Classroom Meetings
Tuesday and Thursday
9:30am – 11:00 am @ EMA 302

Lab Sections:
Section A2 Thursday 12:00pm - 1:00pm
Section A3 Thursday 1:00pm - 2:00pm
Section A3 Thursday 6:00pm - 7:00pm
@ CS teaching lab (EMA 304)

Instructor
Aaron Stevens, azs@bu.edu
Always include “CS108” in email subject.
Office hours @ PSY 228B:
Tuesday 11am-12pm, Wednesday 2-3pm,
Thursday 2-3pm, and by appointment:
http://aaronstevens.youcanbook.me

Teaching Fellow
Sachin Vasant, sachinv@bu.edu
Tutoring Hours: Tuesday 2-5pm and Friday 2-5pm,
and by appointment.

Undergraduate Assistant
Akhil Saji, asaji@bu.edu
Tutoring Hours:
Tuesday and Thursday, 11am-12pm
Tutoring hours held at EMA 302

Course Description
CS108 is an introduction to programming suitable for students not majoring in computer science. Topics include: 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 (input, process, output); flow of control; use of pre-existing modules.
- Conceptual understanding of object oriented programming, including the use of objects from third-party code libraries.
- The role of data structures and databases in application programming, including the use of lists, tuples, dictionaries, and SQL.
- The development of graphical user interfaces, which provide windows, icons, menus, and a point-and-click interface to program functionality.
- The structure and development of web applications, which generate dynamic content for the World Wide Web, social networking, and e-commerce.
Teaching/Learning Method

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 will employ the flipped-classroom model.¹ 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 “Content”).
• Read the relevant textbook sections to be introduced to vocabulary, concepts, and examples.
• Watch a few short videos that present mini-lectures and examples relevant to the new material.
• **Re-create the examples on your own**, solving any syntax or logical issues you encounter. You will submit the completed examples on Blackboard as part of your graded work for the class.

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

In class, we will:
• Begin with a brief discussion of the examples. I will not simply “read” or “review” the examples. Come prepared with your questions, or else we will skip this part.
• Discuss the assignment for the day, and what you should work on first/last, and answer questions about the concepts involved.
• Use most of the class 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 to Blackboard by the day after class.
• Take a short online quiz on Blackboard to prove to yourself that you have acquired the basic concepts, vocabulary and syntax. We can discuss questions from the online quizzes during the following class.

What You Need to Know About Computer Programming²

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. Many people believe that computer programming is extremely difficult, and that the code is written in some arcane syntax understandable only by experts. Although some parts of the process are indeed complex, most of the source code required for homework assignments can be easily understood.

So, what makes programming so hard? It’s not the difficulty: It’s the time required to achieve any decent results. The homework assignments will take time, so make sure you have plenty of it.

---

² Adapted from text in “Core Techniques and Algorithms in Game Programming,” Daniel Sanchez-Crespo Dalmau.
Books and Software

Available on reserve at the Science and Engineering Library.

Online will be assigned for materials not covered in the textbook.

Books and Software

Available on reserve at the Science and Engineering Library.

Online will be assigned for materials not covered in the textbook.

Online Python Resources:
- Python Documentation
  http://www.python.org/doc/
- A Byte of Python (an online wikibook)
  http://swaroopch.info/text/Byte_of_Python:Main_Page

Python Software:
We will be using the Python programming language version 3.3.x, which you can
download for free online: http://www.python.org/download/.
It is available on the lab computers at EMA 302 (the undergraduate CS lab).

Why Python?

Python is a remarkably powerful dynamic programming language that is used in a wide variety of
application domains. Some of its distinguishing features include:

• very clear, readable syntax
• natural expression of procedural code
• intuitive object orientation; very high level dynamic data types
• extensive standard libraries and third party modules for virtually every task

Grading

The following percentages are tentative and may be changed at my discretion at any time:

<table>
<thead>
<tr>
<th>Task</th>
<th>Percentage</th>
</tr>
</thead>
<tbody>
<tr>
<td>Prepare Examples Before Class</td>
<td>5%</td>
</tr>
<tr>
<td>Programming Assignments (about 23)</td>
<td>35%</td>
</tr>
<tr>
<td>Online Quizzes (one per class)</td>
<td>5%</td>
</tr>
<tr>
<td>Written Quizzes (6) and the Final Exam*</td>
<td>25%</td>
</tr>
<tr>
<td>Lab Exams (2 – practical Python programming)</td>
<td>15%</td>
</tr>
<tr>
<td>Final Project</td>
<td>15%</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>

* The written final exam is optional. You may elect to use the average of your in-class quizzes in place of
writing the final exam.

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 Thursday, February 20, 2014.
The last date to withdraw and receive a “W” grade is Friday, March 28, 2014.
How to Prepare for Written Quizzes and the Final Exam

The most effective strategy is active recall. Reading the book, notes and examples alone will give you a false sense of familiarity. Instead, you must actively digest the class material, and practice your recall of it in question and answer form.

After each class meeting, review the class notes and examples, and especially points that I spend time elaborating upon. For each of these “main points” (there are probably about 4-6 of these per class meeting) you should write out a challenging question about the point, and prepare the answer.

By actively preparing questions that you know you can answer, you will be surprised how many of those show up on the quiz/exam.

Getting Help With Homework

The homework assignments in this class will be very challenging and often time consuming. Work on homework assignments as soon as they are given. Plan your time so you can ask questions in class and get assistance in the labs and tutoring hours.

If you get stuck, the teaching fellow and instructor are here to help you. The best avenue for getting help is to come to office hours and/or the scheduled lab times. You may also email your TF or instructor, and emails will be answered within 24 hours (often sooner). Always attach your work-in-progress code to your email, with specific questions (i.e., “I have a syntax error at line 42. Would you please explain it to me?”).

Quiz Rewrite Policy

My goal is to help you do your best work, and for you be graded on your best work. If you are not satisfied with your quiz grade, you have the option to count your final exam score in its place. I recommend this option for anyone who receives a grade less than 60 on a particular quiz, but the option is available to anyone who wants to do the re-write work.

• To earn this option, you must re-write the quiz as a take-home assignment and email your answers to me.
• Please use the title "CS108 QUIZ REWRITE FOR LASTNAME, FIRSTNAME" in your email.
• Quiz rewrites are due by the Friday after the quiz was returned in class.

Frequently Asked Questions
Q. How many quizzes may I rewrite?
A. You may rewrite as many quizzes are you choose.

Q. Does a quiz re-write mean the final exam is required?
A. Writing the final exam is required to replace your quiz grade with your exam grade. However, you can still choose not to take the final exam, and keep your original quiz grade(s).

Example
Suppose you received a 35 on quiz 2. I recommend that you rewrite this quiz as a take-home quiz. I will return your email with brief feedback. Suppose that you write the final exam and receive a grade of 80. The final exam score of 80 will count for the final exam grade, and also replace the quiz 2 grade of 35. However, at the end of the semester you may still choose to not take the final exam, in which case your quiz 2 grade would remain 35.
Assignment Grade Review Policy

Programming assignments will be submitted via Blackboard. In general, assignments will be graded within 5 days of submission. The grader 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.

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.

Lab attendance is optional most weeks, but required for 2 scheduled lab exams.

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 within 24-48 hours of the deadline will be accepted with a 20% penalty.
- Assignments received more than 48 hours past the deadline will not be graded.

There will be no make-up quizzes or exams.
If you have to miss a quiz for a medical reason or other extreme circumstances, you must inform me in advance. You will have the opportunity to write the quiz as an ungraded take-home assignment and then count your final exam score in its place.

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 quizzes or assignments should be brought to your TF or instructor by email (preferred) or in office hours, no later than 2 weeks after the quiz/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.
Plagiarism, Collaboration, and Collusion

All CS108 homework assignments are independent work.

It is the student’s responsibility to know and understand the provisions of the CAS Academic Conduct Code, copies of which are available in room CAS 105.

In addition to the definition of plagiarism in the CAS Academic Conduct Code, with respect to CS108, plagiarism is specifically defined to include (but is not limited to) the following:

- collaboration on the code you write
- 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)

It is my policy to use automatic plagiarism detection software, and suspicious similarities will be uncovered. I am required by Boston University and the College of Arts and Sciences to refer cases of academic misconduct to the Dean’s Office. The University takes acts of cheating and plagiarism very seriously; first time violators are routinely suspended for a semester.

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

However, each student must write his or her own code and other deliverables independently.