

Instructor: David Parkinson
Office: 14-240
Office Hours: Wednesday 5:30-7:30pm, and by appointment
Lecture Room: 14-250
Lab: 14-301
Email Address: dsparkin@calpoly.edu

Course Goals

The course is an introduction to data structures and analysis of algorithms. The goal of the course is that students will be able to:

- Basics of Data Structures: Lists, Stacks, Queues, Trees, Heaps, Hash Tables, Graphs
- Appropriate application of data structures
- Concepts for correct and professional programs: Design recipes (structured design), loop invariants, etc. Testing and test coverage
- Modularity, but not object oriented programming – understand concepts of ADT and information hiding
- Emphasize both recursive and iterative problem solving, tradeoffs

Prerequisites: CPE 101 or equivalent, with a grade of C- or better

Textbook: Problem Solving with Algorithms and Data Structures using Python.

<http://interactivepython.org/runestone/static/pythonds/index.html> and additional Web sources:

Important: Make sure you regularly check the course plan and the labs/assignments on PolyLearn. These are tentative and subject to change, but will give you a good idea of what will be happening in the course and when.

Class Structure

Research in learning and cognitive science indicates that active participation significantly improves learning and retention. Learning takes place when people think hard about how to apply the material they are trying to learn. Thus, the course is structured to encourage active and continuous learning. To do well you should expect to:

- Read and understand the textbook. Lecture will cover some of the same material sometimes from a different perspective. We will also cover some different algorithms to provide more depth to the material.
- Participate in labs. Labs will be a combination of structured and unstructured time to work on labs and assignments. Labs are a great opportunity to work on your assignments and get feedback or help.

Course work

The course outline contains assigned readings, labs, and assignments. You are expected to read and understand the textbook. Being able to read and understand technical material is a crucial skill for every computing professional. The best way to learn the material is by solving problems. On exams and quizzes, the problems given will be very similar to homework problems, assignments, labs, examples from the text, and problems solved and discussed in class. Working together on labs is strongly encouraged; however, all writeups and programming assignments must be individual work.

The more you attempt to solve problems on your own until you can to solve them without any help, the more you will master the material. The midterms and final exam will be closed book. For exams, you should be able to reconstruct problem solutions and algorithms on demand. Note this does not require memorization but an in-depth understanding of the idea behind the data structure and its operations and the ability to translate that into a detailed description of the algorithm similar to those given in class. Many students are overconfident in their ability to do this. **You need to test yourself by doing this without looking at the solution.**

Computing Environment

You will be required to complete the labs assignments in this class using Python 3.4 (or later). It is freely available for all major platforms, including Mac OS, Windows, and UNIX. It is pre-installed on the lab machines.

We do not require you to use a particular IDE (Integrated Development Environment), but we've installed PyCharm on the lab machines, and you are invited to use it.

GRADING

- Labs 10%
- Assignments 30%
- Quizzes 10%
- Midterm 20%
- Final Exam 30% (Cumulative)
 - Midterm: Friday, Feb 8 (date is tentative), during lecture time, in lecture room (14-250).
 - Final Exam: TBD.
 - Exams can only be made up for a documented emergency.
- No late labs or assignments accepted. If you have a specific need for an extension, contact me ahead of the deadline – the sooner, the better.

Course Learning Outcomes

The student should be able to:

- Describe and define the Abstract Data Types (ADT) covered, including stacks, queues, priority queues, binary search trees, AVL trees, hash tables, and graphs.
- Understand, explain, and track by hand alternate (iterative, as well as recursive) implementations of methods associated with each ADT.
- Analyze for performance all algorithms associated with ADTs, reason about correctness and effectiveness of procedures, compare and contrast array and linked implementations of certain ADTs.
- Implement and test each ADT implementation as a generic program.
- Use ADTs effectively in software applications

ADMINISTRATIVE POLICIES

Academic Integrity

Collaboration on labs and assignments is allowed, unless otherwise noted. However the write-ups of assignments and the actual programming to be handed in must be individual work. (Thus you may not hand in assignments jointly or cut and paste material.) Collaboration on exams is not permitted. Any violation of this policy may result in an F in the course and be reported to Academic Affairs. See the campus statement on Academic Dishonesty: Cheating and Plagiarism

Communications

Email and PolyLearn will frequently be used to communicate important information. **You are responsible for reading these.** If you don't use your Cal Poly email account regularly, you should setup your Cal Poly email account to forward your mail to your regular email account. I will respond to communications via email to the best of my ability. You are also encouraged to use of office hours.

Classroom behavior

You are expected to act professionally and respectfully at all times to your colleagues and the instructor. **Use of electronic devices is not permitted during lecture without my express permission.** The use of electronic devices during class has been shown to be detrimental both to the learning of the student using the device but also to other students in the class.

Drop/Withdrawal policy.

You may use CPreReg to drop this course any time during the first eight days of class. Carefully evaluate your schedule and determine if you will remain in the class before the end of the add/drop period. After the drop date, the only way out of the course is called "withdrawing" from the course; this requires a documented "serious and compelling" reason, such as a medical emergency.

Students with special learning needs

If you are having difficulty with some aspect of the course due to any special learning needs you may have, I encourage you to contact me. If you have a disability for which you are or may be requesting an accommodation, you are encouraged to contact both your instructor and the Disability Resource Center, Building 124, Room 119, at (805) 756-1395, as early as possible in the term. Use of Disability Resource Center services, including testing accommodations, requires prior authorization by the DRC and compliance with approved procedures.