Description: The second course for computer science majors and anyone seeking a rigorous introduction. Covers advanced programming techniques and data structures using the Java language. Topics include searching and sorting, recursion, algorithm analysis, linked lists, stacks, queues, trees, and hash tables. Carries MCS divisional credit in CAS. Fulfills a single unit in the following BU Hub areas: Quantitative Reasoning II, Creativity/Innovation, Critical Thinking.

Prerequisites: CAS CS 111, or the equivalent. If you have not had significant prior experience with recursion, you are strongly encouraged to take CS 111 first.

Instructor
Christine Papadakis-Kanaris (cpk@cs.bu.edu)
See the course website for the schedule of instructor and TF office hours.

Teaching Assistants (TFs)
Nathan Cordner (ncordner@bu.edu)
Qiancheng Fu (qcfu@bu.edu)

Meeting Times and Places
Lectures: section SA1: M-Th, 10:00 am-12:00 pm, CAS B12 or remotely (see below)
          section SA2: M-Th, 1:00-3:00 pm, CAS 224 or remotely (see below)
labs: see your schedule for the time; fully remote on Zoom (see below)

In keeping with the University's Learn from Anywhere model:
- The lectures will be offered using a hybrid mode of instruction. Beginning with the first lecture, there will be in-person class meetings at the times listed in your schedule, but classes will also be live-streamed via Zoom for students who cannot attend in person.
- Links to the Zoom meetings for lecture will be provided on the course's Blackboard Learn site before the first lecture.
- In-person lecture attendance will be managed using the InClassLfA app. Students who wish to attend class in person should use this app to indicate their preference for doing so.
- We encourage you to participate in the lectures either in person or on Zoom. However, if you cannot participate in a given lecture, there will be a recording that you can watch asynchronously. To avoid falling behind, you should watch the recording on the day of the lecture.
- The weekly lab sessions will be fully remote. Students are expected to attend their lab session on Zoom at the time at which it is held.
- Students who attend class in person or ask questions via audio on Zoom should be aware that they may be recorded. However, these recordings will
COVID-19-Related Class Expectations
To promote a safe learning environment, those who attend lecture in person must:

- comply with University-mandated COVID-19 testing and health attestation requirements
- wear a face covering at all times during class and when in other public spaces
- maintain physical distancing of 6 feet from the nearest person at all times, including when entering and leaving the classroom

The instructors will follow the above guidelines, and students who choose to attend class in person must also adhere to them. Students who do not wish to follow these guidelines should take the class remotely.

Exams
Information to be provided on our Course Website:
http://www.cs.bu.edu/courses/cs112

Requirements and Grading
1. Weekly problem sets (25% of the final grade)
2. Exams: two midterm exams (30%) and a final exam (40%)
3. Participation (5%; see below)

To pass the course, you must earn a passing grade on each of the first two components.

Collaboration Policy
You are strongly encouraged to collaborate with one another in studying the lecture materials and preparing for quizzes and exams.

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 a partner.

For both types of problems, you may discuss ideas and approaches with others (provided that you acknowledge this in your solution), but such discussions should be kept at a high level, and should not involve actual details of the code or of other types of answers. You must complete the actual solutions on your own (or, in the case of a pair-optional problem, with your partner if you choose to use one).

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 (e.g., via a Zoom meeting) for every problem that
you complete as a pair, and your solution must be a 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 by putting your partner's name at the top of the file.

**Academic Misconduct**

We will assume that you understand BU’s Academic Conduct Code: [http://www.bu.edu/academics/policies/academic-conduct-code](http://www.bu.edu/academics/policies/academic-conduct-code)

You should also carefully review the CS department’s page on academic integrity: [http://www.bu.edu/cs/undergraduate/undergraduate-life/academic-integrity](http://www.bu.edu/cs/undergraduate/undergraduate-life/academic-integrity)

Prohibited behaviors include:

- copying all or part of someone else's work, even if you subsequently modify it; this includes cases in which someone tells you what you should write for your solution
- viewing all or part of someone else's work (with the exception of work that you and your partner do together on a pair-optional problem)
- showing all or part of your work to another student (with the exception of work that you and your partner do together on a pair-optional problem)
- consulting solutions from past semesters, or those found online or in books
- posting your work where others can view it (e.g., online)
- receiving assistance from others or collaborating with others during an exam, or consulting materials except those that are explicitly allowed.

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).**

**Other Policies**

*Laptops:* Students taking CS courses are expected to have a laptop capable of running a currently supported version of Microsoft Windows, Mac OS X, or Linux. See this page for more info: [https://www.bu.edu/cs/undergraduate/undergraduate-life/laptops](https://www.bu.edu/cs/undergraduate/undergraduate-life/laptops)

*Late problem sets:* Problem sets must be submitted by the date and time listed on the assignment (typically by 11:59 p.m.). There will be a 20% 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. This preparation is not graded, but failing to complete it will make it more difficult for you to understand the material presented in lecture.

The *participation* portion of your grade will be based on your completion of online questions connected to the lectures, and on your consistent participation in the lab sessions on Zoom.
The final grades are not curved. The performance of the class as a whole is taken into account in assigning letter grades, but this can only improve your grade, not harm it. 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**
- You are not required to buy a textbook. Instead, we will provide detailed lecture slides and assign readings from freely available online resources.
- **Required:** We will be using the Top Hat Pro platform. More detail will be provided in class.

**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>1</td>
<td>5/24</td>
<td>Course overview and introduction</td>
</tr>
<tr>
<td></td>
<td>5/25</td>
<td>Java Basics; I/O; Control structures I (Java conditionals); Methods</td>
</tr>
<tr>
<td></td>
<td>5/26</td>
<td>Java Loops, Scope of Variables</td>
</tr>
<tr>
<td></td>
<td>5/27</td>
<td>Java Memory Model</td>
</tr>
<tr>
<td>1</td>
<td>5/31*</td>
<td>Memorial Day – no class</td>
</tr>
<tr>
<td></td>
<td>6/1</td>
<td>Java Arrays</td>
</tr>
<tr>
<td></td>
<td>6/2</td>
<td>Classes and Objects, an introduction</td>
</tr>
<tr>
<td></td>
<td>6/3</td>
<td>Classes and Objects, a deeper look</td>
</tr>
<tr>
<td>2</td>
<td>6/7</td>
<td>ArrayBag case study; Introduction to recursion</td>
</tr>
<tr>
<td></td>
<td>6/8</td>
<td>Recursion, and recursive backtracking (N-Queens problem)</td>
</tr>
<tr>
<td></td>
<td>6/9</td>
<td>Algorithm Analysis and Big-O notation</td>
</tr>
<tr>
<td></td>
<td>6/10</td>
<td>Basic Sorting (selection, bubble, insertion)</td>
</tr>
<tr>
<td>3</td>
<td>6/14</td>
<td>Divide and Conquer sorting algorithms</td>
</tr>
<tr>
<td></td>
<td>6/15</td>
<td>Linked List, StringNode case study</td>
</tr>
<tr>
<td></td>
<td>6/16</td>
<td>Linked List traversals</td>
</tr>
<tr>
<td></td>
<td>6/17</td>
<td>List, Stack, Queue ADT</td>
</tr>
<tr>
<td>4</td>
<td>6/21*</td>
<td>Midterm exam</td>
</tr>
<tr>
<td></td>
<td>6/22</td>
<td>Binary Trees, and Binary Search Trees</td>
</tr>
<tr>
<td></td>
<td>6/23</td>
<td>Balanced Trees, 2-3 Trees</td>
</tr>
<tr>
<td></td>
<td>6/24</td>
<td>Hash Tables</td>
</tr>
<tr>
<td>5</td>
<td>6/28</td>
<td>The Heap, Priority Queues</td>
</tr>
<tr>
<td></td>
<td>6/29</td>
<td>Heapsort;</td>
</tr>
<tr>
<td></td>
<td>6/30</td>
<td>Course review</td>
</tr>
<tr>
<td></td>
<td>7/1</td>
<td>Final Exam</td>
</tr>
</tbody>
</table>