CAS CS 112: Introduction to Computer Science II

Boston University, Spring 2026

Syllabus

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 each of 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.

Instructors:
A1/A2: Preethi Narayanan (pnarayan @bu.edu, CDS 909)
A1/A2: Christine Papadakis-Kanaris (cpk@bu.edu, CDS 943)

Undergraduate Teaching Assistant (TF)

Arijit Ray	array@bu.edu

Undergraduate Teaching Assistant (TA)

Shiven (Shiv) Sharma	shivens@bu.edu
Tianyou (Michael) Lou	tianyoul@bu.edu
Liam McDonald	liammcd@bu.edu
Rachel Pastreich	rachpast@bu.edu

Course Assistants (CAs)
We are fortunate to have several 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 info.

Lectures and Labs
Lecture section(s): A1: TuTh, 2:00 - 3:15 pm, CGS 511; A2: TuTh, 3:30-4:45 pm, CGS 511
lab**: a weekly, one-hour session; see your schedule for the time and location**

Important: You must also be able to take the midterm exam, and graded quizzes which will be held on three Wednesday evenings from 6:30-8:00 p.m. as per schedule below. We are not scheduled to meet on any other Wednesday evening but may use this time for supplemental tutorials.

Course Website: http://www.cs.bu.edu/courses/cs112
In addition, announcements and course materials will be posted Blackboard.

Requirements and Grading

1. Weekly problem sets (20% of the final grade)
2. Graded Quizzes (two, 10% each)
3. Exams: one midterm exam (20%) and a (*cumulative*) final exam (35%)
4. In lecture questions and participation (5%)

To pass the course, you must earn a passing grade on each of components 1 and 2.

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:

- *individual-only* problems that you must complete on your own, and
- *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** (at the same computer or via a Zoom meeting) for all problems completed as a pair, and your work must be a collaborative effort.
- You and your partner 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 and have carefully review the CS department's page on academic integrity:
http://www.bu.edu/academics/policies/academic-conduct-code
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 to write for your solution
- with the exception of work that you and your partner do together on a pair-optional problem:
  - viewing all or part of someone else's work
  - showing all or part of your work to another student
- consulting solutions from past semesters, or those found online or in books
- using tools that automate (e.g. ChatGPT) and/or assist with the writing of code or the completion of other types of solutions
- posting your work online or where others can view it - even after you complete the course

- receiving assistance from others or collaborating with others during an exam, or consulting materials except those that are explicitly allowed. Using ChatGPT or any other on-line source for solutions.

Any submission suspected of violating course policy will receive an automatic score of 0 on both parts of the problem set. Students will have the opportunity to speak to the instructors and explain. Students who do not communicate with the instructors will receive an additional penalty of 10% on the next exam or graded quiz.

Repeated 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

Late problem sets: Problem sets will be divided into two parts, *Part I* and *Part II*. Both parts of the problem set will have the same stated due date, but separate submission portals through Gradescope. **We will not accept any submissions for Part I beyond the stated deadline.** However, the submission portal for Part II submissions will remain open for a 24-hour late-penalty (10%) period after the posted deadline.

Pre-lecture preparation: To help you prepare for lecture, you are encouraged to complete an assigned reading. 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 *attendance/participation* portion of your grade will be based on your consistent attendance at the lectures and lab sessions. Attendance will be taken at every lab, and you must attend at least 85% of the lab sessions to get full credit for lab participation. We will be using TopHat as part of our in-lecture participation, and we will have several in-class quizzes which will be graded on completion only, but will be factored into your participation grade.

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

Textbook: You are not required to purchase a textbook. Instead, we will be assigning readings from freely available online resources. If you are interested in purchasing a Java reference book, we will recommend some possible titles in lecture.

In-class software: We will be using the Top Hat platform for in-class activities and attendance. You will need to purchase a subscription to access Top Hat from your web browser or via a mobile app. You must sign in using your BU credentials at https://app.tophat.com/e/503546/.

Schedule (tentative and subject to change)

Week	lecture dates	topics, exams, assignments, and special dates
0	1/20 1/22 *1/23*	Course overview and introduction Java basics *Lab: Java Basics*
1	1/27 1/29 *1/30*	Conditional execution and user input Static methods; loops; variable scope *Lab: More with Java*
2	***2/3*** 2/3 2/5 *2/6*	***Problem Set 1 due*** Primitives, objects, and references Arrays and references *Lab: Understanding references* ***2/2: last day to add a class***
3	***2/10*** 2/10 2/12 *2/13*	***Problem Set 2 due*** Object-oriented programming Inheritance and polymorphism *Lab: Working with Custom classes*
4	**2/17** ***2/17*** ***2/18*** 2/19 *2/20*	***Problem Set 3 due*** ***Classes run on a Monday schedule*** ***Quiz #1*** Inheritance and polymorphism *Lab: Understanding Inheritance*
5	***2/24*** 2/24 *2/26* *2/27*	***Problem Set 4 due*** Introduction to Recursion Recursion and Recursive backtracking *Lab: Recursive programming* *2/24: last day to drop without a 'W'*
6	3/3 3/5 *3/6*	Introduction to Big-O and algorithm analysis Sorting and algorithm analysis *Lab: Understanding Big-O*

7	**3/7-3/15**	***Spring Break***
8	3/17 3/19 *3/20*	Sorting and algorithm analysis (cont.) Introduction to Linked Lists *Lab: Sorting algorithms*
9	***3/24*** 3/24 ***3/25*** 3/26 *3/27*	***Problem Set 5 due*** Linked lists ***Midterm*** Linked list processing *Lab: Linked Lists*
10	3/31 4/2 *4/3*	Abstract data types (ADTs), interfaces and List ADT Java Iterators; Java generic types *Lab: ADTs and Iterators* *4/3: last day to drop a class with a 'W'*
11	***4/7*** 4/7 4/9 4/10	***Problem Set 6 due*** Stack and Queue ADTs Trees and Binary trees *Lab: Working with Stacks and Queues*
12	***4/14*** 4/14 ***4/15*** 4/16 4/17	***Problem Set 7 due*** Binary Search trees and Balanced Trees ***Quiz #2*** Introduction to Hash Tables *Lab: Hash Tables*
13	***4/21*** 4/21 4/23 *4/24*	***Problem Set 8 due*** Hash tables; Introduction to Complete Trees Complete and Heap Trees *Lab: The Heap*
14	***4/28*** 4/28 **4/30**	***Problem Set 9 due*** Applications of Heap Trees (priority queues, heapsort) **Semester review!** **Final exam: time and date TBD** **Please wait until your instructor informs you of the date. The initial date posted by the Registrar may <u>not</u> be correct.** ***Make sure that you are available for the entire exam period – up to and including May 8th!***