

Course Syllabus

Instructor	Questions?
Name: Prof. Blank E-mail: blank@caltech.edu Office: ANB 115	For all course-related questions, please email cs001@caltech.edu . If there is something particularly sensitive, you may email Prof. Blank directly.
Course Website	Lecture
https://cs1.caltech.codes Visit early. Visit often.	Beckman Institute Auditorium MWF 02:00 – 02:55 PM

Course Philosophy

We have structured the course around our four core values:

CS and programming are for everyone!

We have taken care to design and implement CS 1 to be accessible to students interested in any option, **regardless of prior experience**. We firmly believe that every student can gain valuable, lifelong skills from this course! We will only consider the course successful if we are able to serve you in whatever discipline you end up working in. Inclusivity, of all kinds, is one of our core values, and we strive to ensure that everyone feels included and welcome in all aspects of the course.

Our projects cover diverse disciplines!

As CS 1 is a part of core, we aim to provide points of connection to as many other fields as possible. To accomplish this, we've designed weekly projects to be "themed" around different disciplines (physics, chemistry, engineering, math, etc.). We're aiming to have at least one project that feels relevant and cool to every student. (If we miss on this, please please let us know so we can add/edit for the future.)

CS 1 material is sequenced to set you up for success!

The course is separated into three modules: (0) Reading and Modifying Python code, (1) Using Python to Get Things Done, and (2) Another Programming Language and Preparation for Future CS Courses. Here's more details, in case you're curious:

- In Module 0, we will imagine that you're working in an existing codebase (perhaps, for research) and help you navigate how to work with code in this context. We start here, because, realistically, most code you write won't be from scratch.
- In Module 1, we will work on helping you create a mental model of what Python is doing; so, you can effectively debug your own code and write small programs from scratch.
- In Module 2, we will switch to the Java programming language to compare and contrast with Python. This isn't only because we want to prepare you for future CS courses! Lots of research, in many fields, is not done in Python! Languages like C++, C, Rust, and Julia all require some understanding of features that learning Java sets you up for!

CS 1 is not an "easy" course, but we want you to succeed!

Our goal is to help you succeed. Please ask for help if you need it. Productively struggling is good, but struggling in silence is needless and painful. If you are stuck on any single issue for more than a half-hour, ask for help at office hours. We allocate a ton of TA and Professor time (more than twenty people-hours) to office hours because we want to provide as much help as possible, and we hope you will find them as helpful as we have in the past!

Grading Scheme

To pass, you must meet the following criteria:

- Pass **at least two** of the quiz, midterm, and final.
- Complete every DUE and earn at least 50% on the project itself.
- Earn an average of at least 65% of the diagnostics (we will drop your worst two).

To calculate your shadow grade, we will use the following formula:

$$0.5 \times (\text{the best two of quiz, midterm, final}) + 0.4 \times (\text{Project Average}) + 0.1 \times (\text{Diagnostic Average})$$

In-Person Assessments

In this course, there will be both *formative* (i.e., primarily intended to help you learn something new) and *summative* assessments (i.e., primarily intended to see what you have learned).

Formative Assessments (a.k.a. diagnostics)

During lectures, we give in-class exercises (ranging from multiple choice to fill-in-the-blank to write an explanation). Together, these questions will be called a “diagnostic”. Each individual question given in lecture will be graded on a 0-1 basis. Your total score on a diagnostic will be the percentage of questions you got correct on that diagnostic. We will drop your two lowest diagnostic scores. If you attend lecture on a given day, you will be given a chance to work on the in-class exercises fully collaboratively for large chunks of lecture. If you do not go to lecture (or you want to improve your score on a diagnostic from lecture that you didn't get completely correct), you are eligible to complete the “second attempt” outside of class.

Summative Assessments

There will be three summative assessments which will each be during a normal lecture period: a quiz, a midterm, and a final. Each of these will be slightly different, but they will all be 55 minutes in duration.

Written Quiz

The written quiz, which will happen around week 3, is intended as a check-in early in the term to help you assess your current understanding of the material. 60% of the points will be slightly modified diagnostic questions from lecture. The remainder will be “novel” questions that you haven't seen before. **To pass the quiz, you will need to earn a score of at least 40%.**

Written Midterm

The written midterm, which will happen around week 6, is intended as a summative assessment that tests if you're on track to meet our learning objectives for the course. 60% of the points will be slightly modified diagnostic questions from lecture. The remainder will be “novel” questions that you haven't seen before. **To pass the midterm, you will need to earn a score of at least 60%.**

Written Final

The written final, which will happen during the last week of classes, is intended as a summative assessment that tests if you've completely met our learning objectives for the course. **We expect that many students will NOT need to take the final, as you only need to pass two of the three in-person assessments.** However, we will note that, as this is a final exam, it will be cumulative, and thus may be more difficult than the quiz and midterm. 100% of the points will be “novel” questions that you haven't seen before. **To pass the final, you will need to earn a score of at least 60%.**