CAS CS101: Introduction to Computing
2019.FALL.SYLLABUS.1.0

Perry Donham / perryd@bu.edu / 202-567-7362
Mailslot: 111 Cummington / MCS138
Blog: https://sites.bu.edu/perryd Twitter: @perrydBUCS
Office: PSY228C (64 Cummington Mall)

About the course
In the old days one was *literally* introduced to computers. I recall being escorted into a large room full of large, noisy equipment festooned with blinking lights (the equipment, not me) and being told, "This is the computer. Don't touch it." Singular, of course, since we only had one. We've gone from a time when computing was magic and astounding to a time when almost everything that we interact with has embedded computers and we really don't think about them much. And yet the fundamental machinery, both in the hardware and software sense, hasn't changed since that earlier time. Sure, computers are smaller and faster and cheaper, but they are still recognizable as descendants of those first machines.

This course is intended to give you a broad look at how computers work from both a hardware and a software perspective. It also provides a survey of what computer science is, and what computer professionals do. The material will be a blend of theory and practical application. After completing CS101 you should have a better understanding of how the devices and services that you use every day work, and how to better use them in your own life. You'll also pick up some useful computing skills in the area of interactive web page design using HTML and CSS, and Javascript programming.

Text
Our text this semester is a book that I've written for the course:

Introduction to Computer Science, 1st ed., Donham, Cognella Academic Press
ISBN 9781634876735

The book is available at a discount through the publisher's web site at https://store.cognella.com/81560-1B-NI-001 in both ebook and print formats, and also the BU bookstore and on Amazon. The publisher's website is the least expensive option.

Blackboard
We use Blackboard (https://learn.bu.edu) as a repository for the slide sets for each class, homework and lab assignments, sample code, and announcements. You should be enrolled already, so that when you log on to the site you'll see the course listed. Piazza.com will be our tool for questions and discussion; a link to it will be on the Blackboard page. I'll post each week's lecture materials as PDFs on Blackboard for your reference.

You'll use Blackboard to turn in homework, which for the most part will be in the form of lab exercises, and to do weekly quizzes.

Homework/Lab Assignments
CS101 is a lecture/lab course, which means that part of your time will be spent completing lab exercises. Your lab section will meet each Wednesday. A teaching assistant (TA) will conduct lab

sections and will be available for tutoring; their hours will be posted on Blackboard. We also have two undergraduate assistants (UAs) who can help you with questions. You are expected to use your own laptop during labs, but if you don't have a computer you are welcome to use the undergraduate computing lab in EMA-302. A good strategy is to start each assignment right away so that you can use lab time to get advice. There usually is someone in EMA-302 that you can ask for help if you get stuck (a tutoring schedule will be posted later in the semester), or you can post a note on Piazza.

Generally speaking, labs and homework will be due one week after they are assigned. Assignments turned in after the due date will be docked 10 points (out of a total of 100 points). There are no make-ups for labs.

Quizzes
There are 11 or 12 scheduled quizzes for the semester. Each will typically have 5 questions and will cover the material we discussed in Tuesday's lecture; occasionally there will be a question or two from the prior Thursday. The quizzes are taken online in Blackboard and you'll do them during your lab time on Wednesdays. I'll drop your lowest quiz grade at the end of the semester, so if you are ill or for some reason can't make it to lab you are covered at least once. There are no make-ups for quizzes.

Midterm and Final Exam
The midterm exam covers first-half material, and the final covers the second-half. Each is roughly 50 to 60 questions, for the most part multiple-choice, with a small number of fill-in-the-blank. Exams are taken on paper in class.

Grades
The allocation for your final grade looks like this:

Homework/labs	25%
Quizzes	25%
Midterm exam	25%
Final exam	25%

Grades are not negotiable, but if you think that a mistake was made in grading, we'll take a look at it together. The course grading scale is numerical:

96-100	A	80-84	B	65-69	C
90-95	A-	75-79	B-	60-64	C-
85-89	B+	70-74	C+		

If your course grade happens to be close to a boundary, such as an 89.5, I'll bump you up unless your overall course work for some reason doesn't justify it.

Attendance
The format of the course is that Tuesday's lecture will be on something you will work on in lab on Wednesday, and you'll also take a quiz on Wednesday covering material from the lectures from the prior Thursday and Tuesday. Since quizzes are 25% of your grade, it pays to attend both lectures.

Help With Homework
We want you to succeed in this class, and if you are stuck on something, email me or your teaching assistant or drop by my or his / her office hours. You can also post a question on our Piazza forum … that's usually the fastest way to get information. There are lots of ways that we can help, but you need to ask. Our teaching assistants and undergraduate assistants will hold tutoring hours during the week, which will be posted on Blackboard, as will be my office hours.

Contacting me and office hours
The best way to contact me is by email at perryd@bu.edu. You may also IM me at perryd@bu.edu (iMessage). My office is in the Psychology building at 64 Cummington Mall, room PSY228C. Office hours are posted on Blackboard. No appointment needed, just drop by if you have a question or want to hang out a bit. If you need to drop something off, my mail slot is in the CS office in MCS138.

Using Encrypted / Signed Messaging
One of the topics we'll discuss this semester is computer and network security, including encryption. If you'd like to get some practice in using encryption tool, you are invited to send encrypted and/or digitally signed email to me at perryd@bu.edu. On Macs, the GPGSuite at gpgtools.org is a good choice, as is Thunderbird with the Enigmail plugin; for Windows users you can try gpg4win.de. My public encryption key is available at https://pgp.mit.edu, and my key fingerprint/ID is C894 B69B 6576 C394 1452 2E9E 7C38 F315 BCC1 ADDF.

I also am a proponent of the encrypted IM app Signal; if you'd like to practice with it I'm happy to help you get it set up.

Academic Conduct Code
The University the College take cheating very seriously. Cheating and plagiarism will not be tolerated in any course. Cases will be referred to the Dean's office and may result in loss of credit for an exam or assignment or other disciplinary action.

That said, by its nature programming is a collaborative effort, and I fully expect that you will use resources such as Google, fellow students, and our own discussion forum on Blackboard to learn the material and do your assignments. We'll discuss code and approaches in class, and I'll occasionally post sample code on Blackboard that you are welcome to use as a starting point. However, I definitely don't want you to simply copy entire programs that you find on the web and turn them in as your own work. If you do use more than a line or two of someone else's code (including mine), just make a note in a comment in your program to point to where you got it. Dr. Sullivan's guidelines for collaboration are a good explanation of our expectations. You can read them at http://cs-people.bu.edu/dgs/courses/cs111/collaboration.html.

BU Hub Learning Outcomes
This course satisfies Quantitative Reasoning: Quantitative Reasoning II and Communication: Digital/Multimedia Expression.

Quantitative Reasoning II

QR II Outcome 1: Students will learn to program in Javascript as part of their development of a web page project; this involves solving a number of non-trivial problems in data structures and

algorithms. You must define a framework for the solution, execute it, and verify that it is correct and flexible. This is the foundation for computational thinking.

QR II Outcome 2: Students will answer discipline-specific questions with regard to design and execution of web scripts, involving storage of data and appropriate algorithms to manipulate the data. Students will also engage with one of the major questions of our time: How to keep information secure in a connected world? Students will understand the options involving cryptographic protocols, and understand their strengths and weaknesses.

QR II Outcome 3: This outcome uses the word "argument" but in math and CS the term "algorithm" may be used equivalently. In this class, students will formulate algorithms and test their efficiency and correctness using quantitative and logical tools. For example, choosing appropriate cryptographic protocols involves a consideration of efficiency, generality of the solution, and societal impacts.

Quantitative II Outcome 4: Students will create web pages with active content using Javascript. Quantitative information is displayed using charts, tables, or graphs, depending on the application.

QR II Outcome 5: Students will learn about the efficiency of algorithms and their appropriateness in various contexts. Web scripts and programs are evaluated as to their strengths and weaknesses in their context; cryptographic protocols are discussed in terms of advantages and risks, and students will understand ultimately that all algorithms have limitations and risks.

Digital/Multimedia Expression

Digital/Multimedia Outcome 1: Students will do several assignments involving the creation of a web page with active content written in Javascript. These project will use a variety of media (written, numeric, pictures, audio, or video) and using appropriate modes of expression for the web.

Digital/Multimedia Outcome 2: Student web pages will exhibit the various modes of expression in electronic media, and will understand the capabilities of the web for presenting information effectively; in particular, students will understand how to create a "user experience" for communicating on the web.

Digital/Multimedia Outcome 3: Web pages using HTML and CCS, and implementing active content in Javascript, inherently involve a variety of aesthetic and practical choices in designing the user experience. Interactive content is essential to this experience, and audio-visual presentation of information, both qualitative and quantitative, is inherent to the web design process. In short, web design and the programming technologies involved are the ultimate framework for digital communication, and this course involves all these