

Comp Sci-262 Spring 2008

Object-Oriented Design and Programming II (4 units)

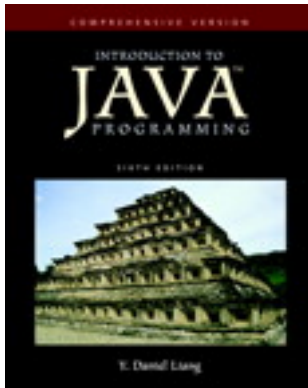
Course: Object-Oriented Design and Programming II (4 units)

Description: A second course in problem solving, software design, and computer programming with the Java language. Problem solving/software design topics will be drawn from: abstract data types, universal modeling language (UML), and finite-state machines (FSM). Data structures and algorithms include: recursive sorting, implementation of linked lists, stacks, queues, and use of other collection classes. Programming topics include: recursion, inheritance, polymorphism, templates, and graphical user interfaces.

Instructor: Wing Huen, HS221, (920)424-1324 **email:** huen@uwosh.edu

Office Hours: Mondays Wednesdays Fridays: 10:10 a.m. to 12:30 p.m.
or by appointment.

Text: *Introduction to Java Programming Comprehensive Version, sixth Edition* by Y. Daniel Liang, Prentice Hall



Lectures: (HS212) Monday, Tuesday, Friday 9:10 – 10:10 a.m.

Labs: (HS101C) Wednesdays 9:10 – 10:10 a.m.

Course Objectives:

- Learn object oriented design/programming concepts
- Learn to implement and test object-oriented software using the Java language

Course Outline:

You are expected to know the following in the prerequisite: Fundamentals of Programming (Chapters 1, 2, 3, 4, 5, 6)

- Introduction (1)
- Primitive data types and operations(2)
- Selection Statements(3)
- Loops(4)
- Methods(5)
- Arrays(6)

This course will cover the following:

- Object-Oriented Design and Programming
 - Objects and Classes (7)
 - Strings and Text I/O (8)
 - Inheritance and Polymorphism (9)
 - Abstract Classes and Interfaces (10)
 - Object-Oriented Design (11)
 - Java Design Patterns
- Debugging with BlueJ, jGRASP, junit
- Graphical User Interface (GUI) (12, 13, 29, 30, 31)
 - JTable
 - JTree
 - JSplitPane
- Event Driven Programming (14)
- Exception Handling, IO, and Recursion
 - Exceptions (17)
 - Binary I/O (18)
 - Recursion(19)
- Data Structures and Collections Framework
 - Lists, Stacks, Queues, Trees, and Heaps(20)
 - Generics(21)
 - Java Collections Framework(22)
- Finite State Machine Design

Courses Slides & Other Material: \Soft on 'Student' (Q:\Shared\Huen\262)

Class Schedule:

	Date s	Monday	Tuesday	Wednesday	Friday	Notables
1	2/4, 2/6, 2/8	Overview, BlueJ, jGRASP Debugging, Finite State Machine	Finite State Machine, Ch. 7 Objects and Classes	Lab 1: Debugging; Finite State Machine Quiz Ch. 5	Ch. 7 Objects and Classes	
2	2/11, 2/13, 2/15	Ch. 7 Objects and Classes	Ch. 9 Inheritance & Polymorphism	Lab 2 Objects/Classes Quiz Ch. 7	Ch. 9 Inheritance & Polymorphism	Assignment 1 on sorting due 2/20
3	2/18, 2/20, 2/22	Ch. 10 Abstract Classes and	Ch. 10 Abstract Classes and	Lab 3 Inheritance & Polymorphism	Java Design Patterns	

		Interfaces	Interfaces	Quiz Ch. 9		
4	2/25, 2/27, 2/29	Java Design Patterns	Java Design Patterns	Lab 4 Interface – sort objects of any type Quiz Ch. 10	Ch. 17 Exceptions and Assertions	Assignment 2 on Inheritance & Polymorphis m due 2/29
5	3/3, 3/5, 3/7	Ch. 17 Exceptions and Assertions	Ch. 8 Text I/O; Finite State Machine 18 Binary I/O; Exam 1 Review	Lab 5 Exceptions	Exam 1	
6	3/10, 3/12, 3/14	Ch. 8 Text I/O; 18 Binary I/O;	Ch. 8 Text I/O; 18 Binary I/O;	Lab 6 Binary I/O Quiz Ch. 8	Ch. 11 Object- Oriented Design; JUnit	Assignment 3 Exception due 3/14
7	3/17, 3/19, 3/21	Ch. 11 Object- Oriented Design; JUnit	Ch. 11 Object- Oriented Design; JUnit	Lab 7 Unit testing Quiz Ch. 18	Ch. 12-13 GUI, Graphics	
	3/23 -30	Spring break	Spring break	Spring break	Spring break	Spring break
8	3/31, 4/2, 4/4	Ch. 12-13 GUI, Graphics	Ch. 14 Event- Driven Programming	Lab 8 Event Quiz Ch. 12-13	Ch. 14 Event- Driven Programming;	Assignment 4 Testing project due 4/1
9	4/7, 4/9, 4/11	Ch. 29, 30, 31 Advanced GUI Programming	Ch. 29, 30, 31 Advanced GUI Programming	Lab 8 Advanced GUI	Ch. 29, 30, 31 Advanced GUI Programming	Assignment 5 Integrating Exceptions with Polymorphis m due 4/11
10	4/14, 4/16, 4/18	Ch. 19 Recursion	Ch. 19 Recursion Review	Lab10 Recursion Quiz Ch. 19	Exam2	
11	4/21, 4/23, 4/25	Ch. 20: Lists, Stacks, Queues, Trees, and Heaps	Ch. 20: Lists, Stacks, Queues, Trees, and Heaps	Lab 11 Data structure assignment Quiz Ch. 14	Ch. 20: Lists, Stacks, Queues, Trees, and Heaps	Assignment 6 Project Due
12	4/28, 4/30, 5/2	Ch. 21 Generics	Ch. 21 Generics	Lab 11 Data structure Quiz Ch. 20		

13	5/5, 5/7, 5/9	Ch. 22 Java Collections Framework	Ch. 22 Java Collections Framework	Lab 12 Collections Quiz Ch. 21	Ch. 22 Java Collections Framework	Assignment 7 due 5/7
14	5/12, 5/14, 5/16	Review	Written Exam3	Lab Exam		

Attendance

Attendance is expected at all class and lab sessions. Students are responsible for all material presented in the course whether or not they attend the class, including announcements about course procedures. It is difficult to envision a student missing and/or arriving unprepared to a number of the class sessions and still succeeding in the course.

Up-to-date information about the course:

You **must** check the class web page regularly, especially when a programming assignment is due. I may need to send out new/modified information; it is *your* responsibility to obtain updates in a timely manner.

Course Grading:

The course grade is based on the total points earned on exams, programming assignments, and quizzes with the following weighting factors:

- Three(3) exams, weighted equally, for a total of 50%
- Between 6 and 8 program assignments, weighted equally, worth 30%
- Class participation: unannounced written quizzes and oral quizzes, worth 10%
- Weekly labs, weighted equally, worth 10%

The course grade will be determined by the following percentage cutoffs:

Score	Grade
92-100	A
89-91	AB
82-88	B
79-81	BC
72-78	C
69-71	CD
57-68	D
< 57	F

Exam Dates

Exam 1: Friday March 7, 2008

Exam 2: Friday April 18, 2008

Exam 3: Tuesday May 13(Written) and Wednesday May 14 (Lab)

Part of the class period before each exam will be a review session. (Changes to these exam dates will be notified approximately two weeks in advance). If you have a

conflict with these dates, please inform the instructor as soon as possible to schedule a makeup exam. The final exam will have a lab component.

Make-up Exam:

If you are unable to take a scheduled exam, you may take a make-up exam provided that you do **BOTH** of the following steps, which are subject to the instructor's approval:

1. Make arrangements *prior* to the scheduled exam (for last minute emergencies, telephone me or leave a message at 424-1324). No after-the-fact notifications will be accepted.
2. Have a written medical excuse signed by the attending physician or a note from the Dean of Students Office.

Only one make-up exam will be given. It will be a comprehensive exam given in the last week of semester.

Programming Assignments: Programming assignments are to be submitted electronically and a hard copy handed in on or before the beginning of the class period on the due date announced. Assignments turned in after this time will be counted as one day late. The grade for the assignment will be reduced by 20% for each day the assignment is late. Saturdays, Sundays, and holidays count when computing penalties. Assignments that are more than 5 days overdue will not be accepted, and will result in a grade reduction. A very late but accepted program will receive a zero grade for that assignment but will satisfy the requirement for all assignments being handed in.

The time of the electronic submission of an assignment will determine the submission date.

It is usually better to turn in a good effort on time rather than be penalized for lateness in order to add finishing touches. However, a non-working program must never be handed in on time just to avoid the late schedule. Do not miss a lecture in order to complete an assignment.

Quality of Submitted Assignments:

1. Adequate and appropriate programming comments, and good programming style.
2. No syntax errors – an assignment submitted with syntax errors will receive a score of zero.
3. Additional written documentation as required.
4. Appropriate and documented test data and test results.
5. Overall professional presentation.

Cooperation and Collusion:

All programming assignments are to be the sole work of the individual student. It is acceptable (and encouraged) for students to discuss the assignment, but all programming and other written work must not involve collusion of any form i.e. sharing, borrowing, or stealing of code (code segments or entire programs). **Program**

coding must be 100% your own work! Sharing or stealing code is academic dishonesty and will result in a zero grade for that assignment. In order to protect yourself, make sure your account is password protected. Don't share your account with anyone. **Don't leave computer listings in the trash or in the computer labs.** When I find two essentially identical listings, I will be forced to apply the penalty to both persons involved.

Laboratory Assignments: Each week you will be given either exercises or programming assignments for the teaching lab. The Lab Assistant and I will be available to answer your questions in the teaching lab. Often you will be asked to do some preliminary work before coming to the lab. It is your responsibility to ensure that you complete this work prior to the lab. The labs are worth 10% of the total grade.

Quizzes: There will be a quiz each week except the weeks of the exams and the first week. The quizzes will be worth 10% of your grade. There will be no make-ups for missed quizzes for any reason. Please see the instructor if you miss a quiz for a valid reason.

Frequently Asked Questions (FAQ)

What do I have to hand in for these programming assignments and what if it's late?

Handing in a programming assignment requires that you (1) submit an electronic copy of your program to the shared directory and (2) hand in a neatly stapled report, which will include a hard copy of your program, as well as notes (including pseudo-code or flowcharts) on how you solved the problem. Each assignment will carry with it a due date. The electronic copy of your assignment is due "at the bewitching hour" on the due date. The hard copy is due at or before the beginning of the first class meeting following the due date. The hard copy must completely match the electronic copy. Failure to submit either component on time means that your assignment is late. Late programming assignments will be accepted but will be penalized at the rate shown above.

Is there any way I can carelessly lose points in the course?

Be late in handing in your work on assignments.

Don't "participate" in the class or lab and do poorly on the quizzes.

Come to your lab session late and unprepared.

What is this class participation stuff? How does one "participate" in a subject like this?

Do well in lab sessions. Prepare for the labs by completing the required reading.

Do well on the written quizzes and oral quizzes.

"Research has demonstrated that after a lecture, students recall 62% of the information. However, only 45% is recalled by students after 3-4 days and in 8 weeks only 24% of the information is recalled. If a quiz or exam was administered after the lecture, recall was doubled at the 8-week period. It is interesting that many faculty members appear to ignore the potential impact which quizzes and tests can have upon learning." -- Bonwell C.C., Eison J.A.: Active Learning: Creating Excitement in the Classroom. Washington, DC: George Washington University, 1991.

Can I get an extension on work that is due on a specified date?

Only if you're gravely ill. Be sure that you have signed documents from a medical professional to verify the illness.

If I miss an exam, can I make it up?

Only if you were gravely ill at the time of the exam. Be sure that you have signed documents from a medical professional to verify the illness. You cannot make up a quiz.

Can I work with others on programming assignments?

There are no team assignments in this course. All the lab and programming assignments are to be the sole work of the individual student. You may help each other by discussing your approach or difficulties with other students but you should not share your program with others or copy the programs of other students. You may discuss the programming assignments at a high level (i.e., work on flow charts) with others, but the lower level work (i.e., all the work on a computer) must be done by you. You are encouraged to seek the help of the instructor and the assigned tutors of this course.

Submitting an assignment that was not entirely done by you is considered academic dishonesty and will result in appropriate disciplinary action.

Can I do programming assignments on my own computer instead of using the computer systems in a university lab?

Sure, if you have your own Windows computer system installed with Java 5.0 and BlueJ 2.0.4. Note that your submitted programming assignments will be graded and tested with the standard versions of Java (currently Java 5.0) and BlueJ (currently 2.0.4). It is your responsibility to ensure that your programming assignments can "Build" and "Start" (i.e. compile and run) on the lab machines.

How can I get a good grade?

Scan the class material before class. Ask the instructor and/or lab assistants on the finer points you don't understand. Work on exercises and the quizzes in the textbook to check your understanding. **Be prepared to spend at least 3 hours for each hour of contact time in class.**