

## CMPS 12B/M

### Introduction to Data Structures

### Summer 2014 (June 23 – August 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.

**Prerequisites:** CMPS 11 or 12A or CMPE 13. Concurrent enrollment in CMPS 12M is required.

**Time and Place:** TTh 10:00 – 12:30 Thimann 3

**Class Webpage:** <http://ic.ucsc.edu/~ptantalo/cmeps12b/Summer14/>

**Instructor:** Patrick Tantalo <http://users.soe.ucsc.edu/~ptantalo/>

**Office:** E2 257

**Office Hours:** TTh 2:00-4:00, W 3:30-4:30, or by appointment

**Email:** [ptantalo@soe.ucsc.edu](mailto:ptantalo@soe.ucsc.edu)

**Phone:** 831-459-3898

#### Teaching Assistants:

Larissa Munishkina ([mlarissa@soe.ucsc.edu](mailto:mlarissa@soe.ucsc.edu))

Brad Hollister ([behollis@soe.ucsc.edu](mailto:behollis@soe.ucsc.edu))

#### LSS Tutor:

Zachary Drumbor ([zdrumbor@ucsc.edu](mailto:zdrumbor@ucsc.edu))

**Required Text:** *Data Abstraction and Problem Solving with JAVA* (2<sup>nd</sup> edition) by Frank M. Carrano and Janet J. Prichard. Addison Wesley 2005 (ISBN 9780321304285). We will cover (roughly) the following sections: 3.1-3.5, 4.1-4.3, 5.1-5.3, 7.1-7.6, 8.1-8.5, 9.1-9.3, 11.1-11.4, 13.2.

#### Supplementary Texts:

*C for Java Programmers: A Primer* Charlie McDowell. Lulu.com 2007.

*Your Unix* (2<sup>nd</sup> edition) Sumitabha Das. McGraw-Hill 2006 (ISBN 978007250422).

*Java By Dissection* Ira Pohl and Charlie McDowell. Lulu.com 2006.

#### Coursework and Evaluation for CMPS 12B:

- **Programming Assignments:** Five assignments due at roughly 10 day intervals.
- **Midterm Exam 1:** Thursday July 10
- **Midterm Exam 2:** Thursday July 31
- **Final Exam:** Thursday August 14, 10:00 – 1:00

Coursework for 12B will be weighted as follows:

Programming Assignments	60%
Midterm Exam 1	10%
Midterm Exam 2	10%
Final Exam	20%

## Coursework and Evaluation for CMPS 12M:

- **Lab Assignments:** Seven assignments due at roughly one week intervals. Topics covered will include: makefiles, executable jar files, command line arguments, file input and output, Java generics, introduction to the C language, data abstraction and information hiding in C. Some of these assignments will build on the 12B programming projects.
- **Final Exam:** The 12B final exam (Thursday August 14, 10:00 – 1:00) will also figure in the grade for 12M.

Coursework for 12M will be weighted as follows:

Lab Assignments	80%
Final Exam	20%

Grading scale for both 12B and 12M:

A+	97%-100%
A	93%-96%
A-	90%-92%
B+	87%-89%
B	83%-86%
B-	80%-82%
C+	76%-79%
C	70%-75%
D	60%-69%
F	0%-59%

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

## Accommodations for Students with Disabilities

If you qualify for classroom accommodations because of a disability, please submit your Accommodation Authorization Letter from the Disability Resource Center (DRC) to me during my office hours or by appointment, preferably within the first week of the Summer Session. Contact DRC by phone at 831-459-2089 or [drc@ucsc.edu](mailto:drc@ucsc.edu) for more information.

## 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' lab or programming assignment, 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 C and Java programming languages; but you may not *copy*, *paste*, *email*, *transfer* or in any way share *source code*. 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. Please go to [http://www.ucsc.edu/academics/academic\\_integrity/](http://www.ucsc.edu/academics/academic_integrity/) to see the full text of the University's policy on Academic Integrity.