Syllabus
version 1.6

Instructor

David J. Malan '99
`malan@harvard.edu`
`http://www.cs.harvard.edu/malan/`
+1-617-523-0925

Description

Introduction to the intellectual enterprises of computer science and the art of programming. This course teaches students how to think algorithmically and solve problems efficiently. Topics include abstraction, algorithms, encapsulation, data structures, databases, memory management, security, software development, virtualization, and websites. Languages include C, PHP, and JavaScript plus SQL, CSS, and HTML. Problem sets inspired by real-world domains of biology, cryptography, finance, forensics, and gaming. Designed for concentrators and non-concentrators alike, with or without prior programming experience.

This course may be taken Pass/Fail or for a letter grade. When taken for a letter grade, this course meets the General Education requirement for Empirical and Mathematical Reasoning or the Core area requirement for Quantitative Reasoning.

Expectations

You are expected to attend all lectures and sections, submit nine problem sets, take two quizzes, and submit a final project.

Grades

This course may be taken Pass/Fail or for a letter grade. You may change from Pass/Fail to a letter grade or vice versa up until the fifth Monday of the term, per the Handbook for Students. The grade of Pass represents letter grades of A to D–.

All students, whether taking the course Pass/Fail or for a letter grade, must ordinarily submit all nine problem sets, take both quizzes, and submit a final project in order to be eligible for a passing grade (*i.e.*, Pass or A to D–) unless granted an exception in writing by the course's instructor.

Final grades will be determined using the following weights:

Problem Sets:[*]	50%
Quizzes:	40%
Final Project:	10%

Problem sets and the final project will be evaluated along axes of scope, correctness, design, and style.

Realize that some concentrations (*e.g.*, Applied Mathematics, Computer Science, Engineering Sciences, *et al.*) do not allow courses taken Pass/Fail to satisfy concentration requirements. Nor do courses taken Pass/Fail satisfy Core or General Education requirements. Take care to register for a letter grade or to change from Pass/Fail to a letter grade by the fifth Monday of the term if you must do so to meet some requirement.

Know that CS50 draws quite the spectrum of students, including "those less comfortable," "those more comfortable," and those somewhere in between. However, what ultimately matters in this course is not so much where you end up relative to your classmates but where you, in Week 12, end up relative to yourself in Week 0.

The course is not graded on a curve. The course does not have pre-determined cutoffs for final grades. Those less comfortable and somewhere in between are not at a disadvantage vis-à-vis those more comfortable. Scores on problem sets and the final project are normalized across sections at term's end. Each student's final grade is individually determined at term's end after input from the teaching fellows. Remarkable effort and upward trending do not go unnoticed.

---

[*] Although you must submit all nine problem sets, your lowest score among those problem sets on which you received a score of perfection for the axis of scope will be dropped when final grades are determined.

Books

No books are required for this course. However, you may want to supplement your preparation for or review of some lectures with self-assigned readings relevant to those lectures' content from either of the books below. The first is intended for those inexperienced in (or less comfortable with the idea of) programming. The second is intended for those experienced in (or more comfortable with the idea of) programming. Both are available online for free via PIN authentication. Both are available for purchase at the Coop and at sites like Amazon.com. Both have been placed on reserve at Cabot Science Library. Realize that free, if not superior, resources can be found on the course's website.

For Those Less Comfortable

Absolute Beginner's Guide to C, Second Edition
Greg Perry
Sams Publishing, 1994
`http://nrs.harvard.edu/urn-3:hul.ebookbatch.SAFAR_batch:9780131388185`
ISBN 0-672-30510-0

For Those More Comfortable

Programming in C, Third Edition
Stephen Kochan
Sams Publishing, 2004
`http://nrs.harvard.edu/urn-3:hul.ebookbatch.SAFAR_batch:9780768689068`
ISBN 0-672-32666-3

The book below is recommended for those interested in understanding how their own computers work for personal edification. It is not available online for free via PIN. But it is available for purchase at the Coop and at sites like Amazon.com. And it, too, has been placed on reserve.

How Computers Work, Ninth Edition
Ron White
Que Publishing, 2007
ISBN 0-7897-3613-6

This last book below is recommended for aspiring hackers, those interested in programming tricks and low-level optimization of code for applications beyond the scope of this course. It is available online for free via PIN authentication. It is also available for purchase at the Coop and at sites like Amazon.com. And it, too, has been placed on reserve.

Hacker's Delight
Henry S. Warren Jr.
Addison-Wesley, 2003
`http://nrs.harvard.edu/urn-3:hul.ebookbatch.SAFAR_batch:0201914654`
ISBN 0-201-91465-4

Website

The address of this course's website is:

```
https://www.cs50.net/
```

Visit the course's website to watch videos, to get help, to download handouts and software, and to follow links to other resources.

Lectures
```
https://www.cs50.net/lectures/
```

Lectures take place in Sanders Theater on Mondays and Wednesdays from 1:00pm until 2:30pm[†].

Each lecture is filmed and made available within 24 hours in streaming and downloadable formats (MP3 and MP4). Once posted, these recordings remain available until semester's end. You are welcome to watch or listen to a recording if you are unable to attend some lecture in person. You are encouraged to watch or listen to these recordings for the sake of review.

A schedule of lectures, subject to change, appears below.

> **Week 0**
> Wed 8/31, Fri 9/2
>
>> Introduction. Bits. Binary. ASCII. Programming. Algorithms. Scratch. Statements. Boolean expressions. Conditions. Loops. Variables. Threads. Events.
>
> **Week 1**
> Wed 9/7, Fri 9/9
>
>> C. Source code. Compilers. Object code. GCC. Functions. Comments. Standard output. Arithmetic operators. Precedence. Local variables. Types. Casting. Standard input. Libraries. Boolean expressions, continued. Conditions, continued. Loops, continued.
>
> **Week 2**
> Mon 9/12, Wed 9/14
>
>> Functions, continued. Global variables. Parameters. Return values. Stack. Frames. Scope. Arrays. Strings. Command-line arguments. Cryptography.

[†] This course will also meet on Fri 9/2 and Fri 9/9. Students with conflicts should watch those lectures online.

Week 3
Mon 9/19, Wed 9/21

Linear search. Binary search. Asymptotic notation. Recursion. Bubble sort. Selection sort. Debugging.

Week 4
Mon 9/26, Wed 9/28

Merge sort. Structures. Dynamic memory allocation. Pointers. Debugging, continued.

Week 5
Mon 10/3, Wed 10/5

CS50 Library. Heap. Pointers, continued. Forensics.

Week 6
Wed 10/12

Quiz 0 on Wed 10/12.

Week 7
Mon 10/17, Wed 10/19

File I/O. Linked lists. Stacks. Queues. Valgrind. Hash tables. Trees. Binary search trees. Tries.

Week 8
Mon 10/24, Wed 10/26

HTTP. HTML. CSS. PHP.

Week 9
Mon 10/31, Wed 11/2

PHP, continued. SQL.

Week 10
Mon 11/7, Wed 11/9

JavaScript. Ajax. APIs.

Week 11
Mon 11/14, Wed 11/16

Life after 50.

Quiz 1 on Wed 11/16.

Week 12
Mon 11/21

Exciting conclusion.

Sections
https://www.cs50.net/sections/

Lectures are supplemented by weekly, 90-minute sections led by the teaching fellows. Sections provide you with opportunities to explore and discuss course materials in a more intimate environment, with only your teaching fellow and a handful of classmates present, as well as to dive into hands-on activities.

Different sections are offered for those less comfortable, those more comfortable, and those somewhere in between.

Sectioning begins in Week 1. Sections themselves begin in Week 3, with course-wide supersections (open to all students) offered in Week 2.

A schedule of sections (and supersections) appears on the course's website.

Walkthroughs

On Sunday nights from 7:00pm until 8:30pm, the teaching fellows hold a "walkthrough" for the current week's problem set during which you receive direction on where to begin and how to approach the week's challenges.

Each walkthrough is filmed and made available within 24 hours in streaming and downloadable formats (MP3 and MP4). You are expected to attend or watch walkthroughs before asking questions about problem sets at office hours or via help.cs50.net.

Office Hours

`https://www.cs50.net/ohs/`

Office hours are opportunities for one-on-one assistance with problem sets alongside the course's teaching fellows and course assistants. Office hours are ordinarily offered four nights per week, per the schedule below. Also in attendance is free food and drink.

Mondays, 9pm – 12am:	Pforzheimer Dining Hall
Tuesdays, 9pm – 12am:	Leverett Dining Hall
Wednesdays, 9pm – 12am:	Quincy Dining Hall
Thursdays, 9pm – 12am:	Lowell Dining Hall

Problem Sets

`https://www.cs50.net/psets/`

Nine problem sets are assigned during the semester. Each is due via electronic submission six or more days after its date of distribution. However, you have four "late days" that you may "spend" during the semester, each of which provides you with an extension of twenty-four hours. You may spend no more than one late day on any particular problem set. You need not inform the staff of your use of a late day; usage of late days is tracked automatically. Lateness of electronic submissions is determined down to the minute by submissions' timestamps. Submitting more than seven minutes late is equivalent to submitting twenty-four hours late. Late work is not accepted once you have exhausted your late days, except in cases of emergency. Technical difficulties are not considered emergencies. These late days cannot be spent on the course's final project.

In order to accommodate students with different backgrounds, some problem sets are released in two editions: a standard edition intended for most students and a "Hacker Edition" intended for some students. Both editions essentially cover the same material. But the Hacker Edition typically presents that material from a more technical angle and poses more sophisticated questions. Hacker Editions are graded separately from standard editions, but those students who submit the former do not receive any form of extra credit outright. When determining grades at term's end, however, we do bear in mind submissions of Hacker Editions.

To be clear, we encourage most students (including aspiring computer scientists) to tackle the standard editions. However, you may choose, week to week, which edition to submit. You may not submit both or some amalgam of the two.

Although you must submit all nine problem sets, your lowest score among those problem sets on which you received a score of perfection for the axis of scope will be dropped when final grades are determined.

A schedule of problem sets, subject to change, appears below.

> **Problem Set 0: Scratch**
> due by noon on Thu 9/8

> **Problem Set 1: C**
> due by noon on Thu 9/15

> **Problem Set 2: Crypto**
> due by noon on Thu 9/22

> **Problem Set 3: The Game of Fifteen**
> due by noon on Thu 9/29

> **Problem Set 4:** 数独
> due by noon on Thu 10/6

> **Problem Set 5: Forensics**
> due by noon on Thu 10/20

> **Problem Set 6: Mispellings**
> due by noon on Thu 10/27

> **Problem Set 7: C$50 Finance**
> due by noon on Thu 11/3

> **Problem Set 8: CS50 Shuttle**
> due by noon on Thu 11/10

Quizzes
`https://www.cs50.net/quizzes/`

The course has two 75-minute quizzes. These quizzes are "closed-book," but you may utilize during each quiz one two-sided page (8.5" × 11") of notes, typed or written, and a pen or pencil, nothing else.

When final grades are computed, your scores on these two quizzes are weighted equally.

A schedule of quizzes, subject to change, appears below; these quizzes take place in lieu of lectures on these dates.

> **Quiz 0**
> Wed 10/12

> > Covers weeks 0 through 5.

Quiz 1
Wed 11/16

Covers weeks 0 through 10 with emphasis on 7 onward.

Unless arranged with the staff in advance, quizzes may not be taken at alternative times even if missed by accident, except in cases of emergency.

Final Project

The climax of this course is its final project. The final project is your opportunity to take your knowledge of programming out for a spin and develop your very own piece of software. So long as your project draws upon this course's lessons, the nature of your project is entirely up to you, albeit subject to the staff's approval. You may implement your project in any language(s) as long as the staff approves. You are welcome to utilize infrastructure other than the CS50 Appliance and `cloud.cs50.net`, provided the staff ultimately has access to any hardware and software that your project requires. All that we ask is that you build something of interest to you, that you make something useful, that you solve an actual problem, or that you somehow impact campus. Strive to create something that outlives this course.

Inasmuch as software development is rarely a one-person effort, you are allowed an opportunity to collaborate with one or two fellow students for this final project. Needless to say, it is expected that every student in any such group contribute equally to the design and implementation of that group's project. Moreover, it is expected that the scope of a two- or three-person group's project be, respectively, twice or thrice that of a typical one-person project. A one-person project, mind you, should entail more time and effort than is required by each of the course's problem sets.

Guidelines for the final project will be distributed by Mon 10/24. A schedule, subject to change, appears below.

Pre-Proposal
due by noon on Mon 11/7

Proposal
due by noon on Mon 11/14

CS50 Hackathon
from 8:00pm on Fri 12/2 until 7:00am on Sat 12/3

Status Report
due by noon on Mon 12/5

Implementation
due by noon on Thu 12/8

CS50 Fair
from 11:00am until 4:30pm on Fri 12/9

Extensions on the final project are not granted, except in cases of emergency. Technical difficulties are not considered emergencies. Problem sets' late days cannot be spent on the final project. Lateness of submissions is determined down to the minute by submissions' timestamps. Submitting more than seven minutes late is equivalent to not submitting at all.

CS50 Hackathon

`http://hackathon.cs50.net/`

From 8:00pm on Fri 12/2 until 7:00am on Sat 12/3 is the CS50 Hackathon, an epic all-nighter during which you can dive into your final project's implementation alongside classmates and staff. If you choose to partake, you'll be asked to propose three milestones for yourself that evening: a "good" one that you intend to achieve no matter what; a "better" one that you think you can achieve; and a "best" one that you hope to achieve.

Pizza will be served at 8:00pm; Chinese food will be served at 1:00am; and those still standing at 5:00am will be treated to breakfast at IHOP.

CS50 Fair

`http://fair.cs50.net/`

From 11:00am until 4:30pm on Fri 12/9 is the CS50 Fair, an epic display of final projects. Not only is the fair a venue at which to see classmates' projects and demo your own, it is an opportunity to mingle with students, faculty, and staff from across campus as well as recruiters from tech companies. Attendance is expected of all students.

Also in attendance will be popcorn and candy, CS50 stress balls, and a raffle with (fabulous) prizes. Family and friends are welcome to join.

Academic Honesty

All work that you do toward fulfillment of this course's expectations must be your own unless collaboration is explicitly allowed in writing by the course's instructor. Collaboration in the completion of problem sets is not permitted unless otherwise stated by some problem set's specification.

Viewing or copying another individual's work (even if left by a printer, stored in an executable directory, or accidentally shared in the course's virtual terminal room) or lifting material from a book, website, or other source—even in part—and presenting it as your own constitutes academic dishonesty, as does showing or giving your work, even in part, to another student or soliciting the work of another individual. Similarly is dual submission academic dishonesty: you may not submit the same or similar work to this course that you have submitted or will submit to another. Nor may you provide or make available solutions to problem sets to individuals who take or may take this course in the future. Moreover, submission of any work that you intend to use outside of the course (*e.g.*, for a job) must be approved by the staff.

You are welcome to discuss the course's material with others in order to better understand it. You may even discuss problem sets with classmates, but you may not share code. In other words, you may communicate with classmates in English, but you may not communicate in, say, C. If in doubt as to the appropriateness of some discussion, contact the course's instructor.

You may turn to the Web for instruction beyond the course's lectures and sections, for references, and for solutions to technical difficulties, but not for outright solutions to problems on problem sets or your own final project. However, failure to cite (as with comments) the origin of any code or technique that you do discover outside of the course's lectures and sections (even while respecting these constraints) and then integrate into your own work may be considered academic dishonesty.

All forms of academic dishonesty are dealt with harshly. If the course refers some matter to the Administrative Board and the outcome for some student is *Admonish*, *Probation*, *Requirement to Withdraw*, or *Recommendation to Dismiss*, the course reserves the right to impose local sanctions on top of that outcome for that student that may include, but not be limited to, a failing grade for work submitted or for the course itself.

Terms and Conditions

By enrolling in this course, you authorize Harvard University and Computer Science 50, and anyone that Harvard or Computer Science 50 may permit, to film, videotape, audio record, and photograph you during Computer Science 50 activities for subsequent broadcast or other dissemination in perpetuity through any media, which includes, without limitation, commercial and public radio, television, cable, and the Internet. And you acknowledge that you might not receive a copy of any film, videotape, audio recording, photograph, or computer file that is or may be produced.