“Smart data structures and dumb code works a lot better than the other way around.”
—Eric Raymond, *The Cathedral and the Bazaar*

**Course Info**

- **Prerequisites**: CSCE 155; MATH 106 or equivalent recommended
- **Instructor**: Chris Bourke  
  [cbourke@cse.unl.edu](mailto:cbourke@cse.unl.edu)  
  Avery Hall 363
- **Office Hours**: See Webpage
- **Textbooks**: See Webpage
- **Labs & Teaching Assistants**: See Webpage

**Course Description**

Data structures, including linked lists, stacks, queues, and trees; algorithms, including searching, sorting, and recursion; programming language topics, including object-oriented programming; pointers, references, and memory management; design and implementation of a multilayer application with SQL database.

**Course Objectives**

The official specification for this course lists the following general course objectives.

1. A second programming language (students new to Java will have an introduction to java, those familiar with Java will work with PHP
2. Creating and querying a relational database and programming a database application.
3. Object-oriented design and programming, including encapsulation, composition, inheritance, and polymorphism.
4. Pointers/references, parameter passing, memory management, memory and buffer attacks and security
5. Implementation and use of unsorted and sorted lists, including array and linked implementation, stacks and queues, circular and doubly linked lists.
6. Searching and sorting algorithms with related data structures including binary search trees.
7. Exposure to computational complexity of algorithms.

A detailed course specification is available:
[http://cse-apps.unl.edu/crspec/course_specifications/22/view](http://cse-apps.unl.edu/crspec/course_specifications/22/view)

**Course Topics & Tentative Schedule**

This is a rough sequence of topics that we’ll cover in this course.
1. Introduction to Java / PHP
2. Object Oriented Programming, OOP in Java
3. Database Design, SQL, Database Connectivity (JDBC)
4. 3-tier architecture
5. Lists (array-based, linked; circular doubly-linked)
6. Algorithm Analysis
7. Recursion
8. Sorting
9. Searching
10. Stacks
11. Queues
12. Trees; Binary Search Trees

**Skill Objectives**
Throughout the duration of this course, you will be presented with a lot of information, some of which will have immediate application and some whose importance is not immediately apparent. As a different measure of your success in this course, by the end, you should have the following skills.

- Be able to more easily pick up and learn a new programming language on your own
- Be able to recognize, use, design and appreciate programs with an Object Oriented flavor
- Be able to design a database-backed application from scratch given a general problem statement
- Be able to understand and implement basic data structures as well as recognize how and when to leverage appropriate simple data structures
- Be able to identify and analyze basic algorithms

Overall, CS2 is a continuation of CS1. At the end of a typical CS1 course, you should be able to approach a reasonably complex problem, design a top-down solution, and code a program in a high-level programming language. By the end of this course your approach to problem solving should be modified to ask and answer design questions *first*:

- What entities/objects would be appropriate to model this problem?
- How would these objects interact to solve the problem?
- What data structures would be the most appropriate or efficient to use?

**Honors Section**
During spring semesters a separate honors section of this course is held. The honors sections covers the same topics but at a faster pace and a greater depth. Some additional advanced topics will also be covered as time allows. These may include the following.

- More web development with PHP and/or JSP
- Additional web-programming languages: JavaScript, jQuery
- Software Design Patterns
- Java Persistence API (JPA)
- Concurrency
- Advanced tree structures, balanced binary search trees

There will be higher expectations with regard to the quality and quantity of work in the honors section compared to the regular section. As a student in this course, you will be expected to meet or exceed those expectations, which may require more of your time and a more proactive approach to studying and greater diligence in completely assignments. In particular, each assignment contains additional requirements or activities. In the regular section these will be opportunities for bonus points. However in the honors section, these are required and will be treated as part of your regular grade. In addition, there are some additional bonus assignments available. Students in the honors section will be required to do at least one of these bonus assignments as part of their regular homework grade.

**Grading**
Grading will be based on homework, labs and exams with the following contributions

<p>| | |</p>
<table>
<thead>
<tr>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>Homework</td>
<td>75%</td>
</tr>
<tr>
<td>Labs</td>
<td>15%</td>
</tr>
<tr>
<td>Midterm</td>
<td>5%</td>
</tr>
<tr>
<td>Final</td>
<td>5%</td>
</tr>
</tbody>
</table>

**Scale**
Letter grades will be awarded based on the following scale. This scale may be adjusted upwards if the instructor deems it necessary based on the final grades only. No scale will be made for individual assignments.

<table>
<thead>
<tr>
<th>Letter</th>
<th>Minimum</th>
</tr>
</thead>
<tbody>
<tr>
<td>A+</td>
<td>&gt;= 97</td>
</tr>
<tr>
<td>B+</td>
<td>&gt;= 87</td>
</tr>
<tr>
<td>C+</td>
<td>&gt;= 77</td>
</tr>
<tr>
<td>D+</td>
<td>&gt;= 67</td>
</tr>
<tr>
<td>A</td>
<td>&gt;= 93</td>
</tr>
<tr>
<td>B</td>
<td>&gt;= 83</td>
</tr>
<tr>
<td>C</td>
<td>&gt;= 73</td>
</tr>
<tr>
<td>D</td>
<td>&gt;= 63</td>
</tr>
<tr>
<td>A-</td>
<td>&gt;= 90</td>
</tr>
<tr>
<td>B-</td>
<td>&gt;= 80</td>
</tr>
<tr>
<td>C-</td>
<td>&gt;= 70</td>
</tr>
<tr>
<td>D-</td>
<td>&gt;= 60</td>
</tr>
</tbody>
</table>

**Homework**
There will be 6-8 homework assignments. Assignments are due at the beginning of class. Code and other relevant files must be submitted using CSE’s web handin. If there are written portions of the homework, they should be typed. You should typeset code snippets using a monotype font (Courier for example) for readability. Figures may be hand drawn, but you are encouraged to use some sort of software to render them.

In this course, many homework assignments build on prior assignments. Therefore, it is extremely important for you to stay on task and not fall behind with assignments. You should start on assignments early and resolve any and all issues (bugs, design flaws) as soon as possible so they do not carry over to the next assignment.

Note: the final homework may be due as late as Friday of Dead Week. As per the 15th week policy, this serves as notice.
Labs
There will be weekly labs that give you hands-on exercises for topics recently covered in lecture. The purpose of lab is not only to give you practical experience with lecture topics, but also to provide you with additional information and details not necessarily covered in lecture. Each lab will have some programming requirements and a supplemental worksheet. Lab materials will generally be available at least 1 week prior to the lab meeting time. You are expected to familiarize yourself with the lab handout and any required or suggested reading as stated in the handout. Some labs may also have a worksheet that that requires you to answer questions and/or demonstrate working code to one of the lab instructors. Unless otherwise stated, you are required to finish the lab by the end of your regular lab meeting time. A lab instructor must sign off on your lab worksheet and you must turn it in to receive credit. No labs will be accepted via email. Labs that have not been completed by then will be given a zero.

Exams
There will be a midterm exam and a comprehensive final exam. Details will be announced closer to the exam dates.

Grading Policy
We make every effort to grade and return materials within one academic week of the due date. You are expected to pick up your graded material in a timely manner. If you have questions about grading or believe that points were deducted unfairly, you must first address the issue with the individual who graded it to see if it can be resolved. Such questions should be made within a reasonable amount of time after the graded assignment has been returned. No further consideration will be given to any assignment a week after it has been graded and returned (regardless of whether you fail to pick it up when handed back). It is important to emphasize that the goal of grading is consistency. A grade on any given assignment, even if it is low for the entire class, should not matter that much. Rather, students who do comparable work should receive comparable grades (see the subsection on the scale used for this course).

Late Work
In general, there will be no make-up exams. Exceptions may be made in certain circumstances such as health or emergency, but you must make every effort to get prior permission. Documentation may also be required.

Homework assignments have a strict in-class (at the beginning) due date. Any written portions should be handed in hardcopy in class while softcopies should be handed in using webhandin as specified in individual homework assignments. The webhandin program that you will use enforces a strict handin time based on the CSE server's clock. Programs that are even a few seconds past the due date/time will be considered late.

Furthermore, many assignments will have requirements (file naming conventions, package requirements, command line input requirements, etc.) that will facilitate grading through an automated script. This script has been made available to you through the webgrader interface. You are expected to utilize this webgrader interface to ensure that your program is running as required and to fix any
issues prior to the final due date (you may handin and run the script as many times as you like up to the due date). Note, however, that the script should not substitute for developing your own test cases and should not be used as the primary resource to debug your program; instead it is intended as a last-check mechanism.

It is understandable that unforeseen events may interfere with your ability to submit all homework assignments on time. As such, this course allows the following late work policy: you may hand in any ONE assignment up to one (academic) week late. Any submissions after a week will not be considered and will be given an automatic zero. Any late submissions after using your one “free pass” will not be considered.

If you work with a partner on a late assignment, both of your late passes will be used. If two people work together on an assignment and one of them has already used their late pass, the other may not use their late pass for both of them.

In addition, failure to adhere to the requirements of an assignment in such a manner that makes it impossible to grade your program via the webgrader means that a disproportionate amount of time would be spent evaluating your assignment. For this reason, we will not grade any assignment that does not compile and run through the webgrader. You will be expected to use one of your late passes to fix the issue(s) before we will consider it for grading. Failure to address the issue or submitting assignments with such problems will result in an automatic zero.

**Exceptions:** You may NOT use a late pass on either the first nor the last homework assignments. The last assignment is due during dead week and it is not possible to grade late materials beyond that week. Late passes are not allowed on the first assignment because an inability to get the first assignment done in a timely manner is a strong indication that you are not academically prepared to take this course.

**Dead Week Policy**
In conformance with UNL’s 15th Week Policy (see Registration and Records main webpage, [http://www.unl.edu/regrec/](http://www.unl.edu/regrec/)), be aware that the final homework may be due during the final week of classes. Further, there will be a regularly scheduled lab during the final week of classes. Finally, all assignments, homework, labs or otherwise, will have a strict final due date during the final week of classes. This supersedes any unused late or screw-up passes that you may have (that is, such passes cannot be used to extend the due date of any assignment past the last week of classes).

**Academic Integrity**
All homework assignments, programs, quizzes, and exams must be your own work unless otherwise stated. No collaboration with fellow students, past or current, is allowed unless otherwise permitted on specific assignments or problems. The Computer Science & Engineering department has an Academic Integrity Policy. All students enrolled in any computer science course are bound by this policy. You are expected to read, understand, and follow this policy. Violations will be dealt with on a case by case basis and may result in a failing assignment or a failing grade for the course itself. The most recent version of the Academic Integrity Policy can be found at [https://cse.unl.edu/academic_integrity](https://cse.unl.edu/academic_integrity)
Communication

The best way to communicate with your instructor is through Piazza. The instructor and teaching assistants will communicate with you either directly or through this system. You are responsible for ensuring that you sign up for this service and follow its instructions.

In addition, the Department of Computer Science & Engineering also maintains an anonymous suggestion box that you may use to voice your concerns about any problems in the course or department if you do not wish to be identified. It is available at the following URL: http://cse.unl.edu/department/suggestion.php

Finally, I will hold regular office hours (to be announced) and will make myself available by appointment; please email me to set one up.

Help

Your success in this course is ultimately your responsibility. That said, there several outlets for you to seek help and assistance.

1. Your Instructor – Attend lecture regularly and engage in class discussions, ask questions in class, visit me during my office hours or setup a meeting time to see me, email me!
2. Your TAs – TAs hold regular weekly office hours, visit with them and ask questions, ask for examples, etc. Attend the weekly homework and lab help sessions
3. Student Resource Center – Though they may not be your direct TA, all Graduate Teaching Assistants (and some advanced undergraduates) hold regular office hours in the Student Resource Center (http://cse.unl.edu/src) open Monday thru Friday 9AM to 7PM and should be staffed most hours. Ask for help from anyone in the SRC.
4. Course materials – start on assignments early, attend lectures and labs, work extra problems from the book, read all required (and optional!) materials.
5. Your colleagues – Chances are, if you are having problems, your classmates are having them too. Discussion and dialog among students is encouraged (within the parameters set by CSE’s academic integrity policy, this course’s policy, and policies set for individual assignments).