CS 300 Data Structures
Syllabus - Fall 2018

Catalog Description

Data structures are fundamental to advanced, efficient programming. Topics including asymptotic analysis, stacks, queues, linked lists, trees, hash tables, searching and sorting will be covered in discussions centering around more sophisticated programming concepts and problem solving techniques. Prerequisite: CS 250 with a minimum grade of C. 4 credits.

Course Student Learning Outcomes
- Write programs that use each of the following data structures: arrays, records/structs, strings, linked lists, stacks, queues, sets, trees, and hash tables.
- Compare alternative implementations of data structures with respect to performance.
- Describe how references allow for objects to be accessed in multiple ways.
- Compare and contrast the costs and benefits of dynamic and static data structure implementations.
- Choose the appropriate data structure for modeling a given problem.
- Explain what is meant by "best", "expected", and "worst" case behavior of an algorithm.
- Determine informally the time and space complexity of simple algorithms.
- State the formal definition of big O.
- List and contrast standard complexity classes.
- Design and implement code with reusability in mind.
- Understand where and when recursion can efficiently and effectively be used.
- Identify the relative strengths and weaknesses among multiple designs or implementations for a problem.

Major Student Learning Outcomes
- Demonstrate a fundamental understanding of computation and programming.
- Apply strategies for abstract problem solving
- Discuss the theoretical basis of the mathematics and symbolic concepts that underlie computing.

Topics
- Asymptotic analysis of upper and average complexity bounds
- Identifying differences among best, average, and worst case behaviors
- Big O / Standard complexity classes / Time and space tradeoffs in algorithms
- Arrays
- Data representation in memory
- Static, stack, and heap allocation
- Runtime storage management
- Pointers and references
- Linked structures
- Implementation strategies for stacks, queues, and hash tables
- Implementation strategies for trees
- Strategies for choosing the right data structure
- The concept of recursion / Implementation of recursion
- Simple numerical algorithms
- Sequential and binary search algorithms

- Quadratic sorting algorithms (selection, insertion)
- Hash tables, including collision-avoidance strategies
- Binary search trees
- Debugging Strategies
- Design for reuse
- Activation records and storage management
- Use of open-source materials
- Tools for source control and their use in particular in team-work

Many of the above topics were copied with permission from the Computing Curricula 2013 recommendations found at: http://www.acm.org/education/CS2013-final-report.pdf

Tools
- Eclipse IDE
- gcc / gdb
- Subversion / Subclipse
- Valgrind
- GNU Make

Language(s)
- C

Operating System(s)
- Linux

Instructor Details

Professor:	Chadd Williams
Email:	chadd@pacificu.edu
Office:	Strain 202
Office Hours:	MWF 11-noon, TTh 1-2pm

Course Details

Course Title:	CS300-01 Data Structures
Prerequisite:	CS 250 Introduction to Computer Science II with a minimum grade of C.
Required For:	A minimum grade of C in CS300 is required for CS380 Algorithms, CS445 Databases, CS460 Operating Systems
Meeting Times:	MWF 9:15am - 10:20am
Location:	Scott 204
Required Textbook	Data Structures in C by Noel Kalicharan; ISBN 9781438253275

Course Website

http://zeus.cs.pacificu.edu/chadd/cs300f18

Course Assessment

Grade Distribution :

Programming Assignments	40%
unscheduled (open note) quizzes	5%
3 Exams	35%
Final	20%

Grading Breakdown:

		92-100%	A	90-92%	A-
88-90%	B+	82-88%	B	80-82%	B-
78-80%	C+	72-78%	C	70-72%	C-
68-70%	D+	60-68%	D		
		0-60%	F		

Program Grading:

Successful execution	50%
Design to include proper use of functions, modules, and makefiles	30%
Acceptable structure, style, documentation, and efficiency. You must follow the C Coding Standards.	20%

Important Dates
Tentative dates for Exams:
Exam 1:   Friday, September 21, 2018
Exam 2:   Friday, October 19, 2018
Exam 3:   Friday, November 16, 2018

Labor Day Holiday:
Monday, September 3, 2018 (No Classes)

Midsemester Break:
Friday, October 5, 2018 (No Classes for Arts & Sciences)

Thanksgiving Break:
Wednesday, November 21, 2018 through Sunday, November 25, 2018

Date of Final:
Tuesday, December 11, 2018, 3:00 pm - 5:30 pm

Academic Calendar:
http://www.pacificu.edu/as/calendar/

Course Policies
   Attendance: Attendance at every class is critical to your success in this course. I expect you to be on time and ready to go once it is 9:15am and that you stay until the end of class. Any missed lecture is your responsibility to make up; just remember that if you fall behind, it may be very difficult for you to catch up.
   • I reserve the right to raise or lower your grade based on class participation and attendance. Specifically, I may lower your grade or may officially withdraw you from the course through the tenth week of the semester for poor attendance or participation. Further, your final grade may

be lowered by 1/3 of your final course grade for each day (or portion thereof) of class missed. Please notify me PRIOR to class if you must miss class for any reason.

- No early or late exams/final will be given. No incompletes will be given.

Programming Assignments: All assignments are to be programmed in C using Eclipse, gcc 7.3 or later, GNU Make 4.2 or later, and will be tested on zeus (a 64-bit Linux server). Both the electronic copy and hardcopy of your assignments are due by 9:15am on the day that the assignment is due.

- The hardcopy must be placed on the instructor's desk before 9:15am on the day the assignment is due. If the hardcopy uses more than one sheet of paper, then all sheets must be stapled in the upper-left corner. The code must be printed in color. Failure to submit a hardcopy of the assignment will result in a loss of 30% of the assignment points.

- Make sure to test your program extensively on zeus before you turn it in. The lab machines in Strain 222 are similar, but not identical to zeus. A program that does not successfully compile or produces no output loses 50% of the assignment grade right off the top. The rest of the design and coding standards will be graded accordingly.

- Assignments can be turned in up to 24 hours late with a penalty of 10% of the grade. Anything turned in later than 24 hours of the assignment deadline will NOT be accepted.

- Make sure that you test your programs before submitting them. You may only submit your assignment solution once using the course submit script. I will discuss more about this in class.

- Neither computer failure, software failure, nor lack of computer access are accepted as excuses for late programs; therefore, start work on the programs as soon as they are assigned, and don't put them off until the last minute. Further, corruption of programs due to bad disk media is also not accepted as an excuse for late programs; therefore, always keep a current backup of all programs on a separate disk. Please note that the Computer Science departmental servers are not backed up.

Academic Dishonesty: Pacific University has no tolerance for academic misconduct/dishonesty. It is university policy that all acts of misconduct and dishonesty be reported to the Associate Dean for Student Academic Affairs. Sanctions that may be imposed for such misconduct range from an "F" for the assignment, an "F" for the course, and suspension or dismissal from the university. Forms of academic misconduct include but are not limited to plagiarism, fabrication, cheating, tampering with grades, forging signatures, and using electronic information resources in violation of acceptable use policies.

- For programming assignments, plagiarism takes the form of, *but is not limited to* copying code from someone else, whether copying files, glancing at someone else's code, typing from someone else's notes, typing from someone's description of a solution (written or verbal) or typing while they dictate. The source can be a classmate, former student, website, program listing found in the trash, or anything else. Furthermore, plagiarism even on a small part of the program is cheating.

- You should also note that aiding someone else's cheating also constitutes cheating. You should never leave your code where someone else could have access to it, such as staying logged onto a machine or placing solutions in the recycling bin where another student may take it.

- Sanctions that may be imposed for academic dishonesty are:
    - First offense for cheating: 12% subtracted from your final course grade
    - Second offense for cheating of any kind: `F' in the course

Other:

- You may be asked to leave the class if you are causing a distraction e.g. cell phone ringing, talking, etc.

- If you have a complaint regarding a grade on an assignment or exam, write a one paragraph description of why you feel the grade is incorrect and deliver it to the instructor within five working days (M-F are considered working days) of when the graded material was returned to you. I will not consider any grade changes later than five working days after the graded material was returned.

Learning Support Services for Students with Disabilities: If you have documented challenges that will impede your learning in any way, please contact our LSS office in Clark Hall (ext.2717; lss@pacificu.edu). LSS staff will meet with students, review the documentation of their disabilities, and discuss the services that Pacific offers and any appropriate ADA accommodations for specific courses.

Tutoring: The TLC is located in Scott Hall, 1st-floor. The center focuses on delivering one-on-one and group tutoring services for foreign languages, math and science courses and writing skills in all subjects. Students should consult with the center's director for information on tutoring available for other subjects. Day and evening hours; walk-ins welcome!

Unauthorized Recordings: Students are prohibited from making audio and/or visual recordings of lectures or presentations without prior consent of the instructor or presenter.