Object Oriented Design & Programming II
Computer Science 262

Instructor:	Erik Krohn
E-mail:	krohne@uwosh.edu
Text Message:	608-492-1106
Class Time:	Monday, Tuesday & Thursday: 12:40pm - 1:40pm
Classroom:	Halsey 202
Lab Time:	Wednesday: 12:40pm - 1:40pm
Lab Location:	Halsey 101C
Office Location:	Halsey 216
Office Hours:	Monday: 11:20am - 12:20pm
	Tuesday: 2:00pm - 3:00pm
	Wednesday: 11:20am - 12:20pm
	Thursday: 2:00pm - 3:00pm
Prerequisites:	Math-108 or equivalent with a grade of C of better, or qualifying for a higher level mathematics via the Mathematics Placement Test, and CompSci-221 or equivalent with a grade of C or better.
Course Website:	`https://canvas.uwosh.edu`
Recommended Textbook:	Introduction to Java Programming, by Y. Daniel Liang Any Edition (10th recommended)

Course Information

A second course in problem solving, software design, and computer programming using an object-oriented language. Problem solving/software design topics include: abstract data types, universal modeling language (UML), simple recursion, unit testing, event-handling, simple concurrency. Data structures and algorithms include: binary search, simple sorting algorithms, use of collection classes and their iteration protocols, sequential file processing. Additional topics include: inheritance, polymorphism, graphical user interfaces, simple use of threads.

Course Website

You should check canvas on a regular basis - it will contain lecture notes, handouts, assignments, announcements, and grades. I'll do my best to let you know when something new and important comes up, but it is your responsibility to check the web site frequently for information that you might not get otherwise.

Mini Assignments

For each lecture, you will have a mini assignment for that lecture. Mini assignments are generally short and should take less than fifteen minutes to complete. You will be assigned a mini assignment every lecture to ensure you are staying current with the material. I will drop your 2 lowest mini assignments. Not all mini assignments will be graded. Solutions to

the mini assignments will be given on the day they are due. Because of this, **no late mini assignments will be accepted.**

Projects & Labs

Most projects and labs will consist of short programming projects. One of your goals (during this class and beyond, in Java or any programming language) should be to write understandable, readable code. You should be making every effort to comment anything that might be confusing to a reader unfamiliar with your program, to name variables intelligently, to use indentation that reflects the code's organization, and so on. All of this will be taken into account during grading: poorly organized or written code may have a negative impact on your grade, even if the resulting program works fine.

One of the goals of this class is to teach you to write functioning programs in Java - thus, your code must compile in order for you to receive any credit. Code that does not compile will not be tested and your score will be a 0. Keep this in mind when writing programs: write your code in small pieces, making sure each piece works before moving on to the next one. It is much better to turn in a project that is not finished but has many working pieces than to turn in one that doesn't work at all, even though most of the code is written.

All assignments must be submitted electronically to the correct dropbox. It is your responsibility to ensure that your submission was submitted correctly. You must double check to ensure your program was uploaded correctly. **See late policy on canvas for late submissions**.

Exams

Exam material will come from the lecture notes, labs, mini assignments, book and assignments. There will be more information about each exam as it approaches. The *tentative* exam dates are listed below. All exams will be taken during the regular class period. These may change, so as the date approaches make sure you've got the most recent information.

- **Exam One** - Monday, October 9th, 2023
- **Exam Two** - Tuesday, November 14th, 2023
- **Exam Three** - Thursday, December 14th, 2023

If you are unable to take a scheduled exam, it may be possible to take a make-up exam provided that you do both of the following, which are then subject to my approval:

1. Make arrangements prior to the scheduled exam. For last minute emergencies, telephone me at 920-424-2063 or leave a message at the computer science office, 920-424-2068 or send me a text message No after-the-fact notifications will be accepted.

2. Have a written medical excuse signed by the attending physician OR have a note of justification from the Dean of Students Office.

If allowed, only one make-up exam will be given. It will be a comprehensive exam given at an arranged time during the last week of the semester.

Grading

Course grades will be based on assignments, mini assignments and exams. Your final grade will be computed with the following percentages:

- 40% - projects

- 15% - mini assignments/labs

- 45% - exams

If you believe anything was graded incorrectly or unfairly and would like to have it regraded, you must let me know about it within *one week* of having the item graded. I will regrade the entire assignment or exam and you may gain or lose points.

Grading will be on a plus/minus system. Grading may be done on a curve depending on the overall performance of the class. If no curve is used, your grade will be computed based on the following:

Percentage	Grade	Percentage	Grade
≥ 92	A	72 - 78	C
90 - 92	A-	70 - 72	C-
88 - 90	B+	68 - 70	D+
82 - 88	B	62 - 68	D
80 - 82	B-	60 - 62	D-
78 - 80	C+	< 60	F

Learning Outcomes

At conclusion of the course, students will be able to:

1. use UML notation to describe the structure of a class and the relationship between classes,

2. identify relationships between classes, including composition, association, and aggregation,

3. create subclasses and superclasses for use in inheritance and polymorphism,

4. create and use abstract classes and interfaces,

5. identify generic types and their uses,

6. handle exceptions,

7. read from and write to files,

8. read from and write to files over a network,

9. write simple queries to retrieve data from a database,

10. create an event-driven program,

11. write recursive methods and identify the parts of a recursive method, and

12. identify when to use different data structures.

Other Information

1. Attendance is not taken in this course. However, you are unlikely to do well if you miss lecture and/or lab.

2. **Academic dishonesty** of any kind will not be tolerated. All assignments, labs, mini assignments and exams are to be completed individually. While discussion of ideas and problems with fellow students is encouraged, all projects and labs must be done individually. In certain circumstances, code fragments from the instructor may be provided to eliminate tedious coding or to provide a common framework for all students. **All other code must be original**. Online resources may be used to help you understand the material, but you may not copy online code nor can you "borrow" code from other students, past or present.

   Any suspected academic dishonesty will be dealt with on a case-by-case basis. Any clarification of what does or does not constitute academic dishonesty must take place **before** you turn in questionable work. For clarification on what constitutes academic dishonesty, contact me or consult the printed policy in the UWO Student Discipline Code, Chapter UWS 14.

3. It is the policy and practice of UW Oshkosh to create an inclusive learning environment. If there are aspects of the instruction or design of this course that result in barriers to your inclusion, please notify me as soon as possible. If you do not have an approved accommodation plan, please complete a registration form through Accommodate: `https://uwosh-accommodate.symplicity.com/public_accommodation/`

4. Students are advised to see the following URL for disclosures about essential consumer protection items required by the Students Right to Know Act of 1990: https://uwosh.edu/financialaid/consumer-information/.

5. Diversity drives innovation, creativity, and progress. At the University of Wisconsin Oshkosh, the culture, identities, life experiences, unique abilities, and talents of every individual contribute to the foundation of our success. Creating and maintaining an inclusive and equitable environment is of paramount importance to us. This pursuit prepares all of us to be global citizens who will contribute to the betterment of the world. We are committed to a university culture that provides everyone with the opportunity to thrive.

6. If any substantive changes are made in the course syllabus, such as changes in schedule or assignments, notification will be provided in a timely manner and a revised syllabus will be made available.