**CSCI 311**

**Algorithms and Data Structures**

Spring 2018 Syllabus

<table>
<thead>
<tr>
<th>Lecture</th>
<th>MW 4:00-5:15 OCNL 124</th>
</tr>
</thead>
<tbody>
<tr>
<td>Activity Section 2</td>
<td>MW 12:00-12:50 OCNL 251</td>
</tr>
<tr>
<td>Activity Section 3</td>
<td>MW 1:00-1:50 OCNL 251</td>
</tr>
</tbody>
</table>

**Instructor: Dr. Judy Challinger**

<table>
<thead>
<tr>
<th>Office Hours</th>
<th>see my <a href="http://www.ecst.csuchico.edu/~jchallinger/">homepage</a> for current information</th>
</tr>
</thead>
<tbody>
<tr>
<td>Office Location</td>
<td>OCNL 226</td>
</tr>
<tr>
<td>Email</td>
<td><a href="mailto:jchallinger@csuchico.edu">jchallinger@csuchico.edu</a></td>
</tr>
<tr>
<td>Web</td>
<td><a href="http://www.ecst.csuchico.edu/~jchallinger">www.ecst.csuchico.edu/~jchallinger</a></td>
</tr>
</tbody>
</table>
Course Information

Prerequisite: CSCI 211, CSCI 217 or MATH 217, all with a grade of C or higher.

Required Textbook:

Introduction To Algorithms, Third Edition - aka "CLRS"
Thomas H. Cormen, Charles E. Leiserson, Ronald L. Rivest and Clifford Stein
The MIT Press
July 2009
ISBN: 9780262033848

On 2 hour reserve in Meriam Library.

- Errata!
- Solutions Manual

CLRS is one of the most highly regarded algorithms books ever published. It is complete and rigorous. It is the best selling algorithms book on Amazon with almost 5 stars. Many computing professionals keep this text as a reference book. Keep yours, at least for a while. This book is also required in CSCI 550!

Required - Access to ECC Server (jaguar): You will access this account via the lab machines or remotely at jaguar.csuchico.edu, using your portal credentials. This ECC Linux server is where your programming assignments will be graded. You are responsible for ensuring that your submissions compile and execute correctly on jaguar. All programming in this course is done in C++ using the GNU toolchain.

Catalog Description: This course focuses on object-oriented methodologies in designing and implementing a variety of data structures and algorithms. Coverage includes recursion, trees, search structures, hashing, heaps, sorting algorithms, and graph algorithms. Data structure and algorithm combinations will be studied and analyzed along with their relative merits using both mathematical and empirical measurements. The course includes a number of large programming assignments focusing on object-oriented software engineering and algorithm development. Students will be required to design, implement, test, and analyze their programs in at least one object-oriented language. 3 hours lecture, 2 hours activity.

Topics:

- growth of functions and asymptotic notation
- algorithm design and analysis, proofs of correctness
- object-oriented design and implementation
- algorithmic strategies, divide-and-conquer, greedy strategies
- sorting algorithms & properties
- hash tables and hash functions
- binary trees, red-black trees, B-trees
- graph data structures, breadth-first search, depth-first search
- minimum spanning trees, shortest paths
- selected advanced topics as time allows
Learning Outcomes: Upon completion of this course, students will be able to...

1. Explain the use of big $O$ (Omicron), big $\Omega$ (Omega), and big $\Theta$ (Theta) to describe the amount of work done by an algorithm; list and contrast standard complexity classes, and state the formal definition of big $O$.
2. Determine informally the time and space complexity of simple algorithms, using big $O$ notation to give the asymptotic upper bounds.
3. In the context of specific algorithms, identify the characteristics of data and/or other conditions or assumptions that lead to different behaviors, and explain what is meant by "best", "expected", and "worst" case behavior of an algorithm.
4. Implement search algorithms of varying time complexity and perform empirical studies to validate hypotheses about runtime complexity stemming from mathematical analysis by running algorithms on input of various sizes and comparing performance.
5. Use recurrence relations to describe, and recursion trees to visualize, the time complexity of recursively defined algorithms.
6. Describe and identify algorithmic strategies, including divide-and-conquer and greedy approaches, and determine an appropriate algorithmic approach to a problem.
7. Implement a hash table with an effective hashing function and collision resolution, and explain the importance of the hashing function on the time complexity of hash table operations.
8. Explain how tree structure affects the efficiency of various binary search tree operations and implement a balanced binary search tree such as a red-black tree.
9. Describe the heap property, min and max heaps, the use of heaps in the implementation of priority queues, and implement a min-priority queue based on a min heap for use in graph algorithms.
10. Represent directed and undirected graphs in a data structure and solve problems using graph algorithms, to include depth-first and breadth-first search, single-source shortest paths, and minimum spanning tree algorithms.

You are responsible for regularly checking the course schedule. The schedule will show lecture topics, reading assignments, programming assignments, due dates, and the dates of the examinations. The schedule is tentative, and subject to change. As the semester progresses, the schedule will be updated to reflect the material actually covered and/or changes in due dates.

Your attendance at all class meetings is required. Students with unexcused absences will be dropped from the course. See the catalog. If you are absent for any reason, it is your responsibility to communicate with your instructor. Coming to class gives you the opportunity to ask questions, do collaborative work, network with your peers, hear tips and hints about exams, etc. Habitual absences negatively impact your learning and your grades.

Come to class prepared. Do all assigned reading and/or exercises before coming to class. The schedule will make clear what you need to do. Discussions and activities during class will assume you have prepared. Coming to class unprepared will impede your success and detract from the learning experience of your peers. Expect to be called on by name to answer questions or give an explanation, example, or demonstration.

Stay focused. The number one predictor of success or failure in this course is "engaged attendance". During class, you are expected to be focused on the material from this course. Put away phones, use laptops and lab machines only for course-related tasks, and minimize distractions to yourself and the other students in the course. Students creating a distraction to themselves or others will be asked to leave the classroom.

Time Commitment: This is a 4 unit course. You should expect to spend at least 2 hours outside of class for each hour in class. So you should expect to spend at least 15 hours per week on this course reading, writing, coding, and attending class. This is a federal requirement! See the catalog. If you don't make the commitment to spending this time, you will not get much out of the course and may not earn a passing grade. So make yourself a weekly schedule and block out the study times and stick to it.
Grading: Your grade will be based on the following:

<table>
<thead>
<tr>
<th>Component</th>
<th>Percentage</th>
</tr>
</thead>
<tbody>
<tr>
<td>Midterm</td>
<td>15%</td>
</tr>
<tr>
<td>Comprehensive Final Exam</td>
<td>25%</td>
</tr>
<tr>
<td>Quizzes</td>
<td>10%</td>
</tr>
<tr>
<td>Activities</td>
<td>10%</td>
</tr>
<tr>
<td>Programming Assignments</td>
<td>40%</td>
</tr>
</tbody>
</table>

Quizzes: There will be quizzes of a variety of types, lengths, and formats. Quizzes may or may not be announced in advance.

Exams: There will be one midterm and a comprehensive final. No makeup exams will be given unless arranged in advance. Final exams are not returned to students, but students are welcome to come to office hours to view their exam during a subsequent semester (exams are kept for 3 years).

Activities: This includes preparation, class participation, and in-class activities. You are encouraged to do this work collaboratively. For the most part this work is not graded, but you will receive points for doing the work and correct solutions will be discussed in class. Some of this work will be due in class and some will be due by a specified date. No late submissions will be accepted.

Programming Assignments: These are larger assignments that you are expected to do primarily outside of class time, although some lab time will be devoted to helping students that need assistance with the programming assignments. Assignments are due by the date specified and late assignments will not be accepted. You are expected to work independently on all programming assignments. Failure to do so will result in disciplinary action for academic dishonesty. All programming assignments are to be done independently with no exceptions. Read the expectations for programming assignments. You must earn a C- or higher in this category (70% of points allocated to programming assignments) or you will receive no higher than a C- in the course. In other words, your final grade will be capped at a C- if you do not do passing work on the programming assignments.

Graded Work: It is your responsibility to examine graded work that has been returned to you. Any issues regarding your grade must be brought to Dr. Challinger within one week of the grade being posted. Questions regarding correct solutions, etc., are welcome any time of course.

Surveys: In order to facilitate continuous improvement and course redesign, there may be periodic surveys posted to the class by both the Office of the Chancellor and Dr. Challinger.

Final Grade Assignment: Your letter grade will be based on the percentage of the available points you earn in the course, as shown in the table below. You must earn 70% of points allocated to programming assignments or you will receive no higher than a C- in the course.

<table>
<thead>
<tr>
<th>Percent</th>
<th>Grade</th>
<th>University Definition</th>
</tr>
</thead>
<tbody>
<tr>
<td>[95-100]</td>
<td>A</td>
<td>Superior Work</td>
</tr>
<tr>
<td>[90-95]</td>
<td>A-</td>
<td></td>
</tr>
<tr>
<td>[87-90]</td>
<td>B+</td>
<td>Very Good Work</td>
</tr>
<tr>
<td>[83-87]</td>
<td>B</td>
<td></td>
</tr>
<tr>
<td>[80-83]</td>
<td>B-</td>
<td></td>
</tr>
</tbody>
</table>
### University Policies and Campus Resources:

**Dropping and Adding:** Dr. Challinger will adhere to the policies published by the University with respect to such things as adding and dropping courses, and grading policies. You should be aware of the deadlines, penalties, and limitations for adding and dropping classes. It is your responsibility to familiarize yourself with these policies. You will find them in the CSU Chico University Catalog under [Academic Policies and Regulations](http://www.ecst.csuchico.edu/~jchallinger/1718S-c...).

**Academic Integrity:** Students are expected to be familiar with the University's Academic Integrity Policy. Your own commitment to learning, as evidenced by your enrollment at California State University, Chico, and the University's Academic Integrity Policy requires you to be honest in all your academic course work. Faculty members are required to report all infractions to the Office of Student Judicial Affairs. It is your responsibility to understand the definition of academic integrity, and conduct yourself accordingly. More information is available from [Student Judicial Affairs](http://www.ecst.csuchico.edu/~jchallinger/1718S-c...). When in doubt, ask your Professor! Infractions of this nature will be vigorously prosecuted by Dr. Challinger and the College of Engineering, Computer Science, & Construction Management. **Cheating on exams or assignments will result in an F in the course and referral to Student Judicial Affairs.**

**Americans with Disabilities Act:** If you need course adaptations or accommodations because of a disability or chronic illness, or if you need to make special arrangements in case the building must be evacuated, please contact the [Accessibility Resource Center](http://www.ecst.csuchico.edu/~jchallinger/1718S-c...) (ARC) as they are the designated department responsible for approving and coordinating reasonable accommodations and services for students with disabilities. ARC will help you understand your rights and responsibilities under the Americans with Disabilities Act and provide you further assistance with requesting and arranging accommodations.

**Accessibility Resource Center**
530-898-5959
Student Services Center 170
arcdept@csuchico.edu