CSE 15/15L
Introduction to Data Structures
Fall 2019

15 Description: Teaches students to implement common data structures and the algorithms associated with each data structure, through progressively difficult exercises. Topics include big "O" notation; pointers, recursion (induction), and dynamic allocation; linked lists and list processing; stacks, queues, binary trees and binary search trees; simple sorting techniques and simple search techniques. Students will gain a working knowledge of the elements of the Java and C programming languages. Prior experience with Unix is assumed. (Formerly CMPS 12B.)

15L Description: Complements 15, gaining additional competence with a number of important software development tools, languages, and techniques. Included are advanced Unix features and utilities such as grep, find, diff, the shell, and pipes; C programs utilizing I/O, arrays, pointers, and structures; a scripting language to perform simple text and file manipulation; and the make utility. (Formerly CMPS 12M.)

Time and Place: MW 7:10pm-8:45pm   Media Theater M110
Class Webpage: https://classes.soe.ucsc.edu/cse015/Fall19/

Instructor: Patrick Tantalo   http://users.soe.ucsc.edu/~ptantalo/
Office: E2 239A
Office Hours: MWF 12:00-2:00pm
Email: ptantalo@soe.ucsc.edu

Teaching Assistants:
Ishani Chakraborty   (ischakra@ucsc.edu)
Jayanth Yetukuri   (jyetukur@ucsc.edu)
Alan Peral Ortiz   (alan.peralortiz@ucsc.edu)
Zekun Zhao   (zzhao99@ucsc.edu)

MSI Learning Assistants:
Darren Eam   (deam@ucsc.edu)
Kevin Zhang   (kszhang@ucsc.edu)

LSS Small Group Tutor:
Adarsh Sekar   (adsekar@ucsc.edu)

Required Texts:
[ODS] *Open Data Structures: An open content textbook* (pseudo-code edition) by Pat Morin.
     main: https://opendatastructures.org/
     pdf: https://opendatastructures.org/ods-python.pdf
     html: https://opendatastructures.org/ods-python/

[DAPS] *Data Abstraction & Problem Solving with C++: Walls and Mirrors* (6th Edition)
     by Frank M. Carrano and Timothy M. Henry. Pearson 2013 (ISBN 978-0-13-292372-9)

Supplementary Texts:
C in a Nutshell (2nd Edition) by Tony Crawford and Peter Prinz. O'Reilly 2015
     https://proquest-safaribooksonline-com.oca.ucsc.edu/book/programming/c/9781491924174
Practical C Programming (3rd Edition) by Steve Oualline. O'Reilly 1997
     https://proquest-safaribooksonline-com.oca.ucsc.edu/book/programming/c/1565923065

Understanding and Using C Pointers by Richard M Reese. O'Reilly 2013
    https://proquest-safaribooksonline-com.oca.ucsc.edu/book/programming/c/9781449344535
Your Unix (2nd edition) Sumitabha Das. McGraw-Hill 2006 (ISBN 978007250422).

Coursework

Although 15 and 15L are separate courses, it is the policy of this class that students will receive the same grade in both. The following weights will determine this grade.

- 10% Lab Assignments: Seven assignments due at roughly 7 day intervals
- 30% Programming Assignments: Five assignments due at roughly 8 day intervals
- 15% Midterm Exam 1: Wednesday October 23, 7:10-8:15pm (lecture to follow)
- 15% Midterm Exam 2: Wednesday November 20, 7:10-8:15pm (lecture to follow)
- 30% Final Exam: Monday December 9, **8:00–10:00pm**

CSE 15 and CSE 15L are co-requisite courses and special permission is required to take only one. If you believe you need this permission, see me in office hours to explain why. In such a case, your grade will be calculated as follows.

15 only:
You need not complete the Lab Assignments (though they are recommended). Programming Assignments will be 40% of your grade, and exams will be weighted as above.

15L only: You need not complete the Programming Assignments (though they are recommended), and you need not sit for Midterms 1 and 2. Lab Assignments will be 70% of your grade, and the Final Exam will be 30%.

In addition, it is required that students earn a passing grade in both the Programming-Lab, and Exam portions of this course in order to receive a grade of C or better. Specifically, if either the combined Programming-Lab average (weighted as above), or the combined exam average (weighted as above), is not passing ($\geq 70\%$), then the student's maximum possible grade will be C-. All scores (Lab, Program, Midterm 1, Midterm 2, Final and Overall) will be rounded to the nearest 10th of a percent. They will not be rounded further. No scores in this class are curved.

Grading scale:

A+	97.0% - 100%
A	93.0% - 96.9%
A-	90.0% - 92.9%
B+	87.0% - 89.9%
B	83.0% - 86.9%
B-	80.0% - 82.9%
C+	76.0% - 79.9%
C	70.0% - 75.9%
C-	67.0% - 69.9%
D+	64.0% - 66.9%
D	61.0% - 63.9%
D-	58.0% - 60.9%
F	0% - 57.9%

Accommodations for Students with Disabilities

UC Santa Cruz is committed to creating an academic environment that supports its diverse student body. If you are a student with a disability who requires accommodations to achieve equal access in this course,

please submit your Accommodation Authorization Letter from the Disability Resource Center (DRC) to me privately during my office hours (or by appointment), preferably within the first two weeks of the quarter. At that meeting, we will discuss how to best ensure your full participation in the course. I encourage all students who may benefit from DRC services, or who wish to just learn more about those services, to contact DRC by phone at 831-459-2089, or by email at drc@ucsc.edu. See also https://drc.ucsc.edu/.

Academic Honesty:
The Baskin School of Engineering has a zero tolerance policy for any incident of academic misconduct.   If cheating occurs, consequences may range from getting zero on a particular assignment to failing the course. In addition every case of academic misconduct is referred to the students' college Provost, who sets in motion an official disciplinary process. Cheating in any part of the course may lead to failing the course, suspension or dismissal from the Baskin School of Engineering, or from UCSC.

What is cheating? In short, it is presenting someone else's work as your own. Examples include copying another students' lab assignment, programming assignment, or exam solution; allowing your own work to be copied; or in any way facilitating misconduct by others. You may discuss lab and programming projects with fellow students, but your collaboration must be at the level of *ideas* only. You may freely give and receive help on the UCSC computer facilities, code editors and IDEs, the UNIX operating system, and on the proper use and syntax of the Java and C programming languages. You may also freely use any *example code* that is posted by me on the class webpage. However, you may not *copy*, *paste*, *email*, *transfer* or *share* in any way the *source code* for projects in this class.

Go to https://www.ue.ucsc.edu/academic_misconduct to see the University's official policy on Academic Misconduct.