

# CS 164 Syllabus

22 January 2014

Welcome to CS 164. Here's what the bulletin says we should do:

A study of software project management concepts, software cost estimation, quality management, process involvement, overview of analysis and design methods, user interface evaluation, and design. Also considered are dependable systems – software reliability, programming for reliability, reuse, safety-critical systems, verification and validation techniques; object-oriented development; using UML; and software maintenance.

Here's what we'll actually do: learn the above stuff in the context of a **real software project**. You will be responsible for many parts of the system yourself, but we will discuss the overall design and direction as a class so that we can stay on track and learn from each other.

**When:** Monday, Wednesday 11am–12:50pm

**Where:** LLC 207

**Credits:** 3

## Contact Info

**Instructor:** Prof. Christopher League, Ph.D.

**Email:** [christopher.league@liu.edu](mailto:christopher.league@liu.edu) – please include the course number (CS164) in the subject.

**Google Hangout:** [cleague@gmail.com](mailto:cleague@gmail.com)

**AIM:** chryslleague

**Skype:** chrisleague

**Office hours:** Monday, Wednesday 2–2:50 (US Eastern) or make an appointment at <https://liucs.net/bookme>

**Office phone:** +1 718 488 1274

**Office location:** LLC 206, LIU Brooklyn

## Resources

**Web sites:** <https://liucs.net/cs164s14/>

<https://github.com/liubrooklyn/cs164s14>

<https://blackboard.liu.edu/>

<http://www.gradechamp.com/>

**Textbooks:** *The Pragmatic Programmer* (**very highly recommended**) by Andrew Hunt and David Thomas, Addison-Wesley 0-201-61622-X. [[Paperback](#), [Kindle](#), [iBooks](#), [PDF](#)]

*The Mythical Man-Month* (**recommended**) by Frederic P. Brooks, Jr., Addison-Wesley 0-201-83595-9. [[Paperback](#), [Kindle](#), [iBooks](#)]

**Library:** Campus library resources tailored for computer science are available at <https://liucs.net/u1>

## Requirements

There are a total of 1,000 points available, broken down as follows:

- There will be 7 **project milestones** scheduled throughout the semester. The exact requirements and expectations for each will be posted to the course web site. Your contribution will be worth **125 points each**, but I will drop the lowest, so that only 6 milestones count, for a total of **750 points**. **Warning:** the *last* milestone cannot be dropped.
- There will be 7 ‘**check-in**’ opportunities scheduled. These vary from week to week, but may involve responding to a survey, taking a brief online quiz, or participating in a discussion forum. Check-ins are worth **25 points each**, but I will drop the lowest two scores so only 5 will count, for a total of **125 points**.
- There is no midterm exam, but there will be a final exam, worth **125 points**.

On the 1,000-point scale, you can expect the following letter grades:

	≥ 870:	<b>B+</b>	≥ 770:	<b>C+</b>	≥ 670:	<b>D+</b>	
≥ 930:	<b>A</b>	≥ 830:	<b>B</b>	≥ 730:	<b>C</b>	≥ 600:	<b>D</b>
≥ 900:	<b>A–</b>	≥ 800:	<b>B–</b>	≥ 700:	<b>C–</b>	else:	<b>F</b>

In the end, I may choose to adjust the scale slightly to compensate for assignments or questions that turned out to be trickier than I intended. Such adjustments would never *lower* your grade from what is designated in the above table; if you achieve 930 points, you are guaranteed an A.

## Policies

**No late assignments will be accepted**, because we will discuss and evaluate your work promptly after the deadline. This helps to ensure that everyone receives timely feedback, and that you can learn from mistakes while they are still fresh in your mind.

**There will be no extra credit.** Students usually ask for extra credit late in the semester after they have already messed up their original opportunities. Be sure to start your

work early, so that we can detect and solve any problems before they can affect your grade.

**Plagiarism** is the use or presentation of ideas, words, or work that is not one's own and that is not common knowledge, without granting credit to the originator. Plagiarism is a practice that is not only unacceptable, but which is to be condemned in the strongest terms possible on the basis of moral, educational and legal grounds. Under University policy, plagiarism may be punishable by a range of penalties from a failing grade in the assignment or course to dismissal from the School of Business, Public Administration and Information Sciences. All students are required to read the handbook on avoiding plagiarism by visiting <https://liucs.net/u2>

**Cheating** includes, but is not limited to the following: falsification of statements or data; listing sources that have not been used; having another individual write your paper or do your assignments; writing a paper or creating work for another student to use without proper attribution; purchase of paper or research work for one's submission as his/her own work; using written, verbal, or electronic or other sources of aid during an examination (except when expressly permitted by the instructor, depending on the nature of the examination) or knowingly providing such assistance to aid other students.

In a course with programming assignments, it is usually okay to work with and learn from other students to **some** extent, but what you submit in the end needs to be your own. The most reliable way to do that would be to set aside whatever code you created together, and then recreate it from scratch on your own.

**Showing up on time** to class every week is extremely important. If you must be absent or more than 5 minutes late, please try to notify me in advance. I will be keeping track of whether you are in class, and when you arrive. A few missed classes will not count against you, but habitual absence will significantly hurt your grade. Additionally, there will be no make-up quizzes. I do not distinguish between 'excused' and 'unexcused' absence. Unless you miss an *exam* due to a severe medical emergency, I don't want to see a doctor's note. If you do miss an exam, the make-up exam will be different – and probably *not* easier.

Long Island University seeks to provide **reasonable accommodations for all qualified persons with disabilities**. This University will adhere to all applicable federal, state and local laws, regulations and guidelines with respect to providing reasonable accommodations as required to afford equal educational opportunity. It is the student's responsibility to register with Special Education Services (SES) as early as possible and to provide faculty members with the formal communication from SES for suitable accommodations. All accommodations must be approved through SES. Contact Information: 718 488 1221 or 718 488 1044.

I participate in the **LIU Safe Zone** program. Representatives of the program serve as contacts for individuals on campus with questions or concerns related to sexual orientation and gender identity, whether of self or of a friend or family member. The goal of the program is to promote a safe and free campus for all students. Safe Zone areas can be identified by a sticker with the LIU Safe Zone logo.

The **Family Educational Rights and Privacy Act (FERPA)** gives students control over the disclosure of their educational records. During this course you may have the opportunity to create accounts or register with certain public online services. In these cases, you need not make any personally identifying information public. You may use a pseudonym or online handle, as long as you identify yourself to the instructor.

## Time commitment

New York State defines one credit as a total of 15 hours instructional time, plus 30 hours of student preparation. Thus, a typical three-credit course will amount to 45 hours instruction plus 90 hours preparation. (For these computations, an ‘hour’ actually consists of 50 minutes.)

To perform well, you will have to spend some time preparing and reviewing outside of class, and a **significant** amount of time completing programming assignments (keeping in mind that earlier assignments will require less time than later ones).

- Lecture time: 4 hours per week  $\times$  15 weeks = 60 hours
- Preparation time (reading, reviewing): 2 hours per week  $\times$  15 weeks = 30 hours
- Assignment completion (problem-solving, programming): approximately 10–14 hours per assignment  $\times$  7 assignments = 90 hours.
- **Total: 180 hours**

## Goals and objectives

Upon completion of the course, students should be able to...

- demonstrate proficiency in basic algorithms and data structures (1.1, mastery level).
- understand the mathematical and logical foundations of computing (1.2, mastery level).
- master the fundamental facilities of various programming languages and software architectures (2.1, mastery level).
- effectively use tools for software development (2.2, mastery level).
- develop a data modeling design for a proposed database application (3.2, mastery level).
- communicate technical ideas and specifications in writing (4.1, introductory level).
- give an effective oral presentation on some technical subject area (4.2, introductory level).
- exhibit awareness of professional organizations and technical opportunities (5.1, mastery level).
- productively attend seminars and workshops outside of class work (5.2, mastery level).

## Schedule

Wed 22 Jan Meeting 1 at 11 am: Introduction

Mon 27 Jan Meeting 2 at 11 am: *Read Brooks ch. 2 “The Mythical Man-Month”.*

Tue 28 Jan Check-in 1 due at midnight.

Wed 29 Jan Meeting 3 at 11 am: *Read Pragmatic §13 “Estimating,” and Spolsky “Evidence-Based Scheduling”.*

Mon 3 Feb Meeting 4 at 11 am: *Read Pragmatic §36 “The Requirements Pit”.*

Tue 4 Feb Milestone 1 due at midnight.

Wed 5 Feb Meeting 5 at 11 am: *Read Brooks ch. 5 “The Second-System Effect”.*

Mon 10 Feb Meeting 6 at 11 am: *Read Brooks ch. 11 “Plan to Throw One Away”.*

Tue 11 Feb Check-in 2 due at midnight.

Wed 12 Feb Meeting 7 at 11 am: *Read Pragmatic §11 “Prototypes and Post-it Notes”.*

Tue 18 Feb Meeting 8 at 11 am: *Read Pragmatic §7 “The Evils of Duplication”.*  
Milestone 2 due at midnight.

Wed 19 Feb Meeting 9 at 11 am: *Read Pragmatic §8 “Orthogonality”.*

Mon 24 Feb Meeting 10 at 11 am: *Read Pragmatic §21 “Design by Contract”.*

Tue 25 Feb Check-in 3 due at midnight.

Wed 26 Feb Meeting 11 at 11 am: *Read Pragmatic §26 “Decoupling and the Law of Demeter”.*

Mon 3 Mar Meeting 12 at 11 am: *Read Pragmatic §29 “It’s Just a View”.*

Tue 4 Mar Milestone 3 due at midnight.

Wed 5 Mar Meeting 13 at 11 am: *Read Pragmatic §34 “Code That’s Easy to Test”.*

Mon 17 Mar Meeting 14 at 11 am: *Read Pragmatic §43 “Ruthless Testing”.*

Tue 18 Mar Check-in 4 due at midnight.

Wed 19 Mar Meeting 15 at 11 am: *Read Pragmatic §18 “Debugging”.*

Mon 24 Mar Meeting 16 at 11 am: *Read Graham “Maker’s Schedule, Manager’s Schedule”.*

Tue 25 Mar Milestone 4 due at midnight.

Wed 26 Mar Meeting 17 at 11 am: *Read Pragmatic §14 “The Power of Plain Text”.*

**Mon 31 Mar Meeting 18** at 11 am: *Read Pragmatic §15 “Shell Games”.*

**Tue 1 Apr Check-in 5** due at midnight.

**Wed 2 Apr Meeting 19** at 11 am: *Read Pragmatic §17 “Source Code Control”.*

**Mon 7 Apr Meeting 20** at 11 am: *Read Pragmatic §42 “Ubiquitous Automation”.*

**Tue 8 Apr Milestone 5** due at midnight.

**Wed 9 Apr Meeting 21** at 11 am: *Read Pragmatic §22 “Dead Programs Tell No Lies”.*

**Mon 14 Apr Meeting 22** at 11 am: *Read Pragmatic §23 “Assertive Programming”.*

**Tue 15 Apr Check-in 6** due at midnight.

**Wed 16 Apr Meeting 23** at 11 am: *Read Pragmatic §31 “Programming by Coincidence”.*

**Mon 21 Apr Meeting 24** at 11 am: *Read Pragmatic §33 “Refactoring” and Graham “Taste for Makers”.*

**Tue 22 Apr Milestone 6** due at midnight.

**Wed 23 Apr Meeting 25** at 11 am: *Read Pragmatic §2 “Software Entropy”.*

**Mon 28 Apr Meeting 26** at 11 am: *Read Pragmatic §44 “It’s All Writing”.*

**Tue 29 Apr Check-in 7** due at midnight.

**Wed 30 Apr Meeting 27** at 11 am: *Read Pragmatic §46 “Pride and Prejudice”.*

**Mon 5 May Meeting 28** at 11 am: *Read Brooks ch. 16 “No Silver Bullet”.*

**Tue 6 May Milestone 7** due at midnight.