CS-241: Data Structures

Calvin Deutschbein (they)

Willamette University, Fall 2021

E-mail: ckdeutschbein@willamette.edu
Web: **cd-public.github.io/**

Office Hours: By Appt. & TBA
Lecture MWF 8:00-9:00 AM
Office: Zoom, Ford 206
Lecture Hall: Ford 204

Lab Room: Ford 202
Lab Hours: MWF 9:10-10:10 AM

Course Description

Theoretical and practical study of programming and abstract data types including lists, stacks, queues, trees and algorithms used on these data structures. The course includes object implementation of structures and sharpens programming skills learned in previous courses.

Required Materials

Required materials will be available on the course webpage under resources.

Prerequisites

This class is meant for computer science students who have completed introductory coursework in programming. The prerequisites are CS 141 or CS 151 (though CS 151 is recommended as the course will primarily use Python).

Accessability

I will make every effort to ensure all coursework and materials are accessible to all students, including working with on-campus specialists. However, there is always room for improvement. I always appreciate hearing from students about how I can make the course more accessible, so please reach out if there is something I can be doing better!

Course Objectives

This course will teach you how to organize the data used in computer programs so that manipulation of that data can be done efficiently on large problems and large data instances. This course will address both how to use the data structures found in the libraries of programming languages, and how those libraries are constructed and why the items that are included in them are there (and why some are excluded).

- You will gain familiarity with important categories of problems that are commonly encountered in software development, and will learn what data organizations will allow practical and efficient solutions to those problems.
- You will learn the basics of how to describe and analyze the performance and efficiency of your algorithms; this will prepare you for the upper level course in algorithms.
- You will demonstrate the concepts you learn by encoding them in correct Python programs.
- You will gain more proficiency in basic programming and in constructing larger programs than you have been doing in your intro classes.
- You will practice the induction proof methods to reason about your programs, and to argue in support of them.

Competency of the data structures taught in this course will prepare you to study higher level areas where those data structures are heavily used: operating systems, networking, graphics, vision, compilers, databases, and security.

Course Structure

The course will be composed of lectures, labs, homeworks, midterms, and a final project.

Lecture Structure

Lectures are scheduled for Mondays, Wednesdays, and Fridays at 8:00 AM in Ford 204. The schedule of lectures will be available on the course webpage under schedule.

Lectures will be primarily on the white board, with some coding. I will support a remote option for all lectures, likely through a streaming platform or Zoom, and will post recordings of lectures on the course website. While I will make every effort to follow best practices for accessible teaching, I will make mistakes! Please, if you find some material is inaccessible for any reason do not hesitate to reach out.

Lab Structure

In addition to lecture, you must also sign up for the associated laboratory component of the class, which would ordinarily meet in Ford 202 in the scheduling slot immediately following the lecture (9:10 to 10:10 AM). I will follow the established practices for operating labratory sessions:

- The labs will be open during the scheduled hours, so that you can use the lab computers to complete the assignments.
- I will be available via Zoom or in person in my office in Ford 206 during the labs to answer questions and help you when you get stuck.
- You may do the lab work on your own computer and need not attend the lab sessions at the scheduled times.
- On weeks when assignments or projects or not due, I reserve the right to assign labs. You are responsible for submitting the the required materials for a lab session by the deadline at 11:10 AM on Fridays.
- There will be no lab assignments during the weeks that assignments are due, to encourage you to use lab sessions to work on the projects.

Homework Structure

Feedback will be provided on homework assignments. Homework assignments will be considered when determining grades for this course.

Homeworks will consist of programming assignments to be completed outside of class and submitted for feedback. There will be a "Homework 0" at the beginning of the semester to get used to programming and assignment submission, then four homework assignments throughout the semester. You will always have at least two weeks to complete homework assignments.

Homework assignment will be due at 8:00 AM on Fridays so we can discuss them in class on the day they are due.

As a rule, I encourage students to submit their assignment as-is at the due date and not to submit late work. By way of explanation, it is my experience as both a student and a grader that time spent working on assignments after their deadline is often better spent working on the next assignment.

As-is submission is supported in the grading policy by dropping the lowest homework grade. In special circumstances such as extended medical problems or other unforeseeable emergencies, please reach out and so we can collaboratively develop a more personal solution to achieve the learning objectives of the course.

Midterm Structure

Feedback will be provided on midterm exams. Midterm exams will be considered when determining grades for this course.

There will be two written midterm final exams, intended to be completed individual without access to notes or documentation. The midterms are intended to achieve a learning focus of reasoning about data structures in isolation from coding environments, as well provide me as an instructor with greater insight into how effective course instruction has been.

Final Project

Feedback will be provided on the final project. The final project will be considered when determining grades for this course.

There will be a final project that will be similar in format but distinct in content from homeworks. Whereas homeworks will including programming assignments for which there is some correct answer that may be measured against an answer key and notions of coding style, the project will be a student-centered exploration of data structures meant to give you an opportunity to apply what you've learned in the course while still receiving support from an instructor.

The final project will be released as soon as the final homework is due, on November 12, and will be due at the time of the final examination as set by the registrar.

Feedback and Grading

Feedback will be provided on assignments, midterms, and the final project using a 100 point scale. This 100 point scale is intended to be familiar to established grading standards, such as letter grades. To provide aggregate feedback for the whole course, these feedback scores will be combined as follows:

- **40%** of your grade will be determined by homework assignments

  10% each for the four highest scored assignments (out of five).

- **40%** of your grade will be determined by midterm exams.

  20% each for the two midterms.

- **20%** of your grade will be determined by the final project.

Feedback scores will constitute the minimum grade on an assignment, but the instructor may exercise discretion at any time to award a higher grade. For example, a submitted homework may not use some important algorithmic technique as submitted, but if the student showed familiarity with this technique on an earlier assignment or exam, the absence of that technique in a specific case need not be counted against a student in grading, but only noted in feedback. This corresponds to the high level notion of feedback corresponding to how well an assignment reached the intended learning goals, while the overall course grade is meant to indicate that a student is prepared to succeed in latter coursework. Under this model, the final project will offer an opportunity to show familiarity with all content in the course, so a strong final project can ensure a high course grade for any student, regardless of prior scores on midterms and homeworks.