CS 106 - Introduction to Data Structures
Spring 2019

Professor: Sorelle Friedler
sorelle@cs.haverford.edu

MW 12:45 - 2:15, KINSC H204
Labs: F 8:30 - 9:30 or 9:30 - 10:30 or 10:30 - 11:30, KINSC H110
http://www.cs.haverford.edu/courses/cmsc106

An introduction to the fundamental data structures of computer science: strings, lists, stacks, queues, trees, BSTs, graphs, sets and their accompanying algorithms. Principles of algorithmic analysis and object reasoning and design will be introduced using mathematical techniques for the notions of both complexity and correctness. More practical issues, such as memory management and hashing, will also be covered. The programming language used to illustrate and implement these concepts will be able to support functional, imperative and object-oriented approaches.

Lab: Includes a weekly required programming lab section. Class sessions will also be taught in a computer lab.
Prerequisites: CS 105 or BMC 110 or 113 with a grade of 2.0 or better.
Students who have taken CS 107 *may not* receive credit for this course.
Enrollment Limit: 30. The course will be divided into three lab sessions of at most 12 each.
Lab Instructor: John Dougherty
Lab monitors and TAs: Lab monitors are in KINSC H110 every Sunday - Thursday from 7-11pm to help you with your lab assignments. TAs will also hold office hours (TBD - see the class Google Calendar).
Office hours: see the class Google Calendar for times. Office hours are also available by appointment.
Textbook (required): *Data Structures and Algorithms in Java* by Goodrich and Tamassia.

Schedule of Topics

This schedule is *tentative*. Labs are due **by 11:59pm on Monday** in the week listed. Students should expect ***at least* 10 hours of work each week.** For the most up-to-date dates and deadlines see the CS 106 Google Calendar.

Week 1. Introduction. Java basics. The importance of documentation and programming style.

- Reading: Chapter 1
- Programming topics: Java syntax, types and type conversion, variable scope, Javadoc comments, very basics of classes.
- In lab: how to check out assignments in Eclipse.

Week 2. Basics of object oriented programming (objects and classes and inheritance), variable scope.

- Lab 0 Due: Java basics
- Reading: Chapter 2
- Programming topics: Objects, Classes, inheritance, variable scope, Strings.

Week 3. Object oriented programming basics.

- Reading: Chapters 2, 3.1
- Programming topics: type generics, exceptions.
- Data structures: basics of arrays, ArrayLists, and main args input.
- Ethics: reproducibility, reading and writing documentation.

Week 4. Linked lists.

- Reading: Chapter 3
- Lab 1 Due: Designing data structures

Week 5. Introduction to big-Oh notation

- Reading: Chapter 4
- Lab 2 Due: ArrayLists.

Week 6. Stacks and queues.

- Reading: Chapter 6

Week 7. Review week and midterm.

- Reading: review chapters 1-4 and 6
- Lab 3 Due: Linked lists.
- Wednesday: **Midterm 1 exam** during class time.
- Friday: all lab sessions cancelled.

Week 8. **Spring Break!**

Week 9. Lists, iterators, interfaces, and graph basics

- Reading: Chapter 7, 8.1, and 14.2
- Computational topics: graph basics (nodes, edges, weights, neighbors, degree, directed vs. undirected) and implementations (adjacency lists and adjacency matrices)

Week 10. Binary trees

- Reading: Chapter 8
- Lab 4 Due: Stacks and queues

Week 11. Priority queues

- Reading: Chapter 9

Week 12. Maps and hash tables

- Reading: Chapter 10
- Lab 5 Due: Binary trees
- Monday class: Dave Wonnacott teaches class

Week 13. Sorting and selection

- Reading: Chapter 12
- Lab 6 Due: Priority queues

Week 14. Search trees and union-find

- Reading: Chapter 11 and 14.7.3
- Lab 7 Due: Maps and hash tables (data deduplication part 1)

Week 15. Review week and midterm

- Reading: review chapters 7-12, 14.2, and 14.7.3
- Lab 8 Due: Sorting and selection (data deduplication part 2)
- Wednesday: **Midterm 2 exam** during class time.

Final project (decision trees) due at the end of final exam period.

Labs and Exams

A general outline of the labs is given below. Lab starter code will be distributed to the class.

0. Java basics
1. Data structure design
2. Array lists
3. Linked lists (with timing)
4. Stacks and queues
5. Binary trees
6. Priority queues
7. Maps / hash tables (data deduplication)
8. Sorting / selection (continuing data deduplication)
9. Final project: decision trees

There will be two midterms.

In addition to the larger labs above, coding exercises will also be given in class to be continued and checked in weekly lab sessions.

Total grade breakdown

Labs	50%
Coding Exercises	5%
Final Project	10%
Midterm 1	15%
Midterm 2	20%

Grades will be awarded based on the number of points earned and according to the percentage breakdowns shown. Students will not be graded on a curve.

Attendance and Participation

Attendance at, and active participation in, all class and lab sessions is expected of all students. Participation will be taken into account in awarding of final grades for students who are "on the edge" between two grades. For example, a student with a B+/A- average and a strong attendance and participation record would receive an A-, while a student with a weak record would receive a B+.

Late work policy

All extensions must be requested **at least 24 hours in advance** of the deadline. Extensions of 24 or 48 hours will be granted based on individual circumstances. Work handed in late without a previously granted extension will not be accepted (i.e., will receive zero credit).

Rules and Pet Peeves

- **Be on time.** This includes class, lab, office hours, and appointments.
- **Expect 24 hours before an email response** and read all emails within 24 hours.
- **Attend all classes and labs.**

Collaboration

Please see the full version of the departmental collaboration policy here. A summary is below, but students are encouraged to read the full linked document.

Work done in collaboration should never be copied from another student (e.g., from their computer or from joint work on the board). Work from previous semesters should never be shared with current students, or looked at by students in the current semester, though it is fine to share notes you make about lectures or the textbook. Code and other material should never be copied from another student or outside sources unless permission is explicitly given in advance by your professor and the code is cited.

If you are ever in doubt about if your collaboration or use of outside sources is appropriate, please talk to the professor or TAs for clarification.

Learning Accommodations

Haverford College is committed to providing equal access to students with a disability. If you have (or think you have) a learning difference or disability - including mental health, medical, or physical impairment, please contact the Office of Access and Disability Services (ADS) at hc-ads@haverford.edu. The Coordinator will confidentially discuss the process to establish reasonable accommodations.

Students who have already been approved to receive academic accommodations and want to use their accommodations in this course should share their verification letter with me and also make arrangements to meet with me privately as soon as possible (ideally within the first two weeks) to discuss the specific accommodations. Please note that accommodations are not retroactive and require advance notice to implement.

It is a state law in Pennsylvania that individuals must be given advance notice if they are to be recorded. Therefore, any student who has a disability-related need to audio record this class must first be approved for this accommodation from the Coordinator of Access and Disability Services and then must speak with me. Other class members will need to be aware that this class may be recorded.