CS231 - Data Structures, Spring 2014

Monday and Wednesdays - 2:00pm - 3:20pm McGlothlin-Street Hall 20
Tuesday and Thursdays - 3:30pm - 4:50pm McGlothlin-Street Hall 20
Instructor: Prof. Robert F. Dickerson
TA: TBD
Textbook: Necaise, Data Structures and Algorithms with Python (optional)

Course Description

This course is a continuation of CS 141: Computation Problem Solving. The major topics of the course include data abstraction, data structures, and algorithms. Common data structures and algorithms, including arrays, lists, maps, stacks, queues, hash tables, trees, heaps, searching, sorting, and recursion, will be covered. The implementation of abstract data structures using classes gives this course a significant programming component.

Grading and Course Project

45% - Programming assignments (about 6)
25% - Midterm exam
35% - Final exam

Assignments

There will be 6 programming assignments. All assigned work is due by the date/time specified and emailed to your TA. Any assignment turned in after the due date/time but on the same day will be penalized 10%. Any assignment turned in after the day on which it is due, will be penalized 50% of the total value for that assignment. No assignment will be accepted that is more than three days late. If you turn an assignment in late, you must indicate this in your email. Test cases will be provided with the assignment instructions to test proper functioning of your program. Commenting you code is not necessary. Do not procrastinate on these assignments, waiting until the day before the assignment is due will be risky since many programming tasks require a lot of time if you get stuck. Seek help from office hours the week the project is due.

Use of laptops

Bringing laptops to class is strongly encouraged. Laptops are not to be used during class for reading email, social networking, completing assignments for other courses, etc. Many classes will involve programming along with the instructor or with teammates during class time.

Exams

The exams will be closed-textbook and closed-notes. No calculators or computers.

Grading policy

Forming groups to discuss programming assignments with other students is encouraged. However, sharing a solution in the form a particular implementation of code, is an honor violation. If you have any uncertainty about what this means, consult with me before you collaborate.

Disabilities

If you have a disability that may effect your participation in this course and wish to discuss academic accommodations, please contact me as soon as possible.

Tentative Course Schedule

Week 1 (01/16 and 01/18): Review of Python vs. JAVA
Week 2 (no class, 01/23, 01/25): Abstract Data Types
Week 3 (01/28, 01/30, and 02/01): Linked Lists and Collections
Week 4 (02/04, 02/06, and 02/08): Object-oriented design and Overloading
Week 5 (02/11, 02/13, and 02/15): Big O and Time Complexity
Week 6 (02/18, 02/20, and 02/22): Searching - Binary Search and Hash Tables
Week 7 (02/25, 02/27, and 03/01): Midterm exam
Week 8 (03/11, 03/13, and 03/15): Queues and Stacks
Week 9 (03/18, 03/20, and 03/22): Recursion
Week 10 (03/25, 03/27, and 03/29): Breadth-first search and Depth-first search
Week 11 (04/01, 04/03, and 04/05): Web services
Week 12 (04/08, 04/10, and 04/12): Heaps
Week 13 (04/15, 04/17, and 04/19): Trees
Week 14 (04/22, 04/24, and 04/26): Graphs and final review