Course Information

Course Overview Welcome to CS106A! Over the next ten weeks, we'll explore the fundamental techniques, programming constructs, and design strategies that form the basis of modern software. By the time you've completed the course, you'll be comfortable writing computer programs that interact with the user, process and manipulate data, and report information back textually and graphically. In the course of doing so, you'll gain exposure to computational problem-solving techniques and learn more about the sheer breadth of computing applications.

Instructor Keith Schwarz (htiek@cs.stanford.edu)

Office: Gates 178
Office Phone: (650) 723-4350
Office Hours: TBA

Head TA Alisha Adam (aadam@stanford.edu)

Office: Gates 160
Office Hours: TBA

Website The course website is http://cs106a.stanford.edu and it's loaded with useful resources. There, you'll find all the handouts for this course, lecture slides, lecture code, assignment starter code, software links, relevant news articles, and much more. I would suggest periodically polling the website to stay abreast of important developments in the course.

Lectures This quarter, we're offering two lectures of CS106A. The lectures should be pretty much identical and you're welcome to attend either of them. Both lectures will be held on Monday, Wednesday, and Friday. The first will be held from 10:00AM – 10:50AM in Bishop Auditorium (in the Lathrop Library) and the other from 3:15PM – 4:05PM in Nvidia Auditorium (in the Huang Engineering Center). The 3:15PM lecture will be recorded and posted online through SCPD, and there will be a link to the videos on the course website.

Prerequisites There are no prerequisites for this course, and everyone is welcome!

If you already have some familiarity with programming, CS106A may still be an appropriate course for you, but you may want to consider moving on to one of the more advanced programming courses. Handout #02 contains placement information that you may find useful in deciding whether CS106A is right for you. As always, feel free to get in touch with us if you have any questions.

Units If you are an undergraduate or are taking CS106A through SCPD, you must enroll in CS106A for five units. If you are a matriculated graduate student, you may enroll for between three and five units, depending on what best fits into your schedule. Regardless of how many units you are enrolled for, the course content and requirements will be the same; the unit flexibility is purely to provide unit flexibility for matriculated graduate students.

Discussion Sessions   In addition to lecture, you must also sign up for a weekly 50-minute section. Section signups are handled online at http://cs198.stanford.edu/section. Section signups open on Thursday, January 8 at 5:00PM and close Sunday, January 11 at 5:00PM. We will then send you an email with section assignments by Tuesday, January 13. Sections begin the second week of classes.

Although Axess lists discussion sections for this course, it's not necessary to enroll in a discussion section on Axess (and, in fact, we completely ignore section enrollments on Axess). We handle all section assignments through the above link, regardless of any section times you've signed up for on Axess.

LaIR Hours   The team of CS106A Section Leaders and Course Helpers are here to help make this course the best that it can be. If you need help on any of the assignments, feel free to stop by the Tresidder LaIR between 6PM and midnight, Sundays through Thursdays. You can view the most current helper schedule by going to http://cs198.stanford.edu and clicking the "Helper Schedule" link. This link is also available on the CS106A web page. LaIR hours begin the second week of class.

Readings   This course has two required readings:

Karel the Robot Learns Java. This 35-page tutorial introduces major concepts in programming through Karel the Robot, a friendly robot who runs around in a grid world. In the first week of the quarter, we'll use Karel to explore programming, problem solving, and software engineering. This course reader will be a valuable resource as you work through the first assignment. The reader is available electronically on the course website and in hardcopy from the Stanford Bookstore.

The Art and Science of Java by Eric Roberts. Once we've acclimated to the wonderful world of Karel, we'll begin building larger and more elaborate programs using the Java programming language. *The Art and Science of Java* is a fantastic introduction to software engineering in Java, and you will definitely want to have a copy as we start moving into more interesting and advanced topics.

In addition to these readings, we'll be periodically distributing handouts in class. These handouts, which will also be available online at the course website, should help supplement the treatment of the material given in the other readings. We will store extra copies of the handouts in the "handout hangout" (the B wing of the first floor of the Gates building), so don't worry if you forget to pick up a copy.

Computers   The assignments in CS106A will ask you to write programs in the Java programming language. We strongly recommend using the Eclipse environment for designing, testing, running, and submitting your assignments. Eclipse is available freely online, and we'll release a handout detailing instructions on how to download and install it.

Eclipse should run smoothly on Mac and Windows computers. If you have a Linux computer, you can probably use that as well; contact Keith or Alisha for details on how to set up Eclipse on Linux.

Assignments   There will be **eight** programming assignments over the course of the quarter, each of which will give you a chance to play around with the material from lecture and build progressively more impressive pieces of software. The assignments will become slightly more difficult and require more time as the quarter progresses, so the later assignments are weighed slightly more than the earlier ones.

Except for Assignment 8 (which is due at the very end of the quarter), each assignment is graded during an interactive, one-on-one session with your section leader, who rates it according to the following scale:

++   An absolutely fantastic submission of the sort that will only come along a few times during the quarter. Any grade of ++ must be approved by the instructor and TA.

+   A submission that exceeds our standard expectation for the assignment. To receive this grade, a program often reflects additional work beyond the requirements or gets the job done in a particularly elegant way.

✓+   A submission that satisfies all the requirements for the assignment. It reflects a job well done.

✓   A submission that meets the requirements for the assignment, possibly with a few small problems.

✓−   A submission that has problems that cause it to fall short of the requirements for the assignment.

−   A submission that has extremely serious problems, but nonetheless shows some effort and understanding.

−−   A submission that shows little effort and does not represent passing work.

From past experience, we expect most grades to be ✓+ and ✓. Dividing the grades into categories means that your section leader can spend more time talking about what you need to learn from the assignment and not have to worry about justifying each point. The overall goal is to maximize the learning experience in doing the assignments, and we have found the "bucket" grading system to work much better for programming assignments than assigning numeric grades from a pedagogical perspective.

Late policy   Each of the assignments is due at 3:15PM on the dates specified in the syllabus. The program code for your assignments must be submitted electronically as described in a separate handout. All assignments are due at 3:15PM sharp on the dates indicated on the assignment handout. Anything that comes in after 3:15PM, even if only by a few seconds, will be considered late.

Everyone gets a bit overwhelmed every now and then, so to give you some flexibility with assignment due dates, everyone begins the quarter with **two** free late periods. Each late period lets you submit one assignment one class period late. For example, an assignment due on Monday could be submitted on Wednesday using a late period, and an assignment due on Friday could be submitted on Monday using a late period. Late periods do factor in national holidays, so an assignment due on the Friday before MLK day could be submitted on the following Wednesday using one late period. You may use both late days on a single assignment if you'd like, though we strongly advise against this as you'll almost certainly fall behind in the course.

If you submit an assignment late after using all your late periods, we will assign a 1% penalty to your overall grade in the course per class period late. No assignment submissions will be accepted more than three class periods after the stated due date, even if you use late periods. Also, note that late days may not be used on Assignment 8, because it is due during the time slot normally reserved for the CS106A final exam.

Midterm Exams	In addition to the seven programming assignments, there will be two midterm exams. The first midterm exam will be held on **Tuesday, February 10** from **7PM – 10PM**, and the second will be held on **Tuesday, March 3** from **7PM – 10PM**. These midterm exams will ask you to solve various coding problems on paper without the aid of a computer.
	If you have a conflict that will prevent you from taking either of the exams, we'd be happy to try to find an alternate time. However, you must let us know about this no later than one week in advance so that we have adequate time to schedule another exam room and time (given that we have 600+ students, it's tricky to find enough space to hold everyone!) Note that no alternate exams will be given after the scheduled exam times, so all alternate exams must be taken prior to the standard exam time.
Grading	Overall, your grade for this course will be determined as

Programming Assignments:	55%
First Midterm:	20%
Second Midterm:	20%
Section Participation:	5%

Note that we do not offer "make-up" assignments or make-up work, though by adding extensions to the assignments we will award extra credit that can be used to raise your overall grade.

Extensions	That said, if there is a medical or family emergency and you will need a deadline extension on any of the programming assignments or exams, please contact the head TA, Alisha Adam, no more than 24 hours before the deadline to request an extension. **Only Alisha can approve extensions**, and in particular your section leader does not have the authority to grant extensions.
	You may find it useful to think of your two free late periods as pre-approved extension requests that you can use without contacting us. In non-emergencies, we will ask that you use your late days rather than requesting a deadline extension on the assignments.
Incomplete Policy	If you have a serious medical or family emergency that will prevent you from completing the work in CS106A, you may want to arrange to take an incomplete in CS106A and resolving the remaining assignments and exams in a future offering of CS106A.
	We reserve incompletes only for emergencies that would prevent you from completing the remaining work in the course. In particular, we do not grant incomplete grades for poor performance on the assignments, nor do we grant incomplete grades for busy work schedules.
	In order to request an incomplete, you must have completed all assignments and exams in the course at the time at which you request an incomplete, except for possibly the most-recently-due assignment or exam.