

# CSCI 356 Fall 2017 Syllabus

## Course meetings:

Lecture	Tues 3:30 - 6:20	ZHS 159
Discussion 1	Fri 10:00 - 11:50 am	SSL 150
Discussion 2	Fri 12:00 - 1:50 pm	THH 208

You are responsible for everything covered in lectures, including administrative announcements. While discussion is not required, it is strongly encouraged that you attend *a discussion* each week. You are not obligated to attend the discussion in which you are enrolled.

**Required Textbook:** Randal E. Bryant and David R. O'Hallaron, *Computer Systems: A Programmer's Perspective, Third Edition (CS:APP3e)*, Pearson, 2016.

Please make sure you have the Third Edition, not the Second Edition.

**Recommended Textbook:** Brian W. Kernighan and Dennis M. Ritchie, *The C Programming Language, Second Edition*, Prentice Hall, 1988.

**Gradebook:** [blackboard.usc.edu](http://blackboard.usc.edu)

**Forums:** [piazza.com/usc/fall2017/csci356](http://piazza.com/usc/fall2017/csci356)

Blackboard will be used for posting of grades. Messages that do not need a particular instructor's attention should be posted to Piazza with the appropriate privacy setting.

## Course Description

Course covers computer systems and related topics with a goal of improving students' ability to understand what a computer does to execute a program, with the intent that this will improve their abilities as programmers. This course will also serve as a basis for other systems courses, such as CSCI 350 (Operating Systems), CSCI 450 (Computer Networks), or CSCI 457 (Computer Systems Organization).

For purposes of graduation and prerequisites of other courses, this will count as CSCI / EE 352L.

**Prerequisite:** CSCI 104L

**Recommended Preparation:** Familiarity with memory management and parameter passing as it is used in the C++ programming language; the prerequisite of CSCI 104L covers this.

## Grade Calculations:

Artifact	Weight	Date	Time
Exam One	15%	Tuesday Oct 3	3:30 - 4:50 pm
Exam Two	15%	Tuesday Nov 7	3:30 - 4:50 pm
Final Exam	30%	Tuesday Dec 12	2:00 - 4:00 pm
Programming Assignments	40% (see below)	various	11:59pm Pacific (according to us)

## Exams

You will be provided with paper on which to take the exam. Exams will be individual effort, closed-book and closed-notes. You will be allowed **one** 8.5x11inch handwritten note sheet (front & back) on the exams.

Students requiring alternate exam arrangements must make such requests within the first two weeks of the term, or as soon as possible after knowing of the conflict or requirement.

## Programming Assignment Grades and Weights

Programming assignments will not be accepted until *you* have completed the syllabus quiz. Furthermore, we will be collecting these via git using the same process that is used in other courses here. If you are not familiar with git, please alert course staff promptly.

Assignment	Weight
1	3%
2	3%
3	6%
4	6%
5	7%
6	7%
7	8%

## **Late Policy and Grace Days**

Five times during the semester, a student may extend the due date of a programming assignment by twenty four hours without needing prior permission. These are known as "grace days." In order to use a grace day, you must submit a form (which will be provided) shortly after you wish to submit the assignment.

Please note that grace days are *in place of* "excused late" submissions, not in addition to. If you request additional grace days from the instructor, you must have a documented reason for each grace day used to accompany your request. Once you have used your grace days, any future late days used will cost you 15% of the related assignment's maximum points (minimum score of 0).

Each assignment has a maximum number of grace days permitted for it; this is a combination of "free extension" grace days and penalty-related grace days, not merely a maximum on free usage.

Note: There is no grace period. Even if you submit a few minutes after the deadline, you will need to use a grace day (even if the wireless network in your dorm room is down or you have a github issue, etc.). It is your job to be on time and not cut it too close. Remember Murphy's Law and leave time for things to "go wrong."

## **Grading Reconsideration**

Reconsideration requests for any graded artifact must be made within one week of our release of grades for the item. A form will be available on the course Piazza page for you to use to explain your reconsideration request. You must fill out the form within that week. Once the reconsideration period has passed, grades for each artifact are considered final.

For the exams, grading reconsideration will require filling out the same form and handing your midterm exam *to the instructor only* within that one week period.

Requests for grading reconsideration submitted in any other fashion will not be considered.

## **Academic Honesty**

It is the expectation that each graded artifact submitted represents the sole work and understanding of the author turning it in.

Please see the academic honesty supplement at the end of the syllabus for a description of what collaboration is and is not acceptable in programming assignments, as well as how to credit acceptable assistance on these assignments.

The university's recommended sanction for plagiarism is an F in the course.

In any course with a pending academic honesty violation report, you may not drop the course; if you drop the course and are later discovered to have violated the academic honesty agreement, you will be re-enrolled.

### CSCI 356 Projected Course Schedule: A Weekly Breakdown

Please note that this is a *projected* schedule and is subject to change. Any changes will be announced via the course Piazza page.  
 In addition to reading the book portions, we encourage you to do the practice problems in the associated sections. Unless stated otherwise, all reading is in the textbook of Bryant and O'Hallron.

	Topics/Daily Activities	Readings	Deliverable/ Due Dates
<b>Week 1</b> Aug 22	Overview Bits and Bytes	Chapter 1 2.1	
<b>Week 2</b> Aug 29	Integers C Programming	2.2 - 2.3 K&R	
<b>Week 3</b> Sept 5	Floating Points Machine-level representation	2.4 3.1-3.5	Project 1 due 9/6
<b>Week 4</b> Sept 12	Machine-level procedures	3.6	Project 2 due 9/17
<b>Week 5</b> Sept 19	data structuring, Combining control + data	3.7, 3.8 3.9, 3.10	
<b>Week 6</b> Sept 26	Memory Hierarchy, caches	6.1-6.3	Project 3 due 10/1
<b>Week 7</b> Oct 3	Exam 1 3:30 - 4:50 <i>Lecture 5-6:20 -- caches</i>	6.4-6.7	
<b>Week 8</b> Oct 10	Code optimization	Chapter 5	Project 4 due 10/14
<b>Week 9</b> Oct 17	Linking,	Ch. 7	
<b>Week 10</b> Oct 24	System Level I/O	Ch. 10	Project 5 due 10/27
<b>Week 11</b> Oct 31	Dynamic Memory Allocation	9.9 - 9.12	Project 6 due 11/5
<b>Week 12</b> Nov 7	Exam 2 3:30 - 4:50 <i>Lecture 5:00 - 6:20 -- virtual memory</i>	9.1-9.3	
<b>Week 13</b> Nov 14	Virtual Memory Address Translation	9.4-9.6	
<b>Week 14</b> Nov 21	Virtual Memory	9.7-9.8	
<b>Week 15</b> Nov 28	Exceptional control flow: exceptions and signals	8.1-8.4 8.5-8.8	Project 7 due 11/30

## CSCI 356 Academic Honesty Supplement

The following are a supplement for this course to the university academic honesty guidelines.

- Any code obtained with help of a course producer, TA, or instructor must be *clearly marked in comments* with who and when. This may not constitute a meaningful portion of your project. The substring “assistance from” must appear in the comment acknowledgement near the code on which you received assistance, followed by the name of who you got assistance from.
- The help you receive from classmates should be limited to *conceptual* help -- how an algorithm you need to implement works as a “big picture” rather than at the level of code.
- If you receive help from others, whether an instructor, course producer, or fellow student, follow the *Kenny Loggins Rule*:
  - You may discuss high-level ideas and receive hints regarding how to solve portions of the assignments. However, neither party should keep any written record from this discussion. Afterwards, take a 30-minute break and do something unrelated to the course (watching an episode of your favorite cartoon show, for example). You may now return to your assignment.
  - When you write a section of code based on help received, add a comment acknowledging the help, including the substring “assistance from” as part.
- You are **explicitly prohibited** from seeking help outside of course resources for the programming projects. The following is an **exhaustive** list of “course resources”:
  - The instructor, TA(s), and course producers.
  - Your fellow students. Remember that this means discussing concepts, not sharing code.
  - The textbook, Randal E. Bryant and David R. O’Hallaron, *Computer Systems: A Programmer’s Perspective, Third Edition (CS:APP3e)*, Pearson, 2016.
  - The *public* portion of the textbook’s website.
  - The course Piazza page for this semester.
  - Course lectures and discussions, along with any notes provided by instructors.
  - The supplemental textbook, *The C Programming Language* by Brian Kernighan and Dennis Ritchie
- The previous bullet point means that if you seek information towards solving part of a programming assignment online, find it, and use it, you risk this as an academic honesty violation. *Do not search for anything related to the programming assignments online for any reason without explicit permission from the instructor. This includes “clarification” and “for reference only”*
- If you use code fragments from one of the textbooks mentioned above, you must include a comment acknowledging the source, including the substring “assistance from” and the textbook citation (textbook name, authors, and page number is sufficient).
- If you are using git, use your *private repository* so your code is not available to your fellow students. Course staff will help set you up with a private repository early in the semester.