

CS 260: Data Structures

Course Syllabus

Spring 2007

Professor: Richard Croft, Ph.D. Badgely Hall 109: 962-3695
e-mail: rcroft@eou.edu
web: www2.eou.edu/~rcroft

Office Hours: MWF 1:00 to 1:50
TTh 11:00 to 12:30
and by appointment

Note that I may be in BH 123 or LH 235 during office hours.

Catalog Description

This class introduces the analysis of algorithms and data structures commonly used in computer science, and the selection and design of data structures for the solution of specific problems.

Prerequisites

CS 162, Foundations of Computer Science II.

Outcomes

Upon completion of this course, students will:

- Define various basic data structures;
- Describe several applications for various data structures;
- Select data structures for specific situations;
- Describe several different ways to implement data structures;
- Write programs that support various basic data structures; and
- Describe the performance of algorithms that operate on data structures, particularly best-case and worst-case performance.

Textbook

Venugopal, S. (2005). *Data structures outside in with java*. Upper Saddle River, NJ: Pearson. ISBN: 0-13-198619-8. (Required)

Course Activities and Means of Assessment

Theoretical content for this class will derive from lectures and discussions in class, combined with assigned readings from the text. Lectures will provide a less formal perspective on material covered in the book. Activities will also include written exercises and programming assignments to provide opportunities to apply theoretical knowledge. Occasional pop quizzes will provide students with an understanding of how well they are learning the material. A final exam will indicate students' overall understanding of course concepts.

Policies

Attendance in this class is important. The concepts covered in the text will be further explained in class, and revisions to assignments may be announced in class. Your contributions to class discussions will help you and your classmates better understand material.

Note-Taking is important in any class, but computer science classes details are critical. Many concepts will be introduced in lecture that are not explained in the text. Your translation of class lectures will be essential for your success. Take thorough notes, and review them (preferably with another class member) frequently.

Assignments: The assignment sheet for each project will include its due date. Programs are due at the beginning of class on the due date. Late work will be penalized 20 percent for each class day it is late.

Except for code provided to all class members as part of an assignment, **all** work must be your own. **No** code may be “borrowed” from other sources, including sample solutions posted for previous terms. Failure to heed this rule will be treated as a violation of EOU’s rules concerning academic misconduct (see below).

Save all returned assignments.

Programming Style , Documentation & Formatting: Your programs should adhere to good software engineering principles. Document appropriately, choose sensible identifiers, format code for legibility, and divide code into logical procedures. Good programming practice will be the topic of a discussion early in the course. Programs that do not adhere to the standards established at that time will suffer a penalty of up to fifty percent.

Occasional **quizzes** over lectures, homework problems, and readings will provide feedback to let you determine if you are assimilating enough detail in course topics. You may drop one quiz grade. Quizzes may be made up only if I deem the documented excuse valid.

If you have any questions, comments, concerns, or suggestions, please feel free to write them on a slip of paper and leave it on the lectern (or hand it to me) when the class breaks. Your feedback may help improve the course.

Academic Misconduct

Eastern Oregon University places a high value upon the integrity of its student scholars. Any student found guilty of an act of academic misconduct (including, but not limited to, cheating, plagiarism, or theft of an examination or supplies) may be subject to having his or her grade reduced in the course in question, being placed on probation or suspended from the university, or being expelled from the university—or a combination of these. Please see Section II of the *2002-2003 Student Handbook and Planning Calendar: Campus Citizenship (Academic)*, p. 32ff; *Campus Citizenship (Behavior)*, p.41ff.

Students with Disabilities

If you have a documented disability or suspect that you have a learning problem and need reasonable accommodations, please contact the Disability Services Program in Loso Hall 234 (telephone 962-3081) **before** the end of the second week of classes.

Grading

Your final grade for this course will depend on your completion of the assigned homework and programs, quizzes, and a midterm and final exams. All activities will measure your ability to apply the concepts introduced in the text and class lectures. Distribution of credit is as follows:

Programming Problems:	40 percent
Exercises	5 percent
Quizzes	20 percent
Midterm exam:	15 percent
Final Exam:	20 percent

Grade cutoffs will be no higher than 92 for A, 84 for B, 75 for C and 65 for D, but *may be lower* if statistical analysis of the distribution of scores indicates they should be.

Course Outline (Tentative*)

Week	Topics	Reading**
1.	Course intro; Java Review; Good programming practices; General information about data structures;	Syllabus; 1.1, 1.2
2	Abstract data types; Data and operations; ADT List Algorithm Analysis	Ch. 2 3.5, 3.6
3	Array lists, Linked lists, Cursor Lists	4.1–4.5 8.1
4	ADT Stack and implementations	7.1, 7.2
5	ADT Queue and implementations	6.1
6	General Trees and implementations; Operations on Binary Trees	9.1–9.2 9.8
7.	ADT Binary Search Tree	10.1–10.3
8	Balancing Trees	10.7
9	More on balancing trees	
10	Course review and special topics	TBA
11	FINAL EXAM: See university time table.	

*Schedule is subject to change based on class dynamics.