CompSci 260P Winter 2020 Syllabus and Course Reference

Lecture: MW 6:30 - 7:50 PM, SH 174
Lab: F 7:00 - 7:50 PM, SH 174

You are responsible for all material and announcements covered in lecture and lab. If you miss one, ask a classmate to fill you in on what you missed. If you are not able to attend lectures or labs regularly, you may want to consider taking this course another quarter.

Your instructor is Michael Shindler, reachable by email at mikes at ics dot uci dot edu. Emails sent to course staff must be sent to this address, must come from your UCI (or ICS) email address, include your full name and ID number, and have a meaningful subject line that begins with the substring “CompSci 260P” -- emails that do not conform to this will probably not be read and do not count as having been sent for purposes of the course.

Piazza: We have a Piazza forum. Questions that do not require the attention of any particular member of course staff should be posted on Piazza with the appropriate privacy settings and labeled with the proper folder. Course staff will not reply to any anonymous posting on Piazza. It is expected that you treat your classmates and course staff respectfully when engaging with them. Abuse of Piazza may result in a revoking of Piazza privileges or referral to appropriate authorities.

Course announcements: On occasion, course announcements may be sent via email to all students enrolled in the class through the mechanism that Piazza allows. You should adjust email settings accordingly if you do not check Piazza regularly. You are considered to be aware of the announcement 24 hours after it has been sent.

Office Hours and associated locations will be announced shortly and will be posted in a place where students can find the information easily. In addition, the professor will typically be available after class for questions.

Office hours are a great time to ask questions about lecture material, the associated reading, and programming assignments. You may also stop by to introduce yourself if you’d like, even if you don’t have a question to ask. We can also talk about topics that aren’t this class, such as future classes, future plans, questions about course planning, life in industry, and so on.

Students with disabilities: Any students who feel that they may need an accommodation based on the impact of a disability should contact me privately to discuss these specific needs. Also, contact the Disability Services Center online or by phone at (949) 824-7494 as soon as possible to better ensure that such accommodations, such as alternative test-taking environments or note-taking services, can be arranged for you in a timely way.
**Commercial note-taking** Students are prohibited from selling (or being paid for taking) notes during this course to or by any person or commercial firm without the express written permission of the professor teaching this course. This includes, but is not limited to, a prohibition for providing notes, handouts, slides, assignment descriptions, or code to websites such as Chegg, Koofers, or CourseHero. Violations of this will be treated as a serious violation of the student code of conduct.

To ensure the free and open discussion of ideas electronic video and/or audio recording is not permitted during classroom lectures, discussion and/or activities unless the student obtains express written permission from the instructor. If permission is granted, any distribution of the recording is prohibited. Students with specific electronic recording accommodations authorized by the Office of Disability Services do not require instructor permission; however, the instructor must be notified of any such accommodation prior to recording. Any distribution of such recordings is prohibited.

We do not have an explicitly required textbook. However, having a textbook that would be appropriate for your undergraduate algorithms course is going to be useful.

I recommend the textbook *Algorithm Design and Applications* by Michael T. Goodrich and Roberto Tamassia

The book is available in hard copy from the usual sources and online at a much cheaper rate. There are other good textbooks available. Please see me if you have any questions.

**Grade calculation:**

- Three programming assignments 10% each for 30%
  - Plus a warm-up project. 2% one-time
- Four written problem set assignments 3% each, for 12% total
  - Plus a warm-up assignment 2% one-time
- Three mid-term exams 18% each, for 54% total
- Final Special - will be announced later

Programming assignment scores will be posted on Canvas. The score presented there does not include the lateness penalties. Written homework and mid-term exams will be posted on GradeScope.

The total points you earn from projects and homework assignments (scaled to 0 through 100) will be capped at 30 points above the mean of your exam scores (scaled to 0 through 100). For example, if you average 90 on the assignments, but your exam average is only 50, then your assignment average is treated as only 80 (because that's 30 more than 50). In other words, your assignment scores won't count fully if you can't show from your exam scores that you learned what you should have from the assignments.
Letter grades will be assigned based on the aforementioned relative weights. We will neither have a straight scale nor a straight curve. Among students who pass the final exam, it is guaranteed that 90% of the available points in the class will constitute at least an A-, although the cut-line for an A- may be lower than that. I anticipate that a large fraction of the class will do well.

The only factor in your grade is demonstrated knowledge in the class, and the only reconsideration requests granted are based on marking error. Requests for a grade bump based on other reasoning, such as scholarship requirements or academic eligibility, will not be considered. If you need a particular grade in this class, the time to consider that is early in the quarter. There is plenty of opportunity for help, practice, and credit during the semester. On a related note, there are no opportunities for extra credit.

**Grade reconsideration requests:** You will contact your instructor (not your TAs or readers!) in a prescribed manner and he will look into each matter. The cut-off time to submit a reconsideration request is 72 hours after the first attempt to return the artifact (programming assignment, problem set, exam) to you. Students who do not follow this procedure for reconsideration requests will be ineligible for any regrading, regardless of the merit of the request. Artifacts returned via GradeScope will use that regrade mechanism; otherwise, send an email to the professor with your request.

**Programming Assignment Policies**

**Late policy:** every programming assignment is due at 6:29PM Irvine time on the day listed. Late submissions are accepted for all except project 0. Every hour, or fraction thereof, an assignment is late being turned in reduces the grade by 1% of what the submission would have been worth had the student done so on time. There is a small and undisclosed grace period early in each hour that is considered part of the previous hour for purposes of submission penalties. Assignments two minutes late are probably not penalized, those fifteen minutes late almost certainly will be. Remember Murphy’s Law and plan to submit your work on time!

Project 0 must be turned in on time and no late submissions will be accepted for it. The largest late penalty, in terms of effect on your final grade in the class, will be forgiven. Remember that this only applies to programming assignments.

If there are extenuating circumstances related to your ability to submit one or more assignments on time, please see your instructor to discuss how to handle this. The sooner you contact me, the more I can do to help you.

**Grouping for projects:** you may select a partner for each project other than project 0, which must be done individually. A form will be provided on the course Piazza; both partners must fill out that form, including their partner’s UCI Net ID. If it is not the case that both partners register on-time, then it will be treated as if there is no partnership for that project.

It is expected that both partners contribute equally to the project and that the work is done via pair programming or similar; it is unacceptable to split the assignment duties. The instructor
reserves the right to question either partner about any work submitted, and to adjust grades accordingly if it is suspected that students did not both contribute to the full assignment. Furthermore, any academic dishonesty discovered on a project will be considered the responsibility of both partners.

When submitting an assignment as a group, exactly one partner should submit the assignment. In the event that neither submits it, the assignment has not been submitted. If both submit it, there will be a grade penalty assessed.

**Grading environment:** A copy of the CompSci 260P virtual machine will be made available to you. While you are not required to use it to develop your code, we will be using it for grading of assignments. We will not entertain regrade requests that ask us to evaluate student code on another machine, nor any that ask us to grade anything other than what was submitted. You are strongly encouraged to test your code on the virtual machine before considering your submission to be final.

Programming assignments that do not compile in our environment may receive scores as low as zero, even if minor changes to the code would have caused a much higher grade.

For various reasons, students are encouraged to backup the code they are writing as they progress. Version control software, such as Git, is strongly encouraged, but not required. Students who use Git for version control are reminded to keep their repositories private.

**Exam Rules:** All exams are expected to be individual effort. Students are not permitted to use notes, electronics of any form, or bring textbooks to the exam. On exam days, once students enter the classroom, they may not leave until explicitly dismissed by the instructor. Students for whom this presents a medical problem should consult the instructor as soon as possible and no later than one week before the exam.

Students are required to bring a legible photo ID issued by either the university or a government: examples include your UCI ID, a state driver's licence or ID, or a passport. The ID must be legible and include your name as it appears on the roster as well as a picture. If this will not be possible, you must contact your instructor as soon as you know. For exams after the first, if you do not write dark enough for the grader to read your work after it is scanned, they may treat it as a non-response.

Students may elect to leave the exam room after finishing their exam, provided at least 15 minutes remain. After that time has passed, all students with an exam are expected to remain seated in the exam room until after exams are collected.

Students will be assigned seating and must take the exam in that seat and the exam that has been assigned to that seat. Students may not open the exam booklet until explicitly told to do so by the instructor. When given the instruction to cease writing, students must immediately cease writing and close their exam booklet. It is prohibited to write any further at that point, including finishing one’s current sentence.

Failure to abide by these exam rules, or directions given by course staff during the exam, may result in disciplinary action, including but not limited to a failing grade in the class.
Missed Exam Policies
If you miss a non-final exam, for an exceptional, documentable circumstance, I will assign a grade for the missed exam based on your final exam score. Note that:

- If the circumstance is something that you would have known about in advance, I need to be informed in advance.
- A work conflict is not a valid excuse for missing a midterm. If you cannot arrange for time off from work to attend the class, you should not be taking the class.

If you miss the final exam and do not have a valid reason, you will receive a score of 0. The following policy applies if you miss the final exam for a valid reason:

- There are only two classes of valid reasons for missing the final exam:
  - An unforeseeable emergency, such as a medical emergency. In such cases, I will ask for documentation.
  - An absence from an exam due to a foreseeable circumstance that I have approved in advance.
- A work conflict is NOT a valid reason for missing the final exam. The exam times are announced at the beginning of the quarter, so there is time to plan your schedule.
- If you wait until after the exam to get a foreseeable excuse approved, and it is not approved, you will receive a grade of zero (0) on the exam.
- If I accept your reason for missing the final exam, at my option I may either (1) give you a makeup exam or (2) assign your grade on the basis of the remaining course work that you did not miss.
- If I give a makeup exam, I may give it less weight than announced in the syllabus.

Academic Integrity Rules
As students in CompSci 260P, you are expected to know and follow the academic integrity expectations of both the Bren School and of the University as a whole. Please take the time to review them at http://catalogue.uci.edu/appendix/#academichonestytext. In addition, please consult the Academic Honesty Guide on the next page of this syllabus for how to stay safe.

In general, it has been the experience of many professors that students who violate the academic integrity policy fell behind and had a moment of panic. If that describes you at some point this quarter, please go see Professor Shindler. We can find a way to get you back on track and find out why you’re falling behind.
CompSci 260P Academic Honesty Guide

For items we collect to grade, it is still important to be able to seek out helpful information, but it is clearly wrong to pass off work by others as your own. Navigating these two principles can be tricky, as it is possible to enter the danger zone between them unintentionally. To help guide you, follow this principle:

The “Kenny Loggins” Rule:
You may discuss high-level ideas, and give hints to other students regarding how to solve homework problems. Any time you seek help on, or discuss with someone else, a homework question that you have yet to solve, or any aspect of a programming assignment you have not yet finished, do not keep any written record of the discussion. Afterwards, take a 30-minute break and do something unrelated to the course (watching a 30-minute episode of your favorite cartoon show, for example). You may now return to your assignment.

This is less an ironclad rule as a guideline. It is a guideline to help you determine what is and is not appropriate collaboration and to avoid trouble from the “danger zone.” Flouting the spirit of the Rule while following its letter does not excuse cases of cheating which arise. For example, it is clearly not ok to study and memorize your friend’s solution, watch a cartoon for half an hour, and then write out your friend’s answer from memory and submit it. The spirit of the rule includes that what you write and submit for take-home assignments must reflect your work and your understanding at the time of submission. Do not submit anything that does not reflect your understanding of the material, no matter its origin.

You are responsible for understanding what is allowed, and what is not. It is possible to violate these guidelines without being malicious, and we still are required to report this to the Office of Academic Integrity & Student Conduct.

You should never:
- Show your take-home assignment to someone else, unless getting allowable help such as for compile-time or run-time errors.
- Write your solutions from notes taken outside of lecture or lab section.
  - This includes solutions from previous quarters, even if you were the author of those.
- Seek help on a required assignment from any source where not all respondents are subject to UC Irvine’s academic honesty policy.
  - This is especially true for websites like Chegg and CourseHero.
- Tell another student specifically how to solve part of a problem.
- Submit anything that you did not play an active role in creating.

If someone copies your work, both of you are culpable! Remember: friends that pressure you for unreasonable help are not really friends. Similarly, do not post your solutions in an open space online, even after the due date.

You should never need to get a solution elsewhere. There are plenty of allowable course resources to help you reach your own solution. Furthermore, academic dishonesty carries a penalty of F in the class, or potentially worse consequences; it isn’t worth the risk!

Lastly, if you do get substantial (but allowable) help from a classmate, cite their help on your submission by clearly stating “help from” and your classmate’s name. You must still follow other regulations here, but as long as you do these two things, you won’t get in trouble for reasonable cited help.
## CompSci 260P Winter 2020 Projected Schedule

Note that this is a *projected* schedule and is subject to change. All reading is in the textbook of Goodrich and Tamassia. All assignments, both programming and problem sets, are due at **6:29PM**. Late submissions are allowed for programming projects after (but not including) project 0, but not for problem sets.

Problem sets are typically due immediately prior to the start of the lecture, while programming assignments are typically due the day before or the day after a lecture.

<table>
<thead>
<tr>
<th>Week</th>
<th>Date</th>
<th>Topic</th>
<th>Due Dates</th>
<th>Reading</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Jan 6</td>
<td>Introduction,</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>Jan 8</td>
<td>Dynamic Programming: Subset Sum, LCS</td>
<td></td>
<td>Ch. 12; 12.5</td>
</tr>
<tr>
<td>2</td>
<td>Jan 13</td>
<td>Dynamic Programming: Edit Distance, Bellman Ford</td>
<td>PS0 due 1/13</td>
<td>14.3,</td>
</tr>
<tr>
<td></td>
<td>Jan 15</td>
<td>Introducing Complexity</td>
<td></td>
<td>Ch. 17</td>
</tr>
<tr>
<td>3</td>
<td>Jan 20</td>
<td>Martin Luther King Jr Day</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>Jan 22</td>
<td>Using Dynamic Programming</td>
<td>Proj 0 due 1/21</td>
<td></td>
</tr>
<tr>
<td>4</td>
<td>Jan 27</td>
<td>How to recognize complexity and how to use dynamic programming to cope with complexity</td>
<td>PS1 due 1/27</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Jan 29</td>
<td></td>
<td>PS2 due 2/3</td>
<td>18.3</td>
</tr>
<tr>
<td>5</td>
<td>Feb 3</td>
<td>Minimum Spanning Trees</td>
<td>Proj 1 due 2/4</td>
<td>Ch. 15</td>
</tr>
<tr>
<td>6</td>
<td>Feb 5</td>
<td>Exam 1</td>
<td></td>
<td></td>
</tr>
<tr>
<td>7</td>
<td>Feb 10</td>
<td>Greedy Algorithms</td>
<td></td>
<td>Ch. 10</td>
</tr>
<tr>
<td></td>
<td>Feb 12</td>
<td>Using Greedy Algorithms</td>
<td></td>
<td></td>
</tr>
<tr>
<td>8</td>
<td>Feb 17</td>
<td>Presidents' Day</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>Feb 19</td>
<td>Greedy algorithms for complexity</td>
<td>PS3 due 2/21</td>
<td>18.1, 18.2</td>
</tr>
<tr>
<td></td>
<td>Feb 24</td>
<td>Exam 2</td>
<td>Proj 2 due 2/23</td>
<td></td>
</tr>
<tr>
<td>9</td>
<td>Mar 2</td>
<td>Finish genetic programming</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>Mar 4</td>
<td>Linear Programming-based Techniques</td>
<td></td>
<td>Ch. 26</td>
</tr>
<tr>
<td>10</td>
<td>Mar 9</td>
<td>Linear Programming-based Techniques</td>
<td>PS4 due 3/9</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Mar 11</td>
<td>Exam 3</td>
<td>Proj 3 due 3/10</td>
<td></td>
</tr>
</tbody>
</table>

**Final Exam:** Fri, Mar 20, 7:00 - 9:00pm