CS445 - Object Oriented Design and Programming **Last Updated -** 04/29/02
Course Manager – Dr. Bogdan Korel, Associate Professor
3 credit hours; elective for CS & CPE; 150 min. lecture

Current Catalog Description - Introduction to methodologies for object-oriented design and programming. Examines the object model and how it is realized in various object-oriented languages. Focuses on methods for developing and implementing object-oriented systems. Prerequisite: CS 331 or CS 401 or CS 403 (3-0-3)

Textbook

- Riel, Arthur J., *Object-Oriented Design Heuristics*, ISBN 020163385X.

References - other textbooks or materials

- none

Course Goals - Students should be able to:

- Explain and justify the principles of Object Oriented concepts (review abstraction & abstract data types, encapsulation, inheritance, polymorphism, aggregation)
- Analyze and identify the strengths (and weaknesses) of in-depth areas of the Object Oriented paradigm.
- Analyze, explain, & compare the qualities of Object Oriented languages and how well they support the object model.
- Explain and analyze the key points of Object Oriented analysis.
- Explain and analyze the key points of Object Oriented design.
- Design, implement, test and debug multi-phased Object Oriented application.
- Explain and utilize contemporary Object Oriented methodologies (data-driven methodology and behavior-driven methodology)
- Utilize contemporary notation (Unified Modeling Language) to express the artifacts of Object Oriented Analysis & Design (class design, class relationships, object interaction, object states, etc.)
- Perform Object Oriented Analysis & Design on a real-world problem.
- Explain and Utilize Complex Design Patterns.
- Create an implementation of the resultant Object Oriented design.
- Examine new & contemporary concepts in Object Orientation.
- Communicate the deliverables of a software development project.

Prerequisites by Topic

- Strong object-oriented programming experience.

Major Topics Covered in Course

1.	Review of The Terminology And Fundamentals Of Object Oriented Concepts	1.0 hours
2.	Abstractions/Abstract Data Types/Encapsulation/Information Hiding/Coupling/Cohesion	3.5 hours
3.	Object Oriented Hierarchies - Advances Topics on Inheritance/Polymorphism/Dynamic Binding/Aggregations	3.5 hours
4.	"Interface" Class Concepts	2 hours
5.	Object Oriented Languages – Survey, Features	2.5 hours
6.	Characteristics of Objects (Object Relationships, Object Interactions, Instantiation, etc.)	2.5 hours
7.	Object Oriented Analysis & Design - Concepts, Methodologies, Unified Modeling Language	6 hours
8.	Structural Modeling (Class Diagram)	3 hours
9.	Behavioral Modeling (Interaction Diagram, State Diagram)	2 hours
10.	Object-Oriented Design Patterns - Understanding & Usage	3.5 hours
11.	End-To-End Case Study of Object-Oriented Analysis & Design	3 hours
12.	Object Oriented Detailed Design	2 hours

13.	Object Oriented Analysis & Design in Large Scale Projects	2 hours
14.	Use Of Persistence & Databases In an Object Oriented Application	2 hours
15.	Contemporary Object Oriented Topics, Including Multi-Threaded Objects	4 hours
16.	Course Administration & Mid-Term Exam	2.5 hours
17.	Final Exam	-
		45 hours

Laboratory projects (specify number of weeks on each)

- A 2-part large-scale Object Oriented software development project is required by each student in which the student will experience designing, coding, testing and debugging a significant Object Oriented application. The combined parts of the course project are generally range from 5000 - 10000 lines of code.
    - Part 1 (6 weeks): object-oriented analysis and design of the software system.
    - Part 2 (6 weeks): object-oriented detailed design, implementation using object oriented language, and testing of the software system.

Estimate CSAB Category Content in Credit Hours

	CORE	ADVANCED		CORE	ADVANCED
Data Structures		0.5	Computer Organization and Architecture		
Algorithms			Concepts of Programming Languages		1.0
Software Design		1.5			

Oral and Written Communications - Every student is required to submit at least __0___ written reports (not including exams, tests, quizzes, or commented programs) of typically __0___ pages and to make __1___ oral presentations of typically __15___ minutes duration. Include only material that is graded for grammar, spelling, style, and so forth, as well as for technical content, completeness, and accuracy.

- Students orally present (individually to instructor) the deliverables of their Object Oriented software development project.

Social and Ethical Issues - Please list the topics that address the social and ethical implications of computing covered in all course sections. Estimate the class time spent on each topic. In what ways are the students in this course graded on their understanding of these topics (e.g., test questions, essays, oral presentations, and so forth)?

- none

Theoretical Foundations - Please list the types of theoretical material covered, and estimate the time devoted to such coverage in contact (lecture and lab) hours.

- The majority of this course is theoretical - refer to the **Major Topics Covered in Course** section**.**
    - Fundamental Core Principles of Object Orientation - 8 hours
    - Object Oriented Analysis Concepts - 6 hours
    - Topics on Object Oriented Theory - 6 hours

Problem Analysis - Please describe the problem analysis experiences common to all course sections.

- A 2-part large-scale Object Oriented software development project is required by each student in which the student will experience designing, coding, testing and debugging a significant Object Oriented application. The combined parts of the course project are generally range from 5000 - 10000 lines of code.
- A Final Exam project that consists of an Analysis & Design problem the students must analyze & then solve.

Solution Design - Please describe the design experiences common to all course sections.

- A 2-part large-scale Object Oriented software development project is required by each student in which the student will experience designing, coding, testing and debugging a significant Object Oriented application. The combined parts of the course project are generally range from 5000 - 10000 lines of code.
- Final Exam project that consists of an Analysis & Design problem the students must analyze & then solve.

Other Course Information

- Contemporary topics in Object Orientation are regularly rotated into the course material
- A new student project is used every semester, incorporating the latest concepts in Object Orientation.

Other Course Information

- Planned Course Enhancements
    - More extensive Design Pattern lectures are planned to incorporate more Design Pattern concepts in class. (Spring 2003)
    - The course projects will involve implementing selected Design Patterns so the students have both a theoretical understanding of these Patterns as well as the hands-on experience of working with them in code. (Spring 2003)
    - The CS Undergraduate Committee is considering changing the introductory programming language taught to CS and CPE majors to Java in the near future and to stress problem solving and object oriented concepts more heavily. This change may effect CS445 as most of the first 4 "Major Topics Covered in Course" would be covered in pre-requisite classes which would present an opportunity for CS445 to contain more advanced theoretical topics.
    - Review of CS441, CS445, CS447 for possible overlap, better transition, and increase in broader theoretical topics.