

CAS CS 112: Introduction to Computer Science II

Boston University, Fall 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.

Instructor:

A1/A2: Preethi Narayanan (pnarayan@bu.edu, CDS 909)

Undergraduate Teaching Assistant (TA)

Koji Aquilino	kojiken@bu.edu
Ricky Yan	rickyy@bu.edu
Matthew Hepburn	mhepburn@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, 5:00-6:15 pm, CAS B18; A2: TuTh, 3:30 - 4:45 pm, EPC 207

Labs: **a weekly, one-hour session; see your schedule for the time and location**

Important: You must also be able to take the quizzes and midterm exams, which will be held on three Wednesday evenings from 6:30-8:00 p.m. 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. Two Quizzes (20% total, 10% each)
3. Exams: one midterm exam (20%) and a (*cumulative*) final exam (30%)
4. In lecture questions and participation (10%)

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

Grading Policy

Letter grades will be assigned at the end of the semester using the following boundaries:

- A : [93, 100]
- A-: [90, 93)
- B+: [87, 90)
- B: [83, 87)
- B-: [80, 83)
- C+: [77, 80)
- C: [70, 77)
- D: [60, 70)
- F: [0, 60)

Adjustments to these boundaries may be made at the end of the semester. If adjustments are made, they will always be in your benefit. ***Adjustments are not guaranteed.***

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. 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:
 - o viewing all or part of someone else's work
 - o 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.

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.

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](https://app.tophat.com/e/821530/) 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/821530/>.

Schedule (tentative and subject to change)

Week	lecture dates	topics, exams, assignments, and special dates
0	9/3 9/4	Course Introduction and Java Basics <i>Lab: Java Basics</i>
1	9/8 9/10 9/11	Conditional execution and user input Static methods; loops; variable scope <i>Lab: More with Java</i>
2	9/15 9/15 9/17 9/18	<i>Problem Set 1 due</i> Primitives, objects, and references Arrays and references <i>Lab: Understanding references</i> <i>9/16: last day to add a class</i>
3	9/22 9/22 9/24 9/25	<i>Problem Set 2 due</i> Object-oriented programming Inheritance and polymorphism <i>Lab: Working with Custom classes</i>
4	9/29 9/29 9/30 10/1 10/2	<i>Problem Set 3 due</i> Inheritance and polymorphism (cont.) <i>Quiz #1</i> Introduction to Recursion <i>Lab: Understanding Inheritance</i>
5	10/6 10/6 10/8 10/9	<i>Problem Set 4 due</i> Recursion and Recursive backtracking Introduction to Big-O and algorithm analysis <i>Lab: Recursive programming</i> <i>10/7: last day to drop without a 'W'</i>
6	10/13 10/15 10/16	<i>Indigenous Peoples Day: Monday Schedule</i> Sorting and algorithm analysis <i>Lab: Understanding Big-O</i>
7	10/20 10/20 10/22 10/23	<i>Problem Set 5 due</i> Sorting and algorithm analysis (cont.) Sorting and algorithm analysis (cont.) <i>Lab: Sorting algorithms</i>
8	10/27 10/27 10/29	<i>Problem Set 6 due</i> Introduction to Linked Lists Linked lists

	10/30	<i>Lab: Linked Lists</i>
9	11/3 11/4 11/5 11/6	Abstract data types (ADTs), interfaces and List ADT Midterm Java Iterators; Java generic types <i>Lab: ADTs and Iterators</i> 11/6: last day to drop a class with a 'W'
10	11/10 11/10 11/12 11/13	Problem Set 7 due The Stack ADT The Queue ADT <i>Lab: Working with Stacks and Queues</i>
11	11/17 11/19 11/20	Trees and Binary trees Binary Search trees <i>Lab: Working with Trees</i>
12	11/24 11/24 11/25 - 11/30	Problem Set 8 due Balanced Trees Thanksgiving Break
13	12/1 12/2 12/3 12/4	Hash Tables Quiz #2 Complete and Heap Trees <i>Lab: Hash Tables</i>
14	12/8 12/8 12/10	Problem Set 9 due Applications of Heap Trees (priority queues, heapsort) Wrap up and 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. <i>Make sure that you are available for the entire exam period – up to and including December 18th!</i>