



CSE 332

Data Abstractions

Winter 2011

Instructor: Ruth Anderson
Email: rea at cs.washington.edu
Office Location: Allen Center (CSE) 360
Office Hours: M & W 3:30-4:20 or by appointment

Teaching Assistants: (Office Hours time and location TBA)

Sandra Fan (sbfan at cs.washington.edu)
 Nathan Armstrong (armstnp at cs.washington.edu)
 Gloria Guo (jiayung at cs.washington.edu)
 Timothy Jang (jangt13 at cs.washington.edu)

Lecture: MWF 2:30-3:20 SMI 205
Sections: AA Th 1:30-2:20 SAV 168
 AB Th 2:30-3:20 SAV 168
 AC Th 12:30-1:20 MGH 241

Note: Sections **will** meet the first week (Thurs January 6th)

Course Web Page: <http://www.cs.washington.edu/332/>

Catalog Description: Covers abstract data types and structures including dictionaries, balanced trees, hash tables, priority queues, and graphs; sorting; asymptotic analysis; fundamental graph algorithms including graph search, shortest path, and minimum spanning trees; concurrency and synchronization; and parallelism.

Prerequisites: either CSE 311 or CSE 321

Course Text(s): Main text: Weiss, Mark Allen. **Data Structures and Algorithm Analysis in Java** 2nd Ed., Addison Wesley: 2007, ISBN: 0-321-37013-9. Recommended: Core Java(TM), Volume I--Fundamentals 8th Edition, Cay S. Horstmann and Gary Cornell, Prentice Hall: 2007, ISBN: 0-132-35476-4, is recommended as a Java programming reference. (Note this book is also recommended for CSE 331.) We will use a set of on-line notes for the material on parallelism and concurrency.

Assignments: Assignments will be a mix of written/typed exercises (eight) and programming projects (three). Written assignments will be due at the BEGINNING of lecture on the day they are due. Programming assignments will be submitted electronically via the web. Exact dates and deadlines will be specified on each assignment.

Exams: We will have one midterm exam (~~Friday, Feb 4, 2011~~) (Monday, Feb 7, 2011) and a final exam (2:30-4:20pm Tuesday, March 15, 2011). Makeup exams will only be offered under extraordinary circumstances; you should plan to attend the exams when they are given. Exams will normally be closed-book, closed-notes, and calculators will not be allowed.

Late Policy: Written assignments: Due *promptly* at the BEGINNING of lecture, late assignments will not be accepted. If you cannot attend lecture please arrange to turn in your homework earlier to the instructor or have a classmate turn it in for you during lecture. The purpose of strict adherence to this policy is not to be a pain, but so that students do not skip lecture to finish their homework.

Programming projects: Programming projects will be turned in electronically (at a time announced for each assignment). Once per quarter you may use your "late day" to buy an extra 24 hours. You must email your TA before the deadline to specify that you will be using your late day so he or she can make appropriate arrangements. For the last two projects, if working with a partner, BOTH partners must have their late day available in order to take the late day.

If unusual circumstances that are truly beyond your control prevent you from submitting an assignment or attending an exam on time, you should discuss this with the instructor, preferably in advance. (Even if you're sick in bed at home, you should still be able to make a phone call or send an email.)

Grading and Evaluation: Grades will be computed *approximately* as follows (weights may be modified):

- 25% - Written Homework Assignments
- 25% - Programming Projects
- 20% - Midterm Exam (Friday, February 4, 2011)
- 25% - Final Exam (Tuesday, March 15, 2011, 2:30-4:20pm)
- 5% - Best of the four items above

Academic Integrity: Unless otherwise specified, you are to complete assignments individually. You may discuss the assignment in general terms (see description of Gilligan's Island rule on course web page), but the code you write must be your own. You are encouraged to discuss ideas, approaches, concepts, bugs, etc., in English, but you may not show or give your code to anyone except this course's TAs and instructor. You are not allowed to write code with another student on an assignment or to show another student your solution to an assignment. Referring to solutions found on the web or solutions from this or other courses from previous quarters is also considered cheating.

Communications: The course message board is a good medium for discussing the course, getting help on assignments, and staying in touch outside of class hours. You can also email the instructor or TAs or go to office hours. In addition, the course staff will occasionally post announcements to the course email list. You will be automatically subscribed to the course email list if you are registered for the course and will be held responsible for anything posted there.

Computing Resources: We will use Java 6 for programming assignments. We recommend although will not require that you use the Eclipse development environment. Links for downloading and installing Java and Eclipse can be found on our course web page.

CS 332 – First Day Assignments

0) **Review Java and explore Eclipse** – Instructions for installing on your home machine are on our course web page. Now would be a good time to review material from CSE 143!

1) **Project #1** – Your first programming project will be posted by Wednesday. Look for an email announcing its arrival to verify you are on the course mailing list!

2) **Preliminary Survey**: Please fill out the preliminary survey posted on our course web page (posted by end of day today) by the evening of Thursday Jan 6th. (Course home page = <http://www.cs.washington.edu/332/>)

3) **Information Sheet**: Please bring a sheet of paper with the following information with you to lecture on or before Friday Jan 7th.

A Picture of you!
Student ID is o.k. but something more
interesting or readable is nice too.

Name (and what you like to be called)

Email address

Year (1,2,3,4,5 i.e. freshman, sophomore, etc.)

Hometown

Interesting Fact about yourself and/or what you did over summer/winter break.

4) **Reading** in *Data Structures and Algorithm Analysis in Java*, 2nd Ed., 2007, by Weiss

- For this week:
 - › Chapter 1 – (review) Mathematics and Java (pp. 1-25)
 - › Chapter 3 – (Project #1) Lists, Stacks, & Queues
 - Lists (pp. 57-81, heavy on Java, much of this should be review)
 - Stacks (pp. 82-83)
 - Applications of Stacks (pp. 83-91, sections on “Postfix Expressions” and “Infix to Postfix Conversion” can be skipped, but read “Method Calls”)
 - Queues (pp. 91-95)
 - › Chapter 2 – (Topic for Wednesday) Algorithm Analysis (pp. 29-50)