CS 635 Advanced Object-Oriented Programming
Spring Semester, 2015
Syllabus

San Diego State University -- This page last updated 1/22/15

CS 635 Syllabus

Instructor	Roger Whitney
Office	GMCS 561
Phone	594-3535
Email	whitney@sdsu.edu
Office Hours	3:15-5:15 pm Tuesday, Thursday, 10 am - noon Friday

Course WWW Site: http://www.eli.sdsu.edu/courses/spring14/cs635/index.html. All course handouts will be delivered via WWW at the above URL.

Texts:

- Design Patterns: Elements of Reusable Object-Oriented Software, Gamma, Helm, Johnson, Vlissides, Addison-Wesley, 1995.
- Object Coupling and Object Cohesion, chapter 7 of Essays on Object-Oriented Software Engineering, Vol. 1, Berard, Prentice-Hall, 1993, Will be on reserve at Love Library and at Cal Copy.
- Big Ball of Mud by Brian Foote and Joseph Yoder, http://www.laputan.org/mud/mud.html

Prerequisites: CS535 and working knowledge of Java, C++, Objective C, Ruby or Smalltalk.

This is a graduate course in object-oriented programming that assumes you have taken an undergraduate course in object-oriented programming and have a working knowledge of one of Java, C++, Ruby or Smalltalk. In the recent past a number of students have taken this course without this background and done poorly in this course. February 3 is the last day to add or drop the course. Grades in this course are based only on performance of the student. Problems with languages (English and/or Java) are not considered in assigning grades.

Grading: Your grade will be based on two exams (50% of your grade) and homework (50% of your grade). If needed there will quizzes. There is no extra credit work in this course. There will be between 4 and 5 programming assignments. Missing a programming assignment may drop your course grade by considerably. Some assignments in this class may seem easy at first glance. This causes some students to delay starting the assignment. Often they find out too late that the assignment is harder than they think, which hurts their grade.

Crash Policy: As seats become available in class graduate students on the crash list will be added based on units earned toward your SDSU masters degree on your SDSU transcript.

Only units currently on your SDSU transcript that apply to your graduate degree will be counted. Students with more units given priority over those with fewer units. Crashers need to turn in a copy (hard copy or electronic) of their SDSU transcript to demonstrate how many units they have. Starting on January 27 each class meeting students will be admitted in the class to replace students that drop the course. Student that have not turned in a transcript will be assumed to have zero graduate units. You can email me an unofficial transcript.

Late Policy: Late homework will be accepted, but with a penalty. An assignment will lose 5% of the total value of the assignment per day late. Once a solution to an assignment has been posted or discussed in class, the assignment will no longer be accepted. Late penalties are always rounded up to the next integer value.

Cheating: Any one caught cheating will fail the course in the course and they will be reported to the SDSU Judicial Procedures Office.

My Dad: Over winter break my father was diagnosed with cancer. The cancer has spread and cannot be treated. I will be spending part of the semester helping taking care of him and will be out of town. During that time the course will be offered on-line. I will be recording lectures and posting them on-line. Details of this will be discussed when the class meets.

Course Goals and Outcomes:

Learn basic design patterns
Understand when to apply design patterns to improve the quality of your code
Understand the affects of coupling and cohesion on code quality

This will improve your ability to:

Design, implement, and evaluate a computer-based system, process, component, or program to meet desired needs
Use current techniques, skills, and tools necessary for computing practice.

Topics covered in the course:

- Coupling, Cohesion,
- Refactoring
- Unit testing
- Basic OO concepts and principles
- OO metrics
- Affect of functional programming features on design patterns
- Design Patterns
  - Abstract Factory, Builder, Factory Method
  - Prototype, Singleton, Adapter
  - Bridge, Composite, Decorator, Facade
  - Flyweight, Proxy, Chain of Responsibilty
  - Command, Interpreter, Iterator
  - Mediator, Memento, Observer
  - State, Strategy, Template Method
  - Visitor, Pipes & Filters, Null Object

Disabled Students: If you are a student with a disability and believe you will need accommodations for this class, it is your responsibility to contact Student Disability Services at (619) 594-6473. To avoid any delay in the receipt of your accommodations, you should contact Student Disability Services as soon as possible. Please note that accommodations are not retroactive, and that accommodations based upon disability cannot be provided until you have presented your instructor with an accommodation letter from Student Disability Services. Your cooperation is appreciated.