Boston University
College of Arts and Sciences
Computer Science Department

CS 108: Application Programming, Fall Semester 2019
Course materials on Blackboard: http://learn.bu.edu

<table>
<thead>
<tr>
<th>Classroom Meetings @ LSE B03</th>
<th>Lab Sections @ FCB 104</th>
</tr>
</thead>
<tbody>
<tr>
<td>Mondays and Wednesdays 10:10-11:25am</td>
<td>Section A2 Wednesday 1:25pm-2:15pm</td>
</tr>
<tr>
<td></td>
<td>Section A3 Wednesday 2:30pm-3:20pm</td>
</tr>
<tr>
<td></td>
<td>Section A4 Wednesday 3:35pm-4:25pm</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Instructor</th>
<th>Teaching Fellow</th>
</tr>
</thead>
<tbody>
<tr>
<td>Aaron Stevens, <a href="mailto:azs@bu.edu">azs@bu.edu</a>. Always include “CS108” in email subject.</td>
<td>Qiancheng Fu (<a href="mailto:qcfu@bu.edu">qcfu@bu.edu</a>) Tutoring Hours held at TBD Mon 12-2pm; Thu 4-6pm; Fri 9-11am.</td>
</tr>
<tr>
<td>Office Hours at QST 546F: Mon 1-2pm; Thu 12:45-1:45pm and by appointment: <a href="http://aaronstevens.youcanbook.me">http://aaronstevens.youcanbook.me</a>.</td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Undergraduate Assistants</th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>Caroline Barone (<a href="mailto:cabarone@bu.edu">cabarone@bu.edu</a>)</td>
<td>Susritha Kopparapu (<a href="mailto:skoppara@bu.edu">skoppara@bu.edu</a>) Tutoring Hours held at TBD To be announced, see Blackboard.</td>
</tr>
</tbody>
</table>

**Course Description**

CS108 is an introduction to programming suitable for students not majoring in computer science. CS108 will cover a broad set of topics about application development, including: basic programming concepts, application decomposition and abstraction; data manipulation and programming with databases; graphical user interfaces; and debugging. Assignments will include applications in science, commerce, games, the Web, and social networking.

Our emphasis in teaching programming in this course centers on the following:

- Problem analysis: organizing problems from our world into the structure, syntax, and semantics in which a computer solution can be applied.
- Fundamentals of procedural programming: writing functions, using pre-existing modules, flow-of-control, and writing interactive programs.
- A brief introduction to object oriented programming, including the use of objects from the Python API and creating user-defined classes.
- The role of data structures and databases in application programming, including the use of lists, tuples, dictionaries, and SQL.
- The structure and development of web applications, which generate dynamic content for the World Wide Web, social networking, and e-commerce.
**Books and Software**

**Required:** How to Think Like a Computer Scientist with Python 3
http://openbookproject.net/thinkcs/python/english3e/

There is a possibility that the purchase of an addition electronic book will be required. Details to be confirmed.

Daily reading assignments will be posted on Blackboard.

We will be using the Top Hat (www.tophat.com) classroom response system in class. You will be able to submit answers to in-class questions using Apple or Android smartphones and tablets, laptops, or through text message. An email invitation will be sent to you by email, but if you don’t receive this email, you can register by simply visiting our course website: https://app.tophat.com/e/598319

Top Hat will require a paid subscription, and a full breakdown of all subscription options available can be found here: www.tophat.com/pricing.

You can visit the Top Hat Overview (https://success.tophat.com/s/article/Student-Top-Hat-Overview-and-Getting-Started-Guide) within the Top Hat Success Center which outlines how you will register for a Top Hat account, as well as providing a brief overview to get you up and running on the system. Should you require assistance with Top Hat at any time, due to the fact that they require specific user information to troubleshoot these issues, please contact their Support Team directly by way of email (support@tophat.com), the in app support button, or by calling 1-888-663-5491.

**Python Software:** We will be using the Python programming language version 3.7.4, which you can download for free online: http://www.python.org/download/.

We will also use the Microsoft VS Code Integrated Development Environment, which can be downloaded here: https://code.visualstudio.com/download

**Piazza Discussion Board**

Students should post questions about examples, tools, and syntax issues on our class' Piazza website: http://piazza.com/bu/fall2019/cs108

You may post small code snippets (2-3 lines is usually OK) but you must not post entire code files with your solutions to the assignments.

**What You Need to Know About Computer Programming**

I believe anyone can succeed at learning to program. This is a first course in computer programming, and there are no formal pre-requisites. The only expectation of students' computer skills before taking this class is to be comfortable with using email, web browsing, and copying and pasting text. In addition, familiarity with high school-level algebra (e.g. MA 118) is assumed.

In addition, you will need time, and this is more important than you can imagine.

Computer programming, like farming, takes time. There are many steps in farming (e.g., seeding, watering, fertilizing, waiting, and harvesting), which must be completed in order. Each step takes time, and it’s simply not possible to cram farming into a shorter time horizon. Programming also has many steps (e.g., understanding the problem, designing an algorithm, writing code, testing code, rewriting code, retesting code, and checking that the solution meets the requirements to solve the problem). Learning to program is similar to farming, and again, it is not possible to skip or rush some steps and cram the learning into a shorter amount of time.

---

1 Adapted from text in “Core Techniques and Algorithms in Game Programming,” Daniel Sanchez-Crespo Dalmau.
**Our Teaching/Learning Method**

Our teaching-learning approach in CS108 that of a **programming workshop**, wherein students spend class time actively doing programming. Our expectation is that you will do most of the independent programming assignments (what used to be called “homework”) in the workshop classroom, so that you will have access to the teaching staff while working on the assignments.

How can we spend class time working on assignments, when you also need to learn new concepts, programming constructs, and syntax for each assignment? This is where pre-class preparation and self-study is crucial. Learning to program is a skill that takes practice, similar to learning a sport or a musical instrument. You cannot learn to play basketball by simply watching the Celtics; rather, you need to go to the gym and practice your shots. To make the best use of your at-home and in-class time, CS108 uses the flipped-classroom teaching/learning methodology. You will be responsible for content-acquisition before coming to class, and we will use in-class time for learning through programming assignments.

Before each class, you will:

- Log into Blackboard to access the Learning Module for the class (under “Course Content”).
- Read the relevant textbook sections to be introduced to vocabulary, concepts, and examples.
- Watch 2-3 short videos (approx. 20-30 minutes total) that present mini-lectures and examples relevant to the new material. Take notes on new concepts/syntax as you go.
- **Write up the examples on your own**, solving any syntax or logical issues you encounter, and getting the code to work on your computer. You will submit the completed examples as a small part of your graded work for the class.

*Your preparation before class is crucial to your success on the assignments.*

In class, we will:

- Discuss your questions about the examples. Come to class prepared with your questions.
- Have a short multiple-choice quiz about the material in the videos and examples.
- Discuss the assignment for the day, what you should work on first/last, and answer questions about the concepts involved.
- Use most of the class time as “workshop time” for you to work independently on the assignments. Students who have adequately prepared the examples before class will be able to finish most of the assignments during class time.

After class, you will:

- Complete the assignment and submit online by midnight the day after class.

**Grading**

The following percentages are tentative and may be changed at the instructor’s discretion at any time:

<table>
<thead>
<tr>
<th>Component</th>
<th>Percentage</th>
</tr>
</thead>
<tbody>
<tr>
<td>Pre-class preparation and in-class clicker quizzes</td>
<td>10%</td>
</tr>
<tr>
<td>Programming assignments (about 21)</td>
<td>30%</td>
</tr>
<tr>
<td>Final project</td>
<td>20%</td>
</tr>
<tr>
<td>Practical programming quizzes* (4)</td>
<td>40%</td>
</tr>
<tr>
<td>Attendance: I will award up to 1 bonus point for perfect or near perfect attendance</td>
<td>1% BONUS</td>
</tr>
</tbody>
</table>

* No make-up quizzes are not allowed. See notes below under “Policies”

To earn a passing grade in CS108, you must earn a grade of at least 60% on each sub-component (pre-class preparation, clicker questions, programming assignments, final project, and programming quizzes).

---

Getting Help With Assignments

The assignments in this class will be very challenging and often time consuming. To be most successful with the assignments, you must be adequately prepared for each class meeting by having completed all of the online videos, prepared the code examples, and read the required readings. You cannot expect to be successful on the assignments without adequate preparation. We will be working on the assignments in class, so plan your time so you can ask questions in class. You can also get assistance in the labs and tutoring hours.

When you get stuck, the instructor, teaching fellow, and undergraduate assistants are here to help you. The best avenue for getting help is during the classroom time.

Outside of class time, you may also post questions on our Piazza class discussion site. Always be specific about the problem you are having (e.g., “I have a syntax error at line 42. Would you please explain it to me?”), and include the line of code that is causing the problem (but do not post your entire code file, function, etc., as this would violate the academic conduct policy of the class!)

Preparing for Practical Quizzes

The practical quizzes will require you to solve some simple programming problems to demonstrate your learning. You will write code from scratch, without notes, examples, or previous assignments. The level of difficulty will be similar to the pre-class examples (rather than the more elaborate/complicated assignments).

The most effective strategy is active practice while learning the new material the first time. That is, you must independently write and debug every line of code that is shown in any examples, and you should practice doing these kinds of example problems without references. Reading the book, notes and examples alone will give you a false sense of familiarity. Additional detail about the types of problems to expect will be given in class before each of the lab exams.

After each class meeting, review the class notes and examples, and especially the points on which we elaborate. You should develop your own practice problems that are similar to the pre-class examples, and practice implementing them on your own. For example, you could define a problem as “write a function to calculate the volume of a cylinder,” (you will understand what this means before the first lab exam), and then practice writing the code for a solution. The great news about computer programming is that you will know with certainty if your solution is correct, because you can test it thoroughly and verify the results.

Assignment Grade Review Policy

Programming assignments will be submitted via Blackboard. In general, assignments will be graded within 5 days of submission. Grading will follow a standardized rubric and include comments about any point deductions.

Please do not bring up grade questions during class or tutoring hours, as this would take time away from helping you and other students with programming assignments. Instead, if you have questions or concerns about a grade on an assignment, you should email the Teaching Fellow to request a review. The TF will process grade reviews within 3-4 days and respond by email.

Requests for grade reviews must be submitted by email within 2 weeks after the assignment was graded. Requests for grade reviews submitted more than 2 weeks after an assignment is graded will not be processed. Do not wait until the end of semester, as this is a very busy time for everyone.
Collaboration Policy and Individual Work

Collaboration is recommended in understanding programming concepts and system features. You are encouraged to discuss the examples, the homework problem statements, algorithms and expected output, and to seek and receive help with the Python programming language, syntax, and any IDE or other debugging tools.

However, each student must write their own code and other deliverables independently.

What is plagiarism?

It is the student’s responsibility to know and understand the provisions of the CAS Academic Conduct Code, which is available here: https://www.bu.edu/academics/cas/policies/academic-conduct/

With respect to CS108, plagiarism is specifically defined to include (but is not limited to) the following:

- collaboration on the code you write (unless you are specifically instructed to do so on a specific task)
- copying any part of someone else’s program, even if you have permission and/or have modified the code
- sharing or giving your code or even a subset of your code to another student to review
- reviewing another student’s solution (including from past semesters)
- excessive collaboration (i.e., writing your solution together)

All CS108 homework assignments are independent work. It is course policy to use automatic plagiarism detection software, and suspicious similarities will be uncovered.

Incidents of academic misconduct will be reported to the Academic Conduct Committee (ACC). The ACC may suspend/expel students found guilty of misconduct. At a minimum, students who engage in misconduct will have their final grade reduced by one letter grade (e.g., from a B to a C).
Administrative Policies

The official administrative business of this class will be conducted by email. Grade questions or requests for review, notification of absence, etc. will be processed via email so that we both have a written record of what was agreed.

Attendance and discussion/asking questions are expected and will be reflected in your grade. If you must be absent, please email me in advance to let me know why you won’t be in class, and to let me know what you will do to keep up with the assignments. CS108 is not a correspondence course. Inadequate attendance is sufficient grounds for a grade of F.

Labs section times will be used for optional office hours with the Teaching Fellow, unless required for a make-up class. In general, you may attend ANY lab section time to meet with the teaching fellows.

Assignments are due on the date stated on the homework assignment (to be posted on web).
• Assignments received within 0-24 hours of the deadline will be accepted with a 10% penalty.
• Assignments received more than 24 hours after the deadline will not be graded.

There will be no make-up practical lab quizzes.
Grades for missing quizzes are computed as follows:
• If you miss one quiz, you will receive 80% of your average grade on the other quizzes
• If you miss 2 quizzes, you will receive 70% of the grade from the remaining 2 quizzes
• If you miss 3 quizzes, you will receive 60% of the grade for the quiz you did take.
• If you miss all 4 quizzes, you will not pass the class.

No special make-up work will be accepted after the end of the semester. Don’t even ask.
In the event of a documented major medical problem, a grade of Incomplete will be given pending the submission of complete work. However, make up work “to improve one’s grade” will not be accepted.

It is the student’s responsibility to retain all papers, quizzes, and exams that have been graded and returned. Should these original documents not be available in the event of a grade dispute, I will need to defer to my own records.

Requests for review or re-grading of lab exams or assignments should be brought to your TF or instructor by email (preferred) or in office hours, no later than 2 weeks after the exam/homework has been returned, and absolutely no later than the last class day of the semester.

Grades are not negotiable. Don’t even ask – just do the work and you’ll get the grade you deserve. Of course, please bring any clerical grading errors to my attention by email and I will gladly fix them.

Withdrawing from the Course
If you feel that you want to drop or withdraw from the class, please come talk to me about it as early as possible; I want to help you succeed, but you need to ask for help.
The last day to drop a class (without a “W” grade) is October 7, 2019.
The last date to withdraw and receive a “W” grade is November 8, 2019.