course Objectives
Through the lectures and labs, the students will
1. Learn the portions of Java programming language that support object-oriented programming, e.g. class inheritance in Java object-oriented frameworks, interface building in Java.
2. Understand the benefits and limitations of reusability and make use of reusable software components such as function libraries and class libraries.
3. Understand the importance of a software development method, and gain some facility in object-oriented design and programming using Java.
4. Learn about programming environments composed of CASE tools, to understand how software tools can aid the development process, and the role of CASE tools and development methodologies.
5. Learn and experience software development strategies for debugging and testing code.
6. Experience a moderate sized software development in a small team following a prescribed method.

Prerequisites
COSC 1010 or advanced placement.

Meetings
There are two lectures and one 2-hour lab each week.
Lectures: TuTh 2:00-3:15PM at CU412
Labs: Wed 8:00-9:50AM at CU101

Office Hours
Rong Ge: TuTh 9:30AM – 11:00PM at 320 Cudahy Hall or by appointment through email rong.ge at marquette.edu or phone call (414)288-6344
Rizwana Rizia: Wed 10:00-11:00AM at TA Room

Course Website
http://www.mscs.mu.edu/~rge/cosc1020

Assignment
You will have assignment roughly every week except weeks of exams.

Grading Policy
Grades will be calculated using the following formula:
Assignment 35%
Labs 15%
Attendance and participation 10%
Exam #1 10%
Exam #2 10%
Exam #3 20%
Course Policies
1. Students are responsible for checking the course website for announcements, assignments and due dates.
2. Students are encouraged to take notes. Lecture notes will not be available on course website.
3. Late Assignments: 10% penalty for 1-2 days late; 25% penalty for 3-4 days late; 40% penalty for 5 days late. After 5 days no submissions will be accepted.
4. You should write the codes and solutions by yourselves and do not copy other’s work for individual assignments. Plagiarism will result in a score of zero on the assignment or exam, and/or dismissed from the course. Review the Academic honesty code for details.
5. It is acceptable to discuss with classmates a programming assignment in a general way, i.e., to discuss the nature of the assignment. In other words, you may discuss with your classmates what your program is required to accomplish but not how to achieve that goal. In no way should the individual statements of a program or the steps leading to the solution of the problem be discussed with or shown to anyone except the instructor.
6. Debugging is a part of the programming assignment. The instructor is not responsible for debugging students’ codes. You should only use it as the last resort.
7. Exams must be taken on the hour they are scheduled. They will not be given early, and can be made up only if documented evidence of medical emergency or death in the family is presented before the time of the exam.

Topics

1. Fundamentals of Object Oriented Design
   a. Basics of object oriented programming
   b. Classes and objects
   c. Class and Inheritance
   d. Abstract classes and interfaces
   e. Polymorphism and overloading
   f. Java API, packages
   g. Data structure, collections
   h. Exceptions
   i. GUI programming with JWT/SWING
   j. Generic program, template
   k. Defensive programming

2. CASE tools (secondary)
   a. Process of software design
   b. The object model
   c. UML
   d. Rational rose, UML plugin for eclipse
   e. Software testing (junit)

Textbook and References

Textbook

References
3. How to Think Like a Computer Scientist, a free textbook about programming in Java.
4. Two other good, free online textbooks are *Introduction to Programming Using Java* and *Thinking in Java*.
5. *Java Basics* is a great tutorial (in progress as of this writing), and its companion, *Java Notes*, is another excellent source for pretty much every useful topic.
6. *Official Java tutorial*