

Syllabus

CSCI 136: Spring 2019
Handout 1
1 February

Data Structures and Advanced Programming

Instructor Prof. Daniel Barowy
Office TCL 307
Email dbarowy@cs.williams.edu

Instructor (Lab) Prof. Bill Jannen
Office TCL 306
Email jannen@cs.williams.edu

Lectures MWF 11:00-11:50am (Barowy) in Schow 030A
Labs W 12–2pm, 2–4pm (Due Sundays before midnight)

Web Page <https://williams-cs.github.io/cs136s19-www/>

Texts

We will be using the $\sqrt{7}$ edition the following text book:

- *Java Structures: Data Structures in Java for the Principled Programmer, $\sqrt{7}$ Edition*, by Duane Bailey.

Do not use earlier editions! A PDF version is available on the course website. We have also printed copies of the text book as a course reader. We encourage you to take a copy of the course reader; your term bill will be charged whether you take a copy or not, but we will reuse unclaimed books for future courses.

Course Objectives

Goal. The goal of this course is to enable you to write good programs, and to instill both an intuitive and an analytical understanding of what we mean by “good” in computer science. Throughout the semester, you will design, analyze, code, and verify that your programs work as expected.

Data structures. The primary vehicle for learning the above skills will be through the study of data structures, which are principled methods for storing and manipulating data. Data structures and algorithms, which you will study in CS256, are two sides of the same coin. Both are essential for the construction of the kinds of large, reliable computer programs used by billions of computers users on a daily basis.

Java. Although it may be unfamiliar to you, we will be using the Java programming language for this course. Unlike Python, which you may have been exposed to in CS134, Java is augmented with a number of features that greatly simplify the challenges of writing large-scale programs (e.g., static types). Developing fluency in Java will benefit not just your performance in this class, but your abilities as a programmer in the future.

The elements of style. In addition to correctness and performance, this course will help you learn how to write programs in a clear and modular manner. Programs written and documented clearly are easier to maintain and result in fewer bugs. Modular code substantially reduces coding effort and also results in fewer bugs. Don’t be surprised if you receive feedback that your program needs work even if it correctly implements an assignment’s specification.

Lab resources. This course will primarily use the MacOS computers in TCL 216 & 217 for programming assignments. You will be given door codes to access this room once the semester begins. While you are permitted to use your own computer if you wish, please be aware that requests to provide software support for your personal machine likely will be answered with a polite, “No, use the lab machines.”

Typical Course Activities

Workload. The work that you should expect to engage with, beyond the scheduled lectures and weekly lab meetings, will involve

- Reading the text: 12-15 pages, on average, per lecture
- Preparing for weekly pop quizzes
- Preparing for the weekly programming labs
- Completing the weekly labs
- Studying for the mid-term and final exam

Some students program quickly but read slowly, and some do the opposite. You should expect to spend at least 10 hours a week beyond the scheduled lecture and lab hours on this course. If you find yourself spending substantially more time than that on a regular basis, please see the course instructor.

Labs. On most weeks, there will be lab programming assignments. **Attendance in lab is mandatory:** there are valid reasons to miss lab, but any unexcused lab absence will result in course failure. All programs will be graded on the basis of design, documentation, style, correctness, and efficiency. Programs should be turned in electronically by 11:59pm on the due date, typically the Sunday following your lab.

Exams. There will be one midterm and one final exam. The midterm will be scheduled during your lab period on **Wednesday, March 13**, and it will replace the lab for that week. The final exam is a scheduled exam during the exam period. Stay tuned for the date.

Github

All assignments for this course will be submitted using Github. Prior to an assignment, a Github repository will be created for you. Repository names generally conform to the following pattern: `https://github.com/williams-cs/cs136.lab<n>_<your github username>`. You will be notified by email when your Github repository is created.

Late days

Each student may use a maximum of **three free late days** during the course of the semester. A late day permits you to hand in an assignment up to 24 hours late, without penalty. Late days are provided to help students deal with unforeseen circumstances and to allow some balancing of occasional uneven work demands. They should be used judiciously; if you find yourself struggling with the workload of this (or any other) course, we encourage you to reach out to your instructor.

You may use **up to two** late days for any one assignment. Once late days are exhausted, late work will not be accepted; the most recent on-time submission will be graded so that some partial credit may still be obtained. Using a late day requires that you

1. Commit a `late.txt` file to your repository before the regular deadline that
 - (a) contains the number of late days you plan to take, and
 - (b) the date of the expected final submission.
2. When you have submitted the completed assignment, please email your instructor to make it clear when your final work has been submitted.
3. Any late assignment that does not follow this procedure will be graded on the basis of the last commit before the regular deadline.

Resubmissions

Even well-intentioned students struggle with course material and occasionally “bomb” an assignment. We want you to know that such an event is not the end of the world, and to incentivize you to engage in self-reflection, we allow **up to two** assignment resubmissions during the semester. This policy includes the first nine labs and the midterm exam, but not the final lab or the final exam.

A resubmission will be accepted at the discretion of the course instructor and allows you to earn back **up to 50% of the missing points**. For example, if you received a 75% on an assignment, you may earn up to 87.5% upon resubmission.

Resubmissions must be submitted in the following manner:

1. They must be submitted within two weeks of receiving graded feedback.
2. They must include both the original work and the new submission.
3. They must be accompanied with a PDF text document, written in plain English, that explains
 - (a) what the mistake is in the original work,
 - (b) how you fixed the mistake, and
 - (c) why the new version is correct.

Code Reviews

You will participate in two one-on-one code reviews during the semester with one of our teaching staff, Carl Rustad. This is an opportunity for you to get personalized feedback on your code. In addition to aspects like correctness, code reviews are a good way to learn how to achieve better clarity or elegance from an experienced programmer.

Carl will meet with 10 students per week. You may voluntarily sign up to meet with Carl at any time, assuming that he is not already booked. To ensure that everyone is seen throughout the semester, if Carl does not have 10 signups, we will randomly choose ~~vietim~~slucky students for him to meet with.

Grades

Grades will be determined as follows:

Final exam:	20%
Midterm exam:	20%
Programs/Labs:	35%
Code reviews:	10%
Engagement:	15%

Engagement is a subjective evaluation of how you interact with the material and the class as well as a measure of your performance on weekly pop quizzes. We will use the full range from 0 (e.g., does not come to class, uses the computer during lectures, etc.) to 15 (e.g., attends all sessions, participates in class, does well on quizzes, refers to the readings, raises new topics, shares knowledge with other students, etc.). Learning is collaborative; this is our way to reward students who positively affect the experience of fellow class members.

Note that you can ensure your preparedness on the weekly quizzes by reading the assigned material on time and by paying attention to the “quiz prompt” listed on the course website.

Help!!!

There are many resources available when you need it. You are encouraged to discuss any questions, concerns, difficulties, or thoughts about the course with your instructors (Dan and Bill). In addition, TAs are available to help you

with challenges you face as you work through the course material and lab assignments. You are welcome at any time to approach course staff to ask for clarification of the assignments, and to discuss your problem-solving process. You do not need to wait until you are stuck and frustrated to speak with us!

If you find yourself facing challenges beyond the typical, please do not stay silent. Talk to your instructor, a friendly face from the Dean's Office, or some of the many professionals across campus who stand ready to help. All faculty and staff at Williams are bound by the Family Educational Rights and Privacy Act (FERPA) to maintain the privacy of your educational records. We understand that difficulties arise, and we are prepared to help you.

Contrary to popular belief, the most successful students are not "effortlessly successful." Instead, they get to know course staff early on and they familiarize themselves with an institution's academic support resources. Williams has ample support resources, including

- The Peer Tutor Program: Tutors can be arranged when 1-1 help is required beyond that available from your instructor and TAs. <https://academic-resources.williams.edu/peer-tutor-program/>
- Math & Science Resource Center: Support is available for students grappling with the more quantitative aspects of their coursework. <https://academic-resources.williams.edu/math-science/>
- Accessible Education and Disability Support Center: Students with documented disabilities may require accommodations in certain situations. If that's you, take advantage of the options available. <https://academic-resources.williams.edu/disabilities/>
- The Health Center: Sometimes your challenges are not course-related. The Health Center provides a range of medical, psychological, and health/wellness services. <https://health.williams.edu>