CMPS 101
Algorithms and Abstract Data Types
Fall 2016

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: MWF 10:40 – 11:45 am Media Theater M110
Class Webpage: https://classes.soe.ucsc.edu/cmps101/Fall16-01/

Instructor: Patrick Tantalo http://users.soe.ucsc.edu/~ptantalo/
Office: E2 257
Office Hours: MW 4:30-6:30, F 4:30-5:30, or by appointment
Email: ptantalo@soe.ucsc.edu
Phone: 831-459-3898

Teaching Assistants:
Vedashree Bagade (vbagade@ucsc.edu)
Kuan-Sung Huang (kuhuang@ucsc.edu)
Noujan Pashanasangi (npashana@ucsc.edu)

MSI Tutor:
Emily Hockel (echockel@ucsc.edu)

Required Text: *Introduction to* Algorithms (3rd edition) by Cormen, Leiserson, Rivest and Stein (CLRS). MIT Press 2009 (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 and Evaluation:
- **Homework:** Will consist of written assignments taken from the exercises in CLRS. Due at beginning of class on Mondays.
- **Programming Assignments:** Due at roughly 10 day intervals.

- **Midterm Exam 1:** Wednesday October 19
- **Midterm Exam 2:** Wednesday November 16
- **Final Exam:** Tuesday December 6, 4:00 – 7:00 pm

Coursework will be weighted as follows:

Written Homework	5%
Programming Assignments	35%
Midterm Exam 1	10%
Midterm Exam 2	20%
Final Exam	30%

Grading scale:

A+	97%-100%
A	93%-96%
A-	90%-92%
B+	87%-89%
B	83%-86%
B-	80%-82%
C+	76%-79%
C	70%-75%
C-	67%-69%
D+	64%-66%
D	61%-63%
D-	58%-60%
F	0%-57%

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 qualify for classroom accommodations because of a disability, please contact the Disability Resource Center (DRC) to get an Accommodation Authorization. Submit the authorization form to me in person during office hours, or by appointment, preferably within the first two weeks of the quarter. All students who may benefit from learning more about DRC services are encouraged 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' written homework or programming assignment, or allowing your own work to be copied. You may discuss your work 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*. See the official University policy on Academic Misconduct at https://www.ue.ucsc.edu/academic_misconduct