CAS CS 111: Introduction to Computer Science I  
Boston University  
Spring 2016

Syllabus

Description: The first course for computer science majors and anyone seeking a rigorous introduction. Develops computational problem-solving skills by programming in the Python language, and exposes students to variety of other topics from computer science and its applications.

Prerequisites: none

Instructors
A1/B1: David G. Sullivan, Ph.D. (dgs@cs.bu.edu, PSY 228D, 64 Cummington Mall)  
C1: Aaron Stevens (azs@cs.bu.edu, PSY 228B, 64 Cummington Mall)  
See the course website for the schedule of instructor, TF and CA office hours.

Teaching Fellows/Assistants (TFs/TAs)  
Alex Breen (abreen@bu.edu)  
Fatih Cakir (fcakir@bu.edu)  
Wenxin Feng (wenxin@bu.edu)  
Stephanie Alibrandi Ferrari (smaf91@bu.edu)

Course Assistants (CAs)  
We are fortunate to have a number of undergraduate course assistants (CAs) as members of the course staff. They will be working with you in the labs and holding office hours each week. See the course website for their names and contact information, and for the schedule of office hours.

Meeting Times and Places

lectures:    section A1: MWF, 10-11, CAS 522  
            section B1: MWF, 12-1, CAS B12  
            section C1: TuTh, 9:30-11, COM 101
plus a weekly, one-hour lab session in the CS teaching lab, EMA 304.

Important: You must also be able to take the midterm exams, which will be held on two Wednesday evenings from 6:00-7:00 p.m. (see the schedule).

Course Website:  http://www.cs.bu.edu/courses/cs111

Course Discussion Site (Piazza):  https://piazza.com/bu/spring2016/cs111
Requirements and Grading
1. Weekly problem sets and final project (40% of the final grade)
2. Exams: two midterm exams (25%) and a final exam (25%)
3. Preparation and participation (10%)

To pass the course, you must earn a passing grade for each of the three components.

Academic Conduct
Problem sets will include two types of problems:
- individual-only problems that you must complete on your own
- pair-optional problems that you may complete alone or with one other student.

General rules for collaboration:
- You may discuss the main ideas of a given problem with someone else (e.g., another student, a friend, or a tutor) provided that you acknowledge doing so in your solution, but you must complete the actual solution by yourself.
- You may not copy all or part of another person’s work, even if you subsequently modify it. This includes cases in which someone else tells you what you should write for your solution.
- You may not view all or part of someone else’s work.
- You may not show all or part of your work to someone else.
- You may not consult solutions from past semesters, or those found in books or on the Web.

Rules for working with a partner on pair-optional problems:
- You may not work with more than one partner on a given assignment. (However, you are welcome to switch partners between assignments.)
- You may not split up the work and complete it separately.
- You must work together at the same computer for every problem that you complete as a pair. While you are working, the screen should be visible to both of you. One person should type, while the other person observes, critiques, and plans what to do next. You must switch roles periodically, and your solution should be a true collaborative effort.
- You must both submit the same solution to each problem that you did as a pair, and you must clearly indicate that you worked on the problem as a pair.
- Although you will look at your partner's work for pair-optional problems that you do together, the above rules for collaboration still apply to your interactions with students other than your partner.

Violations of the collaboration policy will result in a grade of 0 and will be reported to the Academic Conduct Committee (ACC). The ACC may suspend/expel students found guilty of misconduct. We will assume that you understand BU’s Academic Conduct Code: http://www.bu.edu/academics/policies/academic-conduct-code
Other Policies

Late problem sets: Problem sets must be submitted by the date and time listed on the assignment (typically by 9 p.m.). There will be a 10% deduction for submissions up to 24 hours late. **We will not accept any homework that is more than 24 hours late.** Plan your time carefully, and don't wait until the last minute so you will have ample time to ask questions and obtain assistance from the course staff.

Pre-lecture preparation: To help you prepare for lecture, you will typically be required to complete an assigned reading and/or watch one or two short videos. You will also be required to perform some type of brief task (an online quiz or other exercise) to demonstrate that you have completed the necessary preparation. Your work on these tasks will not typically be graded for correctness, but it should demonstrate that you have adequately prepared for lecture. The pre-lecture tasks must be submitted by the specified date and time. **Late pre-lecture work will not be accepted.**

The attendance/participation portion of your grade will be based on your consistent attendance at the lectures and lab sessions, and on your participation in the activities for each class. In particular, you must participate in small-group activities during lecture in which you will discuss questions with other students and "vote" on the answers. These activities are designed to deepen your understanding of the material, and you will be graded on your participation, not on the correctness of your answers. To accommodate unavoidable absences or lateness, you will receive full credit for attendance as long as you make at least 85% of the votes over the course of the semester and attend 85% of the lab sessions. **Voting for someone else is not allowed, and will result in a lowering of the participation grade of both the students involved.**

The final exam will replace your lowest problem-set grade if doing so helps your final grade. (The final-project grade cannot be replaced.) The final exam will also replace your lowest midterm-exam grade if doing so helps your final grade.

Extensions and makeup exams will only be given in documented cases of serious illness or other emergencies.

You cannot redo or complete extra work to improve your grade.

Incompletes will not be given except in extraordinary circumstances.

Course Materials

Textbook: We will use the free online textbook *CS for All* by Christine Alvarado, Zachary Dodds, Geoff Kuenning, and Ran Libeskind-Hadas. It can be found here: [http://www.cs.hmc.edu/csforall/index.html](http://www.cs.hmc.edu/csforall/index.html)

Clicker: We will be using Turning Technologies ResponseWare for in-class activities and attendance. You will need *either:* (a) a Turning Technologies Clicker RF LCD and 4-year account (ISBN 9781934931783; available at the BU Bookstore); or (b) the ResponseWare app for your phone or tablet device and a student license. If you choose to use this app, you can find details at: [https://www.turningtechnologies.com/response-solutions/responseware](https://www.turningtechnologies.com/response-solutions/responseware)
## Schedule (tentative)

<table>
<thead>
<tr>
<th>week</th>
<th>lecture dates</th>
<th>topics, exams, assignments, and special dates</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>A/B: 1/20, 1/22 C: 1/19, 1/21</td>
<td>Course overview and introduction Computational problem-solving with Picobot No labs this week.</td>
</tr>
<tr>
<td>1</td>
<td>A/B: 1/25, 1/27, 1/29 C: 1/26, 1/28</td>
<td>Getting started in Python Data types and expressions Strings and lists A first look at functions <strong>Problem Set 0, part I due on 1/28</strong> <strong>Problem Set 0, part II due on 1/30</strong></td>
</tr>
<tr>
<td>2</td>
<td>A/B: 2/1, 2/3, 2/5 C: 2/2, 2/4</td>
<td>Functions (cont.) Making decisions (conditional execution) Local and global variables; the runtime stack Recursion 2/1: last day to add a class <strong>Problem Set 1, part I due on 2/4</strong> <strong>Problem Set 1, part II due on 2/7</strong></td>
</tr>
<tr>
<td>3</td>
<td>A/B: 2/8, 2/10, 2/12 C: 2/9, 2/11</td>
<td>Functions and recursion (cont.) Higher-order functions and list comprehensions <strong>Problem Set 2, part I due on 2/11</strong> <strong>Problem Set 2, part II due on 2/14</strong></td>
</tr>
<tr>
<td>4</td>
<td>A/B: 2/16, 2/17, 2/19 C: 2/18</td>
<td>More list comprehensions; lists of lists Algorithm design; encryption and decryption Representing information A/B: No lecture on 2/15 (Presidents' Day) A/B: Lecture on 2/16 (Mon. schedule) C: No lecture on 2/16 (Mon. schedule) No labs this week. <strong>Problem Set 3, part I due on 2/18</strong> <strong>Problem Set 3, part II due on 2/21</strong></td>
</tr>
<tr>
<td>5</td>
<td>A/B: 2/22, 2/24, 2/26 C: 2/23, 2/25</td>
<td>Digital logic and circuit design Machine organization 2/23: last day to drop without a 'W' <strong>Midterm 1 on 2/24 from 6-7 p.m.</strong> <strong>Problem Set 4 (all) due on 2/28</strong></td>
</tr>
<tr>
<td>6</td>
<td>A/B: 2/29, 3/2, 3/4 C: 3/1, 3/3</td>
<td>Assembly language <strong>Problem Set 5 (all) due on 3/3</strong> Spring break</td>
</tr>
<tr>
<td>7</td>
<td>A/B: 3/14, 3/16, 3/18 C: 3/15, 3/17</td>
<td>Assembly language (cont.) Loops and imperative programming Cumulative computations <strong>Problem Set 6, part I due on 3/17</strong> <strong>Problem Set 6, part II due on 3/20</strong></td>
</tr>
<tr>
<td>Week</td>
<td>Dates</td>
<td>Topics</td>
</tr>
<tr>
<td>------</td>
<td>----------------</td>
<td>-----------------------------------------------------------------------</td>
</tr>
</tbody>
</table>
| 8    | A/B: 3/21, 3/23, 3/25  
   C: 3/22, 3/24 | Nested loops  
   Design using loops  
   Object-oriented programming: using existing objects  
   References; mutable vs. immutable data  
   **Problem Set 7, part I due on 3/24**  
   **Problem Set 7, part II due on 3/27** |                                                                 |
| 9    | A/B: 3/28, 3/30, 4/1  
   C: 3/29, 3/31 | 2-D lists  
   Object-oriented programming (cont.)  
   File processing  
   Classes: creating your own types of objects  
   **Problem Set 8, part I due on 3/31**  
   **Problem Set 8, part II due on 4/3**  
   4/1: *last day to drop a class with a 'W'* |                                                                 |
| 10   | A/B: 4/4, 4/6, 4/8  
   C: 4/5, 4/7 | Classes: creating your own types of objects (cont.)  
   Dictionaries  
   **Midterm 2 on 4/6 from 6-7 p.m.**  
   **Problem Set 9 (all) due on 4/10** |                                                                 |
| 11   | A/B: 4/11, 4/13, 4/15  
   C: 4/12, 4/14 | Inheritance  
   Games and AI  
   Overview of the final project; Finite-state machines  
   **Problem Set 10, part I due on 4/14**  
   **Problem Set 10, part II due on 4/17** |                                                                 |
| 12   | A/B: 4/20, 4/22  
   C: 4/19, 4/21 | Finite-state machines (cont.)  
   Turing machines  
   **A/B: No lecture on 4/18 (Patriots' Day)**  
   Labs will be held; Wed. is a Monday schedule  
   **Problem Set 11, part I due on 4/21**  
   **Problem Set 11, part II due on 4/24** |                                                                 |
| 13   | A/B: 4/25, 4/27, 4/29  
   C: 4/26, 4/28 | Algorithm efficiency and problem "hardness"  
   Uncomputability  
   Wrap-up  
   **Final project due on 4/28** |                                                                 |

Final exam (all sections): Fri., 5/6, 9-11 am