CS 314: Data Structures

Spring 2015: TTh 2:00 - 3:30 in GDC 2.216, Unique Nos. 51755, 51760, 51765.

Instructor: Gordon S. Novak Jr., GDC 3.824; Office Hours: T Th 3:30 - 5:00.

Teaching Assistant:

- Bing Ai   `bingai@cs.utexas.edu`   Office Hours: M W 1:30 - 2:30, F 4-5 in GDC 1.302, TA Desk 1.

Lab Proctor will be in the Level 3 GDC labs to help with programming projects:

Optional Text: Mark Allen Weiss, Data Structures & Problem Solving Using Java, 4th Ed.;   Amazon; textbooks.com.   **Reading in Weiss**
Optional Text: Eric S. Roberts, Thinking Recursively with Java ;   Amazon.

iClicker: Each student is required to buy/rent an iClicker (any version of iClicker device; not iClicker GO app). This will be used for attendance and to reinforce and practice with the class material. Two iClicker points are given just for voting, and an additional point is given for a correct answer. Most iClicker questions and answers are online at **Clicker Questions**, and it is okay to review them in advance. The clicker scores will be converted to a grade by making the highest student score at least 108 and linearly scaling other scores; this gives some extra points to account for minor illness, forgotten or malfunctioning clicker, etc. Adjustments to clicker scores will only be made for significant causes such as major illness. Bringing another student's clicker to class is considered to be cheating.

Course Notes: Available in printed form at Copy Center, GSB 3.136 . Strongly recommended. Online by **Contents** (`http://www.cs.utexas.edu/users/novak/cs314contents.html`) or Index or PDF   Do not print out the slides on CS printers.

Web Page: `http://www.cs.utexas.edu/users/novak/cs314.html`

Programming assignments will mainly be done in Java, with options to do some in Lisp. Program files may be provided for use with the assignments, in the directory `/projects/cs314/` or in the FTP directory for Program Files, `ftp://ftp.cs.utexas.edu/pub/novak/cs314/`, and in `http://www.cs.utexas.edu/users/novak/cs314/`. It is legal to use any of these files as part of your programs.

Prerequisite: CS 312 or 312H with a grade of at least C-.

Topics:

- **Introduction:** why we *need* clever algorithms and data structures.
- **Performance: Big O:** how much time and storage are needed as data sizes become large
- **Some Lisp:** `(+ x 3)`
- **Lists and Recursion; Stacks and Queues**
- **Arrays**

- **Trees and Tree Recursion**
- **Using Library Packages**
- **Choosing the Right Data Structure**
- **Balanced Trees and Maps**
- **Hashing and Randomization**; XOR
- **Heaps and Priority Queues**
- **Sorting, Merge**
- **Graphs**
- **Map, Reduce, and MapReduce**

Grading:

Grades are kept on Canvas. It is **your responsibility** to check your grades often to make sure that your assignments have been received and graded.

Course grades are assigned on the scale A = 93-100, A- = 90-93, B+ = 87-90, B = 83-87, B- = 80-83, etc. provided that the Final Exam grade is at least 65; if the Final Exam grade is below 65, a lower course grade may be assigned at the instructor's discretion. Grades are averaged using the following weights:

Clicker Participation	10%	
Midterm Exam	20%	Thursday, March 12, in class
Final Exam	30%	Thursday, May 14, 2-5 PM
Programming Assignments	40%	

There will be a 10% per day late penalty for late programming assignments up to 4 days; no late assignments will be accepted after 4 days. It is the **student's responsibility** to submit assignments correctly using Canvas.

Programming projects must be your own individual work. Students may discuss concepts or help with specific problems in another student's code. However, sharing code constitutes plagiarism. The Moss program will sometimes be used to detect duplicated programs. It is not possible to pass the course without doing the programming assignments.