Professor: Dr. David Mathias

email: dmathias@uwlax.edu

website: https://cs.uwlax.edu/~dmathias

Time: Monday, Tuesday, Wednesday and Friday 8:50 AM – 9:45 AM

Classroom: Cowley 301 and Wing Technology 016

Office Hours – Virtual Only: Monday 11:00 – 12:00, Wednesday 2:15 – 3:15 and Friday 11:00 – 12:00, or by appointment. This is the link you will need: https://wisconsin-edu.zoom.us/j/84106988375?pwd=d1FteTRHzDNkM3BBWTlkeXZYQ3R4Zz09

What you can expect from this class: As a student in this course, you will learn more advanced programming principles, fundamental data structures, and the basics of algorithmic efficiency. Though the course is taught using Java, the principles and translate to most other programming languages.

How to succeed: This section is a new addition to my syllabus because of the pandemic. Why? Because the changes necessary to course format make it vitally important that you work to engage with the class: with the material, with me, and with your fellow students. Doing so will be different since we will be together less than during a “normal” semester. So please, force yourself to engage. Force yourself to stay current with assignments. Force yourself to meet your fellow students. Force yourself to visit my virtual office hours. Doing these things will help you in this class.

I encourage you to ask questions. Ask any question. Ask often. Ask again. Then think of more questions. Don’t be surprised if I don’t provide a direct answer but rather lead you to figure things out on your own.

Programming is not something that you learn by just showing up for class and going through the motions. You have to work at it. Some of you will have to work at it a lot. The pandemic doesn’t change that. So think about what you want to get out of this class. Determine what defines success for you and work to make it happen. I hope
that you want to build a strong foundation for the rest of your CS courses and for the 40-year career that will follow graduation. Don’t let the pandemic stand in your way or serve as an excuse. You might have to work differently. You might have to work harder. But you can succeed if you work.

One last note. The three most important words on this syllabus are just below the box on the first page: SUBJECT TO CHANGE. We all have to be flexible. The only thing I can promise in that regard is that I will be as flexible as possible without compromising the course and that I will keep you informed about changes as they occur. This will require that you check Canvas announcements regularly, at least once per day. Anything less than that and you may miss important updates from me.

I’m looking forward to getting to know you and to a good semester. Now to the mundane stuff.

Text Book: We will use two texts this semester. Both are freely available for download or can be purchased if you prefer a hardcopy.


Course Materials: The materials you need in this class are available via my website (URL above). These include in-class assignments, out-of-class assignments, lecture slides, code examples, the current syllabus, helpful links, etc.

Etiquette: When joining a live session via video conference, it is important to observe proper etiquette. Most importantly, make sure that your microphone is muted. Un-mute only to ask questions, and then, only if called on. This rule is intended to prevent chaos. You are encouraged to turn on your camera to make virtual meetings feel more like in-person meetings. If you turn on your camera, please ensure that what is visible in the background is appropriate. While I am not easily offended, others may be more sensitive.

Attendance: I do not take attendance. However, attending class is almost always a very good idea. Knowledge is not poured into your head as water into a glass. Learning requires engaging with the course. If you must miss class due to illness, quarantining, or other obligations, please notify me so that I can advise you of what to do to stay current with the material and any assignments. I will make all reasonable accommodations but ultimately you are responsible for whatever you miss.
Learning Management System: We will use Canvas in this course. IMPORTANT NOTE: I use canvas to record grades for individual assignments. I do NOT use Canvas to calculate your grade for the course. While the total grade displayed on Canvas may approximate your grade, it is not guaranteed to be accurate. The formula I will use to calculate your final grade, not including any curve I apply at the end of the semester, appears below. Feel free to use it to determine where you stand in the class.

Catalog Description: Four hours. Prerequisites: CS 120. Co-requisite: CS 225. This is a second course in the design of programs. Emphasis is placed on data abstraction and its application in design. Definitions of abstract data types are examined. The following structures are examined as methods for implementing data abstractions: recursion, generics, stacks, queues, strings, and various linked lists. Students will be expected to write several programs using these techniques in a modern programming language.

Student Learning Outcomes: By the end of the course, students should be able to:

- Write well-designed code involving inheritance, overloading, and overriding.
- Write code involving multi-dimensional arrays.
- Understand abstraction vs. implementation and choose between alternative linear container representations.
- Write and read code involving exceptions, including try blocks, throw instructions, and throws qualifiers.
- Understand the relationship between files and directories in a hierarchical file directory system and name files with both relative and absolute names.
- Understand the distinction between binary and text files, select between them, and translate data of each type to the equivalent other type.
- Read and write code using the following classes: File, DataInputStream, DataOutputStream, BufferedReader, PrintWriter, Scanner.
- Read recursive definitions; read and write recursive methods, both void and non-void.
- Design programs involving the template design pattern and use Java interfaces and abstract classes to implement this pattern.
- Trace the behavior of code using static, dynamic and automatic memory, understanding the usage of each.
- Read and write code implementing singly and doubly linked lists including traversal, item insertion and removal algorithms, sentinel cells, and pre-pointers.
• Read and write code involving inner private classes; understand their utility for implementing linked lists in the style of `java.util.LinkedList`.

• Read and write code implementing stacks and queues.

• Read and write code involving the following Java classes: `java.lang.Comparable`, `java.util.Collection`, `java.util.List`, `java.util.Iterator`.

• Perform simple counting analyses on linear, polynomial and logarithmic algorithms.

• Develop programs involving all of the following algorithms: binary search, linear search (for both arrays and lists), merge sort, quicksort.

• Interpret and utilize object type conformance and subtype polymorphism.

• Read, write, and use generic classes.

**Programming Assignments:** One of the largest components of your grade in this course is outside-of-class programming assignments. The only way to learn how to program is by programming. You will not be successful in this class unless you write your own code. I encourage you to study with others and to discuss concepts and ideas with classmates. However, you must write and submit your own work.

Rules for program submission:

• Programs are due at 11:59 PM on the due date. There is a long and proud tradition of programs being due at that time. You are, of course, welcome to submit well before the deadline.

• Late submissions are accepted up to 48 hours after they are due. Up to 24 hours late, there is a 20% penalty. Up to 48 hours late, there is a 50% penalty. Programs will not be accepted more than 48 hours late.

• You must include the following at the top of your program file: a block comment with the following information (in this order and each on a separate line): your name, CS 120 followed by your section number, the assignment number, and the due date. Next is a blank line followed by a brief description of the assignment.

• Comment your code (we’ll discuss in class what this means). You will be tempted to skip this. Resist that temptation. Good commenting is critical.

• Adhere to the coding conventions we discuss in class. I don’t care that your uncle’s wife’s cousin told you to do things differently. There are many ways to format code, and I don’t claim that my way is the one and only right way, but it will be much easier for us to talk about code if we all use a common format.

**Evaluation and Assessment:**
• 35% – Programming assignments
• 10% – Closed labs
• 5% – In-class exercises
• 25% – Midterm
• 25% – Final exam

**Grading Scale:** Letter grades will be assigned according to the table below. Let $x$ be your numeric score for the course:

- A: $x \geq 93$
- AB: $89 \leq x < 93$
- B: $83 \leq x < 89$
- BC: $79 \leq x < 83$
- C: $70 \leq x < 79$
- D: $60 \leq x < 70$
- F: $x < 60$

**Course Policy on Collaboration:** For programs, you may discuss general concepts with classmates. However, you can not collaborate in the preparation of solutions or programs. Using any solution that you did not create yourself, including but not limited to solutions found on the Internet, is considered academic misconduct.

**Academic Integrity:** Academic misconduct is a violation of the UWL Student Honor Code ([http://catalog.uwlax.edu/undergraduate/academicpolicies/studentconduct](http://catalog.uwlax.edu/undergraduate/academicpolicies/studentconduct)) and is unacceptable. I expect you to submit your own original work and participate in the course with integrity and high standards of academic honesty. When appropriate, cite original sources, following the style rules of our discipline.

PLEASE NOTE that whenever a grade penalty is imposed due to academic misconduct, the instructor is required to write a letter documenting the misconduct. Copies are sent to the student, to the Office of Student Life (where the letter remains on file in the student’s record), and to the Dean of the student’s College. Refer to [https://www.uwlax.edu/student-life/student-resources/student-handbook](https://www.uwlax.edu/student-life/student-resources/student-handbook) for a detailed definition of academic misconduct, and for possible sanctions and consequences. The Office of Student Life can also assist.

Plagiarism or cheating in any form may result in failure of the assignment or the entire course, and may include harsher sanctions. Refer to the Student Handbook #14.02 for a detailed definition of academic misconduct.
For helpful information on how to avoid plagiarism, go to “Avoiding Plagiarism” on the Murphy Library website (http://libguides.uwlax.edu/plagiarism2). You may also visit the Office of Student Life (https://www.uwlax.edu/student-life/) if you have questions about plagiarism or cheating incidents. Failure to understand what constitutes plagiarism or cheating is not a valid excuse for engaging in academic misconduct.

**UWL COVID-19 Statement:** All UWL students are encouraged to be vaccinated against COVID-19. All students are required to be masked in classrooms and other indoor campus communal spaces. Campus-wide mask guidance may change during the semester. Students with COVID-19 symptoms or reason to believe they were in contact with COVID-19 should call and consult with a health professional, such as the UWL Student Health Center (608-785-8558), regardless of their vaccination status. Students who are ill or engaging in self-quarantine at the direction of a health professional should not attend class. Students in this situation will not be required to provide formal documentation and will not be penalized for absences. However, students should:

- notify instructors in advance of the absence and provide the instructor with an idea of how long the absence may last, if possible
- keep up with classwork if able
- submit assignments electronically
- work with instructors to either reschedule or electronically/remotely complete exams, labs, and other academic activities
- consistently communicate their status to the instructor during the absence

Instructors have an obligation to provide reasonable accommodation for completing course requirements to students adversely affected by COVID-19. This policy relies on honor, honesty, and mutual respect between instructors and students. Students are expected to report the reason for absence truthfully and instructors are expected to trust the word of their students. UWL codes of conduct and rules for academic integrity apply to COVID-19 situations. Students may be advised by their instructor or academic advisor to consider a medical withdrawal depending on the course as well as timing and severity of illness and students should work with the Office of Student Life if pursuing a medical withdrawal.

**PRO-UWL:** This class is participating in the Navigate Progress Report Online early alert system designed to promote student success. If I notice that you are experiencing difficulties early in the semester (e.g., low assignment scores or limited participation), I may note this information and you will receive notification indicating that I have entered feedback. I encourage you to meet with me and/or utilize helpful campus
resources listed on UWL’s Student Success page:
https://www.uwlax.edu/info/student-success/

Inclusive Excellence: UWL’s core values include “Diversity, equity, and the inclusion and engagement of all people in a safe campus climate that embraces and respects the innumerable different perspectives found within an increasingly integrated and culturally diverse global community” (https://www.uwlax.edu/chancellor/mission).

If you are not experiencing my class in this manner, please come talk to me about your experiences so I can try to adjust the course if possible.

Student Course- and Faculty-Related Concerns, Complaints, and Grievances:

Informal Complaints:

If a student has a concern or a complaint about a faculty member or course, the general process for making informal complaints is outlined in steps 1-3 below. Students are welcome to bring a friend or a UWL staff member with them during the following steps. Students who report concerns/complaints/grievances, whether informally or formally, will be protected from retaliation and have the right to expect an investigation and the option to have regular updates on the investigation:

1. The student should speak directly to the instructor.

2. If the student is uncomfortable speaking with the instructor, or they are unsatisfied with the solution, they should go to the chair of the faculty member’s home department.

3. If the student is uncomfortable speaking with the department chair, or the chair is the faculty member in question, or they are unsatisfied with the solution, the student should speak with their college dean.

Depending on the specifics of the student’s concern, it may be helpful for them to reach out to additional offices:

- Complaints/concerns/grievances about grades, teaching performance, course requirements, course content, incivility, or professional ethics should follow the process outlined above. Students may also wish to seek support from the Student Life office.

- Complaints/concerns/grievances related to hate/bias and discrimination may follow the process outlined above, and in addition or instead students may contact the Campus Climate office and/or submit a hate/bias incident report.

- Complaints/concerns/grievances related to sexual misconduct may begin with the process outlined above, but will need to also involve the Equity & Affirmative
Action and Violence Prevention offices, and/or the Title IX Team. Students should know that faculty members are mandatory reporters of sexual misconduct, but that confidential resources are available to them.

Formal Complaints:
If the student is unsatisfied with the solution of their informal complaint, they have the right to file a formal institutional complaint with the Student Life office, as described in the Student Handbook.

Student Evaluation of Instruction: UWL conducts student evaluations electronically. Approximately 2 weeks prior to the conclusion of a course, you will receive an email at your UWL email address directing you to complete an evaluation for each of your courses. In-class time will be provided for students to complete the evaluation in class. Electronic reminders will be sent if you do not complete the evaluation. The evaluation will include numerical ratings and, depending on the department, may provide options for comments. The university takes student feedback very seriously and the information gathered from student evaluations is more valuable when a larger percentage of students complete the evaluation. Please be especially mindful to complete the surveys.

Useful Resources: The following links are provided for your convenience. This is not an exhaustive list of services available on campus.

ACCESS Center: http://www.uwlax.edu/access-center
Student Support Services: http://www.uwlax.edu/student-support-services

For statements regarding Sexual Misconduct, Religious Accommodations, Students with Disabilities, and Veterans and Active Military Personnel, please see: https://www.uwlax.edu/info/syllabus
Approximate schedule (subject to change):

<table>
<thead>
<tr>
<th>Week:</th>
<th>Topics:</th>
<th>Text:</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>January 24</td>
<td>Course introduction</td>
</tr>
<tr>
<td>2</td>
<td>January 31</td>
<td>Abstract classes and interfaces</td>
</tr>
<tr>
<td>3</td>
<td>February 7</td>
<td>Memory management</td>
</tr>
<tr>
<td>4</td>
<td>February 14</td>
<td>File systems, file I/O, exceptions</td>
</tr>
<tr>
<td>5</td>
<td>February 21</td>
<td>Multi-dimensional arrays</td>
</tr>
<tr>
<td>6</td>
<td>February 28</td>
<td>Collections, ArrayLists, generics</td>
</tr>
<tr>
<td>7</td>
<td>March 7</td>
<td>Linked lists</td>
</tr>
<tr>
<td></td>
<td>March 14</td>
<td>Spring Break</td>
</tr>
<tr>
<td>8</td>
<td>March 21</td>
<td>Midterm</td>
</tr>
<tr>
<td>9</td>
<td>March 28</td>
<td>Iterators</td>
</tr>
<tr>
<td>10</td>
<td>April 4</td>
<td>Doubly-linked lists, algorithmic analysis</td>
</tr>
<tr>
<td>11</td>
<td>April 11</td>
<td>Recursion</td>
</tr>
<tr>
<td>12</td>
<td>April 18</td>
<td>Search algorithms, stacks, queues</td>
</tr>
<tr>
<td>13</td>
<td>April 25</td>
<td>Sorting algorithms</td>
</tr>
<tr>
<td>14</td>
<td>May 2</td>
<td>Review</td>
</tr>
<tr>
<td><strong>Final</strong></td>
<td></td>
<td>Tuesday May 10 at 12:15 PM</td>
</tr>
</tbody>
</table>