Syllabus

Overview
This course is a fast-paced introduction to computer science designed for students who plan to work extensively with computers (for example, engineers, biologists, physicists, and economists), as well as future concentrators who plan to take more advanced courses in the field.

The first half of the course covers foundational programming concepts such as data types, conditional execution, iteration, and recursion. It also explores the key features of object-oriented programming, and the manipulation of data stored in files and arrays.

The second half of the course provides a survey of fundamental data structures for information processing, including lists, stacks, queues, trees, and graphs. It explores the implementation of these data structures using both array-based and linked representations, and it examines classic algorithms that use these structures for tasks such as sorting, searching, and text compression. Techniques for analyzing the efficiency of algorithms are also covered. The course provides complete coverage of the syllabus for the advanced placement examination in computer science.

Prerequisites
Familiarity with precalculus. No prior programming experience is required. Students who have completed the Harvard Extension School courses CSCI E-10a, CSCI E-10b, CSCI E-22, or CSCI E-50 (CSCI-50a and CSCI E-50b) cannot earn degree credit for CSCI S-111.

Instructor
David G. Sullivan, Ph.D. (dgs@cs.bu.edu)
Senior Lecturer on Computer Science, Boston University
office hours: after lecture, and by appointment

Teaching Assistants (see the course website for office hours)
Cody Doucette (doucette@bu.edu)
Caitlin Fournier (crf18@bu.edu)
Justin Ingwersen (justini@bu.edu)
Eli Saracino (esaracin@bu.edu)
Meeting Times and Places

**Lectures:** Mon-Fri, 8:30-11:30 a.m., Sever 113

There is one day each week without lecture – usually Wednesday, but there are exceptions. See the schedule below for more detail.

**Sections:** daily one-hour meetings in the early afternoon on days when lecture is held; times and locations TBA.

Attendance at both the lectures and sections is essential, as this course moves very fast. We also encourage you to meet regularly with a member of the teaching staff to review any problems that you are having with the homework or with specific topics.

Requirements

The course is divided into ten distinct units. Units 1-5 cover programming fundamentals, and units 6-10 cover data structures and algorithms.

1. **Problem sets:** Each unit has a problem set consisting of two parts. Part I consists of short "written" problems that test your understanding of the key concepts from the unit. Your answers to Part I must be submitted as a single plain-text file. Part II consists of one or more programming problems that require you to employ the concepts from the unit. All programming problems **must be completed in Java**, and they must compile and run in order to be eligible for full credit.

2. **Unit tests:** At the conclusion of most units, students will take a 50-minute test on the material in that unit. Each unit test is worth 25 points. If you score less than 70% on a unit test (i.e., a 17.5 or lower), you may take a retest for a maximum score of 18.

3. **Final exam:** a three-hour comprehensive exam at the end of the course.

**Important note:** The problem sets tend to be extremely time-consuming. Don’t wait until the last minute to begin them! You should plan on devoting approximately 20-30 hours of work per week. **If you have other major time commitments, you should reconsider whether to take this course.**

**Graduate-credit students:** Students taking the course for graduate credit must complete additional homework. On most problem sets, the problems required of all students will be worth a total of 100 points; grad-credit students will complete one or two additional problems worth a total of 10 points. These grad-credit problems are typically more challenging than the other problems, and thus grad-credit students should plan to spend approximately 20% more time on the homework.

Grading Policies

Late penalties: Homework is due **by 10 p.m.** on the date listed on the assignment. There will be a 10% deduction for homework that is 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 to begin an assignment. Starting early will give you ample time to ask questions and obtain assistance.

Determining the final grade: homework 50%, unit tests 25%, final exam 25%
The final exam will replace your lowest assignment grade if doing so helps your final
grade. The final exam will also replace your lowest unit-test grade if doing so helps
your final grade.

The final grades are not curved. The performance of the class as a whole is taken into
account when assigning letter grades, but this can only improve your grade, not harm
it.

Extensions and makeups will only be given in documented cases of serious illness or
other emergencies. You cannot redo or complete extra work to improve your grade.

An EXT (extension) grade will be granted only in extreme circumstances (e.g.,
ilness), and only when appropriate documentation has been provided. Please bring
any such circumstances to Dr. Sullivan's attention as soon as possible.

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

Rules for individual-only problems:
- You may discuss the main ideas of a given problem with other students
  (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.
- You may not view all or part of another student's work.
- You may not show all or part of your work to another student.

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.

For both types of problems, you may not consult solutions from past semesters, or
those found in books or on the Web.

If we believe that a student is guilty of academic misconduct, we will refer the matter
to the Administrative Board of the Summer School, who could require withdrawal
from the course and suspension from all future work at the School.

The Summer School provides resources to support academic integrity here:
http://www.summer.harvard.edu/resources-policies/resources-support-academic-integrity
**Summer School Policies**

We also expect you to know and adhere to the general policies and procedures of the Summer School: [http://www.summer.harvard.edu/policies/student-responsibilities](http://www.summer.harvard.edu/policies/student-responsibilities)

**Accessibility Services**

The Summer School is committed to providing an accessible academic community. The Accessibility Services Office offers a variety of accommodations and services to students with documented accessibility issues. This site has more information: [https://www.summer.harvard.edu/resources-policies/accessibility-services](https://www.summer.harvard.edu/resources-policies/accessibility-services)

**Textbooks**

- CSCI S-111 coursepack. This contains all of the lecture notes for the course. It will be available from Gnomon Copy (1308 Mass Ave., across from the Yard), or you can access electronic copies of the notes on the course website.

- **Optional**: *Building Java Programs, 4th ed.* by Stuart Reges and Marty Stepp (Pearson, 2016). Older versions are also fine. This book is *not* required.


**Course Outline**

**Unit 1: Getting started.** Programming in Scratch (a graphical language developed at MIT that will allow us to quickly introduce a number of key programming concepts). Simple Java programs. Statements. Standard output. Procedural decomposition using simple methods.

**Unit 2: Imperative programming, part I.** The programming process. Data types. Literals, variables, and expressions. Definite loops. Simple conditional execution.


**Unit 5: Object-oriented programming.** Writing "blueprint" classes. Fields, non-static methods, and constructors. Inheritance and polymorphism.

**Unit 6: Foundations of data structures.** Defining and implementing an abstract data type. Memory allocation (stack and heap storage). Recursion revisited, including recursive backtracking algorithms.

**Unit 7: Sorting and algorithm analysis.** Sorting arrays using the following algorithms: insertion sort, selection sort, bubble sort, Shellsort, quicksort, and radix sort. Algorithm analysis: running-time analysis; big-O notation; worst-case, average-case, and best-case analyses.
Unit 8: Sequences. Linked lists. List, stack, and queue abstract data types, including both array and linked-list implementations of each of these ADTs. Implementing a generic collection.


Schedule (tentative)  
key: BJP = Building Java Programs; L = Lafore book

<table>
<thead>
<tr>
<th>lecture date</th>
<th>unit(s) covered</th>
<th>due dates</th>
<th>unit tests/retests</th>
<th>optional readings</th>
</tr>
</thead>
<tbody>
<tr>
<td>Mon June 19</td>
<td>Unit 1</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Tues June 20</td>
<td>Unit 1</td>
<td></td>
<td></td>
<td>BJP 1</td>
</tr>
<tr>
<td>Wed June 21</td>
<td>No lecture</td>
<td>PS 1, part I due</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Thurs June 22</td>
<td>Unit 2</td>
<td>PS 1, part II due</td>
<td>BJP 2, 4.1</td>
<td></td>
</tr>
<tr>
<td>Fri June 23</td>
<td>Unit 2</td>
<td></td>
<td>Unit 1 test</td>
<td></td>
</tr>
<tr>
<td>Mon June 26</td>
<td>Unit 3</td>
<td>PS 2, part I due</td>
<td></td>
<td>BJP 3</td>
</tr>
<tr>
<td>Tues June 27</td>
<td>Unit 3</td>
<td></td>
<td>Unit 1 retest</td>
<td></td>
</tr>
<tr>
<td>Wed June 28</td>
<td>No lecture</td>
<td>PS 2, part II due</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Thurs June 29</td>
<td>Unit 3</td>
<td></td>
<td></td>
<td>BJP 4</td>
</tr>
<tr>
<td>Fri June 30</td>
<td>Unit 3</td>
<td></td>
<td>Unit 2 test</td>
<td>BJP 5.1-4</td>
</tr>
<tr>
<td>Mon July 3</td>
<td>Unit 4</td>
<td>PS 3, part I due</td>
<td>Unit 2 retest</td>
<td>BJP 7</td>
</tr>
<tr>
<td>Tues July 4</td>
<td>No lecture</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Wed July 5</td>
<td>Unit 4</td>
<td>PS 3, part II due</td>
<td></td>
<td>BJP 6</td>
</tr>
<tr>
<td>Thurs July 6</td>
<td>Unit 4</td>
<td></td>
<td></td>
<td>BJP 12</td>
</tr>
<tr>
<td>Fri July 7</td>
<td>Unit 5</td>
<td>PS 4, part I due</td>
<td>Unit 3 test</td>
<td>BJP 8</td>
</tr>
<tr>
<td>Mon July 10</td>
<td>Unit 5</td>
<td>PS 4, part II due</td>
<td></td>
<td>BJP 9</td>
</tr>
<tr>
<td>Tues July 11</td>
<td>Unit 5</td>
<td></td>
<td>Unit 3 retest</td>
<td></td>
</tr>
<tr>
<td>Wed July 12</td>
<td>Unit 6</td>
<td>PS 5, part I due</td>
<td></td>
<td>L 1</td>
</tr>
<tr>
<td>Thurs July 13</td>
<td>Unit 6</td>
<td></td>
<td>Unit 4 test</td>
<td>L 6</td>
</tr>
<tr>
<td>Fri July 14</td>
<td>No lecture</td>
<td>PS 5, part II due</td>
<td>Unit 4 retest</td>
<td></td>
</tr>
<tr>
<td>Mon July 17</td>
<td>Unit 7</td>
<td>PS 6, part I due</td>
<td></td>
<td>L 2, 3, 7 (B 13)</td>
</tr>
<tr>
<td>Tues July 18</td>
<td>Unit 7</td>
<td></td>
<td>Unit 5 test</td>
<td></td>
</tr>
<tr>
<td>Wed July 19</td>
<td>No lecture</td>
<td>PS 6, part II due</td>
<td></td>
<td>L 4, 5 (B 16, 15)</td>
</tr>
<tr>
<td>Thurs July 20</td>
<td>Unit 8</td>
<td></td>
<td>Unit 5 retest</td>
<td></td>
</tr>
<tr>
<td>Fri July 21</td>
<td>Unit 8</td>
<td>PS 7, part I due</td>
<td>Unit 6 test</td>
<td>(B 14)</td>
</tr>
<tr>
<td>Mon</td>
<td>July 24</td>
<td>finish unit 8 start Unit 9</td>
<td>PS 7, part II due</td>
<td>Unit 6 retest</td>
</tr>
<tr>
<td>------------</td>
<td>---------</td>
<td>-----------------------------</td>
<td>-------------------</td>
<td>---------------</td>
</tr>
<tr>
<td>Tues</td>
<td>July 25</td>
<td>Unit 9</td>
<td>PS 8, part I due</td>
<td>Unit 7 test</td>
</tr>
<tr>
<td>Wed</td>
<td>July 26</td>
<td>No lecture</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Thurs</td>
<td>July 27</td>
<td>Unit 9</td>
<td>PS 8, part II due</td>
<td>Unit 7 retest</td>
</tr>
<tr>
<td>Fri</td>
<td>July 28</td>
<td>finish Unit 9 start Unit 10</td>
<td>PS 9, part I due</td>
<td>Unit 8 test</td>
</tr>
<tr>
<td>Mon</td>
<td>July 31</td>
<td>Unit 10</td>
<td>PS 9, part II due</td>
<td>Unit 8 retest</td>
</tr>
<tr>
<td>Tues</td>
<td>August 1</td>
<td>Unit 10</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Wed</td>
<td>August 2</td>
<td>Wrap-up/review</td>
<td>PS 10 due</td>
<td>Unit 9 test</td>
</tr>
<tr>
<td>Thurs</td>
<td>August 3</td>
<td>No lecture</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Fri</td>
<td>August 4</td>
<td>Final exam, 8:30-11:30 a.m.</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

*Other important dates:*
- June 21: late registration ends; course drop deadline for full-tuition refund
- June 28: course drop deadline for half-tuition refund
- July 21: last day to withdraw for a grade of WD (no refund)