

SYLLABUS

CS 50700: Object-Oriented Design and Programming SPRING 2010

INSTRUCTOR: Dr. James H. Hill

LECTURES: MW 6:00 PM – 7:15 PM, SL 109

OFFICE HOURS: MW 4:00 PM – 5:30 PM, or by appointment

PHONE: (317) 274-8527

EMAIL: hillj@cs.iupui.edu, **URL:** <http://www.cs.iupui.edu/~hillj>

REQUIRED TEXTBOOK: *Design Patterns: Elements of Reusable Object-Oriented Software*, Erich Gamma, Richard Helm, Ralph Johnson, and John Vlissides, Addison-Wesley Professional, 1994, ISBN 0-201-63361-2.

RELATED REFERENCES

- *Performance Solutions – A Practical Guide to Creating Responsive, Scalable Software*, Connie U. Smith and Lloyd G. Williams, Addison-Wesley, ISBN: 0-201-72229-1
- *Pattern-Oriented Software Architecture Volume 1: A System of Patterns*, Frank Buschmann, Regine Meunier, Hans Rohnert, Peter Sommerlad, and Michael Stal, John Wiley & Sons, 1995, ISBN 0-471-95869-7.
- *Pattern-Oriented Software Architecture Volume 2: Patterns for Concurrent and Networked Objects*, Douglas Schmidt, Michael Stal, Hans Rohnert, Frank Buschmann, John Wiley & Sons, 2000, ISBN: 0-471-60695-2.
- *Pattern-Oriented Software Architecture, Volume 3: Patterns for Resource Management*, Michael Kircher and Prashant Jain, John Wiley & Sons, 2004, ISBN: 0-470-84525-2.
- *Pattern-Oriented Software Architecture, Volume 4: A Pattern Language for Distributed Computing, Volume 4*, Frank Buschmann, Kevlin Henney, and Douglas C. Schmidt, John Wiley & Sons, 2007, ISBN: 0-470-05902-8.
- *Pattern-Oriented Software Architecture, Volume 5: On Patterns and Pattern Languages*, Frank Buschmann, Kevlin Henney, & Douglas C. Schmidt, John Wiley & Sons, 2007, ISBN: 0-471-48648-5.

PREREQUISITE: CS 452 or CS 506, or graduate standing in the department

COURSE DESCRIPTION: This course is designed to teach students best practices in designing and implementing object-oriented systems of high quality. To accomplish this task, we start with an overview of *software design patterns* and their role in developing high-quality software. We then begin surveying different design-level software design patterns, such as the Bridge, Strategy, Wrapper Facade, and Visitor software design pattern. Next, we touch on software design patterns for building distributed systems. Finally, we finish the course by surveying *software anti-patterns*, which are common design mistakes that negatively impact system quality, such as degrading performance as the system scales in size and complexity. Students will have to opportunity to apply learned techniques on several programming projects throughout the semester.

COURSE OBJECTIVES: The primary goal of this course is to expose students to best practices and techniques for designing and implementing object-oriented systems with high quality. By the end of the course, students will have developed an understanding of the following main topics:

- The purpose of software design patterns
- The design goal and intent of different design patterns

- Knowing what software design pattern to apply to a particular problem
- Techniques for debugging and techniques object-oriented software systems
- Techniques for identifying design flaws that negative impact system quality

TEST & ASSIGNMENTS: There will be no mid-term in this course. Instead, there will be 8 quizzes throughout the semester, and a comprehensive final exam on Wednesday, 5-5-2010 from 6:00 PM – 8:00 PM. In addition, there will be 6 graded programming assignments. The details of the programming assignments, such as detailed description, guidelines, requirements, and deliverables, will be given on their assignment date. Lastly, assignments are due on the specified date. NO LATE ASSIGNMENTS WILL BE ACCEPTED.

GRADING: The following is the weighted percentage for each category toward your final grade.

- Class Participation – 10%
- Programming assignments – 30%
- Quizzes – 30%
- Final Exam – 30%

Grades will be calculated based on a straight scale. Please note that this is a programming intensive course. You are expected to invest a considerable amount of time in designing and implementing programming assignments of high quality. Your participation in class sessions will be taken into consideration towards your final grade.

The following grading scale will be used throughout this course:

<50	>=50	>=55	>=60	>=65	>=70	>=73	>=77	>=80	>=83	>=87	>=90	>=95
F	D-	D	D+	C-	C	C+	B-	B	B+	A-	A	A+

CODE OF CONDUCT: All students are expected to complete their own work without assistance from other students. Moreover, all students are expected to abide by the IUPUI Student Code of Conduct. Failure to comply with the IUPUI Student Code of Conduct will result in disciplinary actions.

TENTATIVE SCHEDULE

Week	Topics & Activities	Related Readings
1	<ul style="list-style-type: none"> • Review of fundamental Software Engineering definitions, principles, & related issues • Review of C++ Concepts • Quiz 1 	Related notes and lecture slides
2	<ul style="list-style-type: none"> • Review of C++ Concepts • Introduction to software design patterns 	Related notes and lecture slides
3	<ul style="list-style-type: none"> • Design-level Software Design Patterns • Assignment 1 Due • Quiz 2 	TBD from textbook, related notes, and lecture slides
4	<ul style="list-style-type: none"> • Design-level Software Design Patterns • Assignment 1 Resubmission 	TBD from textbook, related notes, and lecture slides
5	<ul style="list-style-type: none"> • Design-level Software Design Patterns • Quiz 3 	TBD from textbook, related notes, and lecture slides
6	<ul style="list-style-type: none"> • Design-level Software Design Patterns • Assignment 2 Due 	TBD from textbook, related notes, and lecture slides
7	<ul style="list-style-type: none"> • Design-level Software Design Patterns • Assignment 2 Resubmission • Quiz 4 	TBD from textbook, related notes, and lecture slides
8	<ul style="list-style-type: none"> • Design-level Software Design Patterns 	TBD from textbook, related notes, and lecture slides
9	<ul style="list-style-type: none"> • Design-level Software Design Patterns • Assignment 3 Due • Quiz 5 	TBD from textbook, related notes, and lecture slides
10	SPRING BREAK – NO CLASSES <ul style="list-style-type: none"> • Assignment 3 Resubmission 	
11	<ul style="list-style-type: none"> • Quiz 6 	Related notes and lecture slides
12	<ul style="list-style-type: none"> • Introduction to Distributed Systems • Assignment 4 Due 	Related notes and lecture slides
13	<ul style="list-style-type: none"> • Distributed System Software Design Patterns • Assignment 4 Resubmission • Quiz 7 	Related notes and lecture slides
14	<ul style="list-style-type: none"> • Introduction to Software Anti-patterns 	Related notes and lecture slides
15	<ul style="list-style-type: none"> • Software Anti-patterns • Assignment 5 Due • Quiz 8 	Related notes and lecture slides
16	<ul style="list-style-type: none"> • Software Anti-patterns 	Related notes and lecture slides
17	FINAL EXAM	5-5-2010 from 6:00 PM – 8:00 PM