

CS 106A — General Information

Professor: Eric Roberts

E-mail: eroberts@cs.stanford.edu
Office: Gates 202
Phone: 723-3642
Drop-in hours: Tuesdays, 9:30–11:30 A.M.

TA: Jeremy Keeshin

E-mail: jkeeshin@cs.stanford.edu
Office: Gates 160
Phone: 723-6083
Office hours: TBA

Lectures

Lectures are scheduled for Monday, Wednesday, and Friday at 1:15 P.M. in Hewlett 200. The schedule for individual lectures is given on the accompanying calendar handout.

Discussion sections

In addition to lecture, you must also sign up for a weekly 50-minute section. In order to take CS 106A, you must sign up for a section between 5:00 P.M. on Thursday, April 5, and 5:00 P.M. Sunday, April 8. The signup form is available on the web at the URL <http://cs198.stanford.edu/section/>. Important note: Do not use Axxess to sign up for sections; section enrollments made through Axxess are simply ignored.

Units

If you are an undergraduate, you are required to take CS 106A for 5 units of credit. If you are a graduate student, you may enroll in CS 106A for 3 units if it is necessary for you to reduce your units for administrative reasons. Taking the course for reduced units does not imply any change in the course requirements.

Drop/add deadlines

You may not add or drop courses from your study list after 5:00 P.M. on Friday, April 20, without having that course appear on your transcript with a notation indicating that you have withdrawn from the course. The last day to change your status to CR/NC or to withdraw from the course is Friday, May 25.

Web page

The web page for CS 106A is <http://cs106a.stanford.edu/>. All the materials and course announcements will be posted on this web site, so be sure to check it frequently.

Texts and handouts

There are two required texts for this class, both of which are available from the Stanford Bookstore. The first is *Karel the Robot Learns Java*—a 35-page tutorial that introduces

the major concepts in programming in the context of an extremely simple robot world. The second is the textbook, *The Art and Science of Java* (Addison-Wesley, 2008). In addition to these texts, we will also distribute additional material in the form of class handouts. After class, any extra copies of the handouts will be placed in the “Handout Hangout” bins in the entryway to the Gates B-wing between Gates 182 and 188. The handouts are also available in PDF® format on the CS 106 web site. If you miss a handout in class, you can print your own copy from the web.

Examinations

The midterm examination will be administered at two different times on Tuesday, May 8: from 9:00 to 11:00 A.M. and from 3:15 to 5:15 P.M. The final examination will also be offered at two times: Friday, June 8 from 8:30–11:30 A.M. and Monday, June 11 from 8:30–11:30 A.M. The examinations themselves are written so that you should be able to complete them in less than the fully allotted time: the midterm is designed to take one hour and the final is designed to take 100 minutes. All examinations are open-book, and you may use any notes or materials from the class.

Programming assignments and problem sets

CS 106A requires seven programming assignments, which are due on the dates given in the calendar handout. Except for Assignment #7 (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 then assigns two grades: one for functionality and one for programming style. In each of those categories, the grades are chosen according to the following scale:

- ++ An absolutely fantastic submission of the sort that will only come along a few times during the quarter. To ensure that this score is given only rarely, any grade of ++ must be approved by the instructor and TA. Since your section leader would almost certainly want to show off any assignment worthy of a ++, this review process should not be too cumbersome.
- + A submission that exceeds our standard expectation for the assignment. The program must reflect additional work beyond the requirements or get the job done in a particularly elegant way.
- ✓+ A submission that satisfies all the requirements for the assignment—a job well done.
- ✓ A submission that meets the requirements for the assignment, possibly with a few small problems.
- ✓– A submission that has problems serious enough 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.

For each assignment, you must make an appointment with your section leader for an interactive-grading session. Your section leader will explain in section how to schedule these sessions and go over the grading process in more detail.

In addition to the assignments, there will also be two problem sets during the quarter. These problem sets help to reinforce your understanding and focus on concepts that require practice beyond what you are likely to get from the programming assignments. Problem sets are distributed in class and are due on the dates shown on Handout #4.

Late policy

Assignments are submitted electronically using the procedure described in the first assignment. **Important note: all assignments are due at 5:00 P.M. sharp on the dates indicated on the assignment handout. Assignments submitted after 5:00 will be considered late.**

Because each of you will probably come upon some time during the quarter where so much work piles up that you need a little extra time, every student begins the quarter with two free “late days.” To avoid any ambiguity, a “day” is defined as a class day. Thus, if your assignment was due on Friday but turned in the following Monday, that assignment would be one day late. After your late days for the quarter are exhausted, programs are assessed a late penalty of one category point per late day used (a ✓+ turns into a ✓, and so forth). Late days are valuable, and it pays to keep some around for the harder assignments toward the end of the quarter. In all cases, assignments must be turned in within a calendar week of their published due date.

In special circumstances (such as extended medical problems or other emergencies), extensions may be granted beyond the late days. To request an extension, send e-mail to jkeeshin@cs.stanford.edu no later than 24 hours before the program is due. Only Jeremy is authorized to approve such extensions. In particular, do not ask your section leader.

Contests

As shown on the calendar handout, there are three contests scheduled at different points during the term. The point of these contests is to give you a chance to show creativity and initiative beyond what is formally required by the course. Rules for each contest will be distributed in class when they are announced.

To encourage greater participation in the contests, we will offer two additional incentives. First, every reasonably serious entry will get you one virtual ticket in a random drawing for a special grand prize at the end of the quarter. The more contests you enter, the more chances you have. Winning runner-up prizes or honorable mentions in a contest or submitting assignments that are candidates for ++ scores give you additional chances. The random drawing will take place at the beginning of the review section for the final exam.

As a second incentive, we're going to borrow a page from Harry Potter and award "house points" for extra-credit activities according to your year of entry. If you're a first-year student, for example, entering a contest not only gives you a virtual ticket in the end-of-the-quarter contest, but also gives the first-year class one point in the standings. The class with the most points in proportion to the number of students from that class will win bragging rights at the end of the term. The current standings will be displayed on the course web page. Graduate students and others who have managed to be around more than four years will be counted with the seniors.

Grading

For as long as anyone can remember, the most important component of the final grade in CS 106A has been the programming assignments, which have typically counted for approximately half the final grade. Even so, one of the biggest complaints we hear from students is that the assignments don't count for enough relative to the exams. Many students feel that exams are not a particularly good measure of one's knowledge. If, after all, someone can implement the assignments effectively, why should it matter how that person performs on an examination taken under arbitrary time constraints without the aid of a computer.

Although I'm entirely sympathetic with this argument in theory, there is a problem. Computer science courses—here at Stanford as well as at most other institutions—have historically been marked by an intolerably high number of Honor Code cases, often representing more than half of the cases reported in a year. Given that problem, it has always seemed necessary to use exams as a check to make sure that students have in fact learned the material. Someone who copies their assignments from someone else may do very well on those assignments (assuming we don't catch it), but will in all probability do poorly on the midterm and the final.

Two years ago, I tried an experiment in CS 106B that I have repeated ever since in both CS 106A and CS 106B. If Honor Code violations do not seem to be a problem, we'll give the assignments even greater weight than we traditionally have. If, however, cheating rears its ugly head, we'll shift that weight to the exams.

Here, then, are the weights for the different components of the course:

Programming assignments	60% minus 5% for each Honor Code case reported
Final examination	15% plus 5% for each Honor Code case reported
Midterm examination	15%
Problem set	5%
Section participation	5%

For example, if no Honor Code cases at all come up this quarter in CS 106A, the programming assignments would be worth 60% of the grade, and the final would be worth only 15%, making it the same as the midterm. If, however, two Honor Code cases were discovered, the assignments would be worth only 50% and the final would be worth 25%, which is pretty much what we've traditionally done. If more Honor Code cases are discovered, the weight on the assignments will continue to decrease, but in no case will drop below 15%.

There is one further clarification that I need to make. The only violations that count are suspected Honor Code violations reported by the course staff. If *students* in CS 106A report an Honor Code violation (as you are supposed to do, after all), we won't count that report against the class as a whole. This policy, therefore, gives everyone a strong collective incentive to maintain academic integrity.

Computer facilities

As in any programming course, the assignments in CS 106A require extensive hands-on use of a computer. The preferred platform for doing the work is the Eclipse environment, which runs under both Mac OS X and the various flavors of Microsoft Windows. Instructions for obtaining copies of the Eclipse environment—which is an open-source software project and therefore free—will be distributed on Wednesday.