CMPS 101
Algorithms and Abstract Data Types
Fall 2017

Description: Studies basic algorithms and their relationships to common abstract data types. Covers the notions of abstract data types and the distinction between an abstract data type and an implementation of that data type. The complexity analysis of common algorithms using asymptotic (big O) notation is emphasized. Topics include sorting and searching techniques, basic graph algorithms, and algorithm design techniques. Abstract data types covered include priority queues, dictionaries, disjoint sets, heaps, balanced trees, and hashing. Familiarity with C, Java, and Unix is assumed.
Prerequisites: CMPS 12B or 13H; and CMPE 16 or 16H; and MATH 19B; and one course from the following: MATH 21, 22, 23A, 24 or AMS 27.

Time and Place: TTh 5:20-6:55 pm Classroom Unit 002
Class Webpage: https://classes.soe.ucsc.edu/cmps101/Fall17/

Instructor: Patrick Tantalo http://users.soe.ucsc.edu/~ptantalo/
Office: E2 255
Office Hours: TTh 1:00-3:00, W 10:00-12:00, or by appointment
Email: ptantalo@soe.ucsc.edu

Teaching Assistants:
Ryan Compton (rcompton@ucsc.edu)
Ankit Gupta (agupta29@ucsc.edu)
Wen Cui (wcui7@ucsc.edu)
Garvit Mantri (gmantri@ucsc.edu)
Shubhangi Tandon (shtandon@ucsc.edu)
Isaak Cherdak (icherdak@ucsc.edu)

MSI Tutors:
Elisabeth Oliver (elaolive@ucsc.edu)
Hunter Bingham (hbingham@ucsc.edu)

Required Text: *Introduction to* Algorithms (2nd or 3rd edition) by Cormen, Leiserson, Rivest and Stein (CLRS). McGraw-Hill 2001 (ISBN 9780262033848). The following reading schedule is a rough guide to what we will discuss and when. Section numbers are from the 3nd edition. I expect that the material from appendices A.1-A.2, B.1-B.3, and C.1-C.2 is already familiar.

Week	*Sections*	*Topics*
1	1.1-1.2, handouts	ADTs, Analysis of Algorithms
2	2.1-2.3, 3.1-3.2, handouts	Asymptotic Growth Rates
3	4.3-4.5, handouts	Induction Proofs, Recurrences
4	B4, B.5 handouts	Graphs, Trees
5	22,1-22,5	Graph Representations, BFS, DFS
6	6.1-6.5	Heaps, Heapsort, Priority Queues
7	21.1-21.3, 23.1-23.2	Disjoint Sets, Minimum Weight Spanning Trees
8	24.1, 24.3	SSSP Problem, Bellman-Ford and Dijkstra's Algorithms
9	12.1-12.3, 13.1-13.4	Binary Search Trees, Red-Black Trees
10	7.1-7.4, 8.1-8.4	Sorting Algorithms

Coursework:

- Homework will consist of written exercises, mostly from CLRS, due at beginning of class on Thursdays.
- Programming Assignments will be due at roughly 10 day intervals.
- Midterm Exam 1 will be held Tuesday October 24
- Midterm Exam 2 will be held Tuesday November 21
- Final Exam will be held Friday December 15, 8:00 – 10:00 am

Coursework will be weighted as follows:

Written Homework	5%
Programming Assignments	25%
Midterm Exam 1	10%
Midterm Exam 2	20%
Final Exam	40%

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%

Letter grade boundaries may be lowered at my discretion in order to eliminate some borderline cases.

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 this time, I would also like us to discuss ways we can ensure your full participation in the course. I encourage all students who may benefit from learning more about DRC services to contact DRC by phone at 831-459-2089, or by email at drc@ucsc.edu.

Academic Honesty:

The Baskin School of Engineering has a zero tolerance policy for any incident of academic dishonesty. If cheating occurs, consequences may range from getting zero on a particular assignment to failing the course. In addition every case of academic dishonesty 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 would include copying another students' programming assignment, written homework or exam, or allowing your own work to be copied. You may discuss programs with fellow students, but your collaboration must be at the level of *ideas* only. You may freely give and receive help with the computer facilities, editors, the UNIX

operating system, and the proper use and syntax of the Java and C programming languages; but you may not *copy*, *paste*, *email*, *transfer* or in any way share *source code*, or *share in the act of writing your solutions*. If you do collaborate (legitimately) or receive help from anyone, you must credit them by placing their name(s) at the top of your program or paper. Please see the University's policy on Academic Misconduct at: https://www.ue.ucsc.edu/academic_misconduct.