# CS 101 Syllabus

18 January 2017

Welcome to CS 101, a broad overview of the main areas of study in Computer and Information Sciences. Topics include computer organization, information processing, algorithms, and programming. The main ideas behind the theory and design of Operating Systems, Databases, and Computer Networks, along with current views on the theory and practice of Software Engineering, and the basics of Artificial Intelligence are also explored. The course highlights the uses of computing systems in business, the sciences, and other professional fields. This course is required for all students majoring in Computer Science or Information Systems. It is also suitable for majors in other disciplines who want to go beyond being casual users of computers to gain a deeper appreciation of some of the most important computing and information technologies developed over the last fifty years.

"Computation and algorithmic thinking have become essential components for solving problems in many different fields. As such, computer scientists are intimately involved in finding solutions to some of the most pressing social, economic, and scientific problems of our day" — Goldweber, Barr, and Patitsas, in SIGCSE 2013

**When:** Monday 9–10:50 AM
**Where:** LLC 207
**Credits:** 3
**Prerequisites:** None

## Contact Info

**Instructor:** Prof. Christopher League, Ph.D.
**Email:** christopher.league@liu.edu — please include the course number (CS101) in the subject. I have several email addresses, but all messages end up in the same place, so **please use only one.**
**Google Hangout:** cleague@gmail.com
**Office hours:** Monday, Wednesday 2:00–2:50, or make an appointment at https://liucs.net/bookme
**Office phone:** +1 718 488 1274
**Office location:** LLC 206, LIU Brooklyn

## Resources

- We will use several web resources:
  - https://liucs.net/cs101s17/ — notes, schedule, assignment handouts
  - http://www.gradechamp.com/ — grade reports

If you have a question or problem that might also apply to other students, *please* ask on the discussion forum rather than by email. Then the GA and other students can help you too, and the solution is available for all to see. Try to use email only for personal matters such as your grades.

- The textbook is *Computer Science Illuminated* by Dale and Lewis (6th edition, ISBN 144-967-2841, http://amzn.to/1yksnd6). An older edition is fine, if that saves you some money! I created a table that correlates section numbers across editions 3–6: https://liucs.net/u562

## Requirements

Your grade will be computed based on assignments, exams, quizzes, and participation. There are a total of 1,000 points available, broken down as follows:

- There will be **8 assignments** during the semester. Assignments are worth **50 points each,** for a total of **400 points.**
- There are **6 quizzes** scheduled throughout the semester, to make sure you are following along with the lectures and online review resources. Quizzes are worth **20 points each,** but I will **drop the lowest two scores** so only 4 will count, for a total of **80 points.**
- There will be **10 online 'check-in' opportunities** scheduled, roughly one per week. 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 **15 points each,** but I will **drop the lowest two scores** so only 8 will count, for a total of **120 points.**
- There will be a midterm and final exam, worth **200 points each** for a total of **400 points.**

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

| | | | | | | | |
|---|---|---|---|---|---|---|---|
| | | ≥ 870: | **B+** | ≥ 770: | **C+** | ≥ 670: | **D+** |
| ≥ 930: | **A** | ≥ 830: | **B** | ≥ 730: | **C** | ≥ 600: | **D** |
| ≥ 900: | **A–** | ≥ 800: | **B–** | ≥ 700: | **C–** | else: | **F** |

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

It is important to **complete tasks on time,** so you don't fall behind. Missed check-ins and quizzes will receive a zero, and cannot be made up (but remember, the lowest

two scores are dropped). If you need to miss an exam, try to notify me in advance so we can make other arrangements. **Late assignments** will be graded as follows.

This formula specifies a *lateness factor* $\mathrm{f}$ that is multiplied by your earned score to determine a late score. The variable $\mathrm{h}$ represents the number of hours the submission is late.

$$\mathrm{f} = \frac{8.5 - \log_2\left(\frac{\mathrm{h}}{24}\right)}{10}$$

**There will be no extra credit.** Students usually ask for extra credit late in the semester after they have already squandered 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 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,** whether psychological, neurological, chronic medical, learning, sensory, or physical. The 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 Student Support Services as early as possible and to provide faculty members with the formal communication for suitable accommodations. Visit Pratt 410, call 718 488 1044, or visit http://www.liu.edu/Brooklyn/SSS

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.

## Goals and objectives

Upon completion of this course, I expect that you will be able to…

1. understand and follow basic algorithms.
2. understand the mathematical and logical foundations of computing.
3. understand the role of programming languages in software architecture.
4. explain the key concepts of operating systems and computer networks.
5. work with simple data models structured as tables and keys.
6. exhibit awareness of professional organizations and technical opportunities.
7. be inspired to attend computing seminars and workshops outside of classwork.

## Assessment of learning

This course is part of a B.S. program in Computer Science. This section relates programmatic objectives with objectives and assessment instruments used in this course. At the completion of the B.S. program…

**1.1: Students will demonstrate proficiency in basic algorithms and data structures.** Introduced with course objective 1.

**1.2: Students will understand the mathematical and logical foundations of computing.** Introduced with course objective 2.
**2.1: Students will master the fundamentals of programming languages and software architecture.** Introduced with course objective 3.
**3.1: Students will master the key concepts of operating systems and computer networks.** Introduced with course objective 4.
**3.2: Students will develop a data modeling design for a proposed database application.** Introduced with course objective 5.
**5.1: Students will exhibit awareness of professional organizations and technical opportunities.** Introduced with course objective 6.
**5.2: Students will attend seminars and workshops outside of classwork.** Introduced with course objective 7.

Program objectives 1.2 and 2.1 are routinely assessed by scoring a selection of questions from students' midterm exams in CS101.

## Schedule

Sections numbers in the textbook are noted as '§1.2', and correspond to the 5th edition. If you have a different edition, see the table at https://liucs.net/u562 to translate the section numbers.

**Wed Jan 18** **Meeting 1** at 9 am. Introduction, and positional numbering. *Read §2.1.*
**Mon Jan 23** **Meeting 2** at 9 am. Binary natural numbers. *Read §2.2.*
**Wed Jan 25** **Meeting 3** at 9 am. Signed two's complement numbers. *Read §3.1, 3.2.*
**Thu Jan 26** **Check-in 1** due at 23:59.
**Mon Jan 30** **Meeting 4** at 9 am. Text encodings. *Read §3.3.* **Quiz 1.**
**Wed Feb 1** **Meeting 5** at 9 am. Text compression.
**Sun Feb 5** **Check-in 2** due at 23:59.
**Mon Feb 6** **Meeting 6** at 9 am. Graphics encoding. *Read §3.5.*
**Tue Feb 7** **Assignment 1** due at 23:59.
**Wed Feb 8** **Meeting 7** at 9 am. Audio/video encoding. *Read §3.4, 3.6.* **Quiz 2.**
**Sun Feb 12** **Check-in 3** due at 23:59.
**Mon Feb 13** **Meeting 8** at 9 am. Boolean logic. *Read §4.1, 4.2.*
**Tue Feb 14** **Assignment 2** due at 23:59.
**Wed Feb 15** **Meeting 9** at 9 am. Activity about combinational circuits. *Read §4.3, 4.4.*
**Tue Feb 21** **Meeting 10** at 9 am. Memory and the Turing Machine model. *Read §4.5, 5.1.* **Check-in 4** due at 23:59.
**Wed Feb 22** **Meeting 11** at 9 am. The von Neumann architecture and machine code. *Read §5.2, 6.1, 6.2, 6.3.* **Quiz 3.**
**Sun Feb 26** **Assignment 3** due at 23:59.
**Mon Feb 27** **Meeting 12** at 9 am. Algorithms and complexity. *Read §6.5, 7.1, 7.2.*

**Wed Mar 1** **Meeting 13** at 9 am. Searching and sorting. *Read §7.4, 7.5.*
**Sun Mar 5** **Assignment 4** due at 23:59.
**Mon Mar 6** **Meeting 14** at 9 am. Python syntax.
**Wed Mar 8** **Meeting 15** at 9 am. Midterm exam.
**Thu Mar 9** **Check-in 5** due at 23:59.
**Mon Mar 20** **Meeting 16** at 9 am. Python programming.
**Tue Mar 21** **Check-in 6** due at 23:59.
**Wed Mar 22** **Meeting 17** at 9 am. More programming.
**Sun Mar 26** **Check-in 7** due at 23:59.
**Mon Mar 27** **Meeting 18** at 9 am. Other programming languages. *Read §9.2, 9.3.*
**Wed Mar 29** **Meeting 19** at 9 am. What is an Operating System? *Read §10.1, 10.2, 10.3, 10.4.*
**Sun Apr 2** **Assignment 5** due at 23:59.
**Mon Apr 3** **Meeting 20** at 9 am. Introduction to Linux and the network stack. *Read §15.1, 15.2, 15.3, 15.4.*
**Tue Apr 4** **Check-in 8** due at 23:59.
**Wed Apr 5** **Meeting 21** at 9 am. Languages and protocols of the web. *Read §16.1, 16.2.* **Quiz 4.**
**Mon Apr 10** **Meeting 22** at 9 am. The relational data model. *Read §12.3.*
**Wed Apr 12** **Meeting 23** at 9 am. Introduction to SQL.
**Mon Apr 17** **Meeting 24** at 9 am. Planning and searching techniques. *Read §13.1, 13.2, 13.3.*
**Tue Apr 18** **Assignment 6** due at 23:59. **Check-in 9** due at 23:59.
**Wed Apr 19** **Meeting 25** at 9 am. The Turing Test and strong AI. *Read §13.4, 13.5, 13.6.* **Quiz 5.**
**Mon Apr 24** **Meeting 26** at 9 am. Authentication and authorization. *Read §17.1, 17.2.*
**Tue Apr 25** **Assignment 7** due at 23:59. **Check-in 10** due at 23:59.
**Wed Apr 26** **Meeting 27** at 9 am. Cryptography. *Read §17.4, 17.5.* **Quiz 6.**
**Mon May 1** **Meeting 28** at 9 am. Wildcard day and wrap-up.
**Wed May 3** **Final exam** at 8 am.
**Thu May 4** **Assignment 8** due at 23:59.