CS201 Data Structures and Algorithms

Syllabus

Instructor

<table>
<thead>
<tr>
<th>John C. Lusth</th>
<th>office: 3427 SEC</th>
</tr>
</thead>
<tbody>
<tr>
<td>phone: 204-348-1618</td>
<td>email: <a href="mailto:lusth@cs.ua.edu">lusth@cs.ua.edu</a></td>
</tr>
<tr>
<td>home page: <a href="http://beastie.cs.ua.edu/cs201">http://beastie.cs.ua.edu/cs201</a></td>
<td>office hours: sign-up sheet</td>
</tr>
</tbody>
</table>

Catalog Statement

Provides an introduction to formal techniques for analyzing the complexity of algorithms. The course surveys important classes of algorithms used in computer science and engineering.

Data structures including balanced search trees, heaps, hash tables, and graphs. Algorithm design techniques including divide-and-conquer, greedy method, and dynamic programming. Emphasis on problem solving, design, analysis, and reasoning about data structures and algorithms.

Implementing the Catalog Statement

You will be required to demonstrate your proficiency with implementing algorithms through three programming assignments. The C programming language will be the language used for the three projects.

You will also be tested on your knowledge of algorithms through two concept exams and a final exam.

Prerequisites

CS 101, MATH 301, MATH 302 (with concurrency)

There will be an exam on prerequisite material.

Textbook

*Introduction to Algorithms*, Third Edition

Thomas H. Cormen, Charles E. Leiserson, Ronald L. Rivest, and Clifford Stein

ISBN-10: 0-262-03384-4

Topics

The following sections, chapters and topics from the text will be covered:

- Section III: Data Structures — Chapters 10...12
- Section II: Sorting — Chapters 6...8
- Section II: Order Statistics — Chapter 9
- Section I: Foundations — Chapters 1...4
- Section III: Data Structures — Chapter 13
- Section IV: Advanced Design and Analysis Techniques — Chapters 15,17
- Section V: Advanced Data Structures — Chapters 19,21
- Section VI: Graph Algorithms — Chapters 22,23
- Section VII: Selected Topics — Chapter 34

though not necessarily in order.
Grading

Grades will be determined by the following percentages:

<table>
<thead>
<tr>
<th>Percentage</th>
<th>Component</th>
</tr>
</thead>
<tbody>
<tr>
<td>10%</td>
<td>Prereq Exam (click here for a practice exam)</td>
</tr>
<tr>
<td>20%</td>
<td>First Concept Exam</td>
</tr>
<tr>
<td>20%</td>
<td>Second Concept Exam</td>
</tr>
<tr>
<td>20%</td>
<td>Final Exam</td>
</tr>
<tr>
<td>30%</td>
<td>Four programming assignments</td>
</tr>
</tbody>
</table>

Grading Methodology

For programming assignments, you must pass all tests. If you fail even one test, you will have to resubmit. You are allowed four resubmissions, with each resubmission resulting in a 10 point deduction. If you cannot pass all tests after your fourth resubmission, you will receive a zero for the assignment.

Submissions and resubmissions have due dates, as specified in the schedule. Missing a submission or resubmission date is the same as failing all tests. Generally, but not always, submission and resubmission dates are seven days apart.

For programming assignments, you will receive a bonus based upon how successful your initial submission was. The formula for the bonus is the percentage of tests passed divided by twenty. For example, suppose you passed 80% of the tests with your original submission and you passed all tests on your second resubmission. In this case, your score for the assignment would be $100 - 10 - 10 + \frac{80}{20}$ or 84 out of 100. Passing all tests with the original submission will yield a score of 105 out of 100.

Grading for the class is straight scale with the following cut-offs: 90% and above is an A, 80% and above is a B, 70% and above is a C, 60% and above is a D, and anything lower than 60% is an F. These cut-offs are strict. For example, 79.99999% is a C. Plusses and minuses are awarded to the upper and lower percentages of a grade (2% window).

Other Details

The following conventions will be followed in CS201. Please make sure that you fully understand each of the items listed below.

- Attendance is not required. However, I generally will not answer questions that I have already answered in class. Use the forum (see below) to ask questions about material you missed from being absent.
- There are no make-up exams (unless you have a written, university-authorized excuse). If you miss an exam, a zero will be recorded for your score.
- Please be in your seat by the beginning of class. There is to be no food or drink in the classroom. Sleep is allowed only in the event of a stultifyingly boring lecture.
- Signing up for and keeping current with the CS201 Forum at http://beastie.cs.ua.edu/forums is mandatory. Schedule changes, assignment clarifications, and such will be announced via the forum.
- Programming assignments are to be submitted electronically by 4:59:59 p.m. on the day they are due. For these projects, there is a grace period: late projects will be accepted up to 48 hours after the due date with no penalty. Please do not ask for an extension as I will most assuredly say no and then we’ll both end up feeling bad.
- Each assignment is to be done individually. There can be no sharing of code or solutions.
- The University of Alabama Code of Academic Conduct will be rigorously enforced. Please review the code at http://www.studenthandbook.ua.edu/academicintegritypolicies.html and be sure to follow this Code in all activities associated with CS 100. Any violations of this code will be handled as an Academic Misconduct case. The minimum punishment requested for plagiarism, or any other academic misconduct, is dismissal from the class with a failing grade. You are required to protect your work from plagiarism. If your work is plagiarized, it will be assumed that you were a willing participant in the plagiarism and you will receive the same punishment as the plagiarist, absent extraordinary evidence to the contrary.