Computer Science Department
cs.salemstate.edu

SYLLABUS **Spring, 2026**

CSC 260 Data Structures and Algorithms **4 cr.**
Prerequisites: CSC105 and CSC115

Instructor:	Beifang Yi	**Office**:	MH 208B	**Phone**:	(978) 542-7246
email:	byi@salemstate.edu	**Hours**:	WF (1:30-3:45)	**Website**:	http://weblab.salemstate.edu/~byi/

Section	Time	Room	Final Exam
01	W & F 10:50am-12:05pm	MH 210	**May 4, Monday, 11:00am-1:00pm** **MH 210**
L21	W & F 12:15pm-1:30 pm	MH 210	
Office Hours		1:30—3:45pm (Wednesday, Friday), MH208B	
Communication Policy		I will reply to student emails within 24 hours.	

Catalog description

Basic data structures such as stacks, queues, linked lists, and trees are studied and applied to problems in data storage and manipulation. Applications include basic searching and sorting algorithms. Fundamental strategies for algorithm design are reviewed and extended. Design, analysis, implementation, and quality assurance techniques are discussed. Three lecture hours and three hours of scheduled laboratory per week, plus extensive programming work outside of class.
Prerequisites: CSC105 and CSC115.

Goals

The purpose of this course is to develop students' knowledge and appreciation of organization and retrieval techniques and to familiarize students with the basic concepts of order-of-magnitude analysis. The goals of this course are:

CG01: to develop an appreciation for the process of data abstraction and its usefulness in software development;
CG02: to develop the skills and knowledge necessary to perform design and basic analysis of algorithms;
CG03: to present a selection of the most common data structures and their standard implementations and uses;
CG04: to present a selection of the most common algorithms for searching and sorting.

Objectives

Upon successful completion of the course, student will have:

CO01: applied data abstraction techniques;
CO02: implemented several classic data structures "from scratch";
CO03: demonstrated knowledge and use of ADTs available in one or more language libraries;
CO04: recognized the factors required to perform algorithm design, analysis of algorithms and performed order-of-magnitude analysis;
CO05: chosen, with justification, an appropriate structure to match the requirements of a given problem, implemented the structure if necessary, and used it in an appropriate way to solve the problem;
CO06: utilized standard techniques for program validation;
CO07: demonstrated the ability to use the UML modeling language;
CO08: produced documentation for at least one major completed project, including formal class diagrams and rigorous test set specification and results;
CO09: participated in at least one group project involving problem analysis and design specification and selection;
CO10: demonstrated recognition of the need for future professional development through research into future trends in the areas of analysis of algorithms and application development and profiling.

Course Topics

A detailed topics list and a general course bibliography can be found on the Computer Science Department website at Salem State Computer Science Department (click on “Degrees & Courses”, choose “Computer Science Major”, Click on “Course Descriptions”, and then select “CSC 260 Data Structures and Algorithms” to access a PDF document.).

Text

(Required) Data Structures and Algorithms in Java, 6nd Edition, by Michael Goodrich, Roberto Tamassia, and Michael Goldwasser, Wiley, 2014 (ISBN: 978-1-118-77133-4).

Course Materials & Software

Thumb (flash) drive, 64 GB minimum or online storage (for saving your projects and coursework) in addition to your personal computer/laptop (Windows, MacBook, or Linux machines).

Download the JDK (i.e., Java Development Kit) at https://www.oracle.com/java/technologies/downloads/. Get JDK 21 or above and install it on your computer. You may use your preferred IDE for working on Java programming assignments, but the IDE is ***not*** required.

You will need to use Microsoft Word or similar software packages to complete some assignments.

It is expected that you work on your laptop for the lab/coding exercises during the lab hours. If you need technical help regarding your computer configuration or setup issues including software installation, please contact Information Technology Service (ITS) (at https://www.salemstate.edu/offices-and-services/information-technology-services).

Additional references

- Course website: http://weblab.salemstate.edu/~byi/CSC260/index.html .
    - This website is password-protected and ask the instructor of the password (or log into Canvas for it).
- Course online system (Canvas): https://elearning.salemstate.edu/.
    - Access to this site via the username and password given/assigned by SSU.
- *Java How to Program: Early Objects, 11th Edition, by Deitel & Deitel. Prentice-Hall, 2017*
- (And other references are listed in the web pages from the above linked websites).

Class/Lab Attendance

Regular attendance and active engagement in both class and lab sessions are expected and are essential to successful course completion. Lab exercises are evaluated during scheduled lab periods, and quizzes, tests, and other assessments may be administered during class or lab hours. Attendance and engagement are also assessed through **In-Class Learning Activities** (see below).

Class and lab sessions are used for instruction, hands-on exercises, discussion of course topics and Java implementation details beyond the textbook, project support, and assessment activities. Sessions begin promptly at the scheduled time, and students are expected to arrive on time and remain engaged throughout.

All course materials, assignments, grades, and announcements are delivered through **Canvas** (https://elearning.salemstate.edu/). Students are responsible for accessing Canvas using their SSU credentials, monitoring course communications, and ensuring their SSU email address is current.

Students are responsible for completing all course requirements and keeping up with course content and announcements, regardless of attendance at class or lab sessions.

Student-Instructor Communication

For the most effective assistance with course material, particularly programming assignments, please consult with the instructor during class, lab, or office hours.

While Canvas is used for assignment submission and grade posting, it is not the primary method for seeking academic support. To ensure timely responses, please direct questions to the instructor during class or lab or via email.

Grading Policy & Course Assessments

Final grade will be determined using the following grading weights:

in-class learning activities	15%
reading/writing assignments	10%
lab exercises	20%
programming projects	20%
mini-tests (quizzes)	12%
final examination	23%

The final numeric grade will be converted into a letter grade according to the grading scale below, and this letter grade will be submitted as the official course grade.

Overall Final	**Letter Grade**
94-100	A
90-93	A-
87-89	B+
84-86	B
80-83	B-
77-79	C+
74-76	C
70-73	C-
67-69	D+
64-66	D
60-63	D-
0-59	F

The following table shows how the course work is assessed against the course objectives:

	Mini-tests/Exams	Assignments (labs, programming projects, and other exercises)
CO01	✓	✓
CO02	✓	✓
CO03	✓	✓
CO04	✓	✓
CO05	✓	✓
CO06		✓
CO07	✓	✓
CO08	✓	✓
CO09		✓
CO10		✓

In-Class Learning Activities

In-Class Learning Activities are short, structured exercises completed primarily **during scheduled class meetings** and contribute **15% of the overall course grade**. These activities may include brief individual or group discussions, hand-drawn design sketches or flowcharts, short written responses (e.g., multiple-choice, true/false, or fill-in-the-blank questions), and collaborative problem-solving tasks. Typically lasting 5–15 minutes, they are designed to reinforce fundamental data-structures concepts and support preparation for laboratory work, programming projects, and examinations.

Grades are based on **active participation and engagement during on-site class meetings**; **no submissions** are needed. Because these activities occur in real time, **no make-up opportunities are provided**; students who miss an activity due to absence will **not** receive credit. However, to allow for unforeseen or emergency situations, students may miss one or two activities during the semester without penalty and still receive full credit for this portion of the course. Documented justification may be required when applicable.

Reading/Writing Assignments

These assignments are designed to help you understand the course topics, prepare for programming practices, and get ready for the tests (i.e., mini-tests and the final examination). More importantly, many of the test questions will be similar to these assignment questions.

Readings from the textbook will be assigned regularly. To get the most out of these readings, complete them before the material is covered in class. You'll find that many assignment questions are based on the textbook readings. After completing labs and projects related to the reading topics, *reviewing these materials will be especially beneficial and rewarding*.

The course will include various types of questions (such as true/false, fill-in-the-blank, multiple-choice, and short-answer questions) designed to test your understanding of the topics and help you prepare for your exams.

Lab Exercises

Lab exercises are designed not only to help you understand the course topics but also to prepare you for the programming projects. Submitting your lab work to Canvas by the deadline alone does ***not*** guarantee credit—you must have it *tested by the instructor in the lab to receive a grade*.

Please note that lab exercises and programming projects are separate assignments, and the credits earned from programming projects do not count toward your lab assignment grades, even though some labs may be part of projects.

Lab submission must follow the in Lab Exercise Submission requirements defined in Lab Syllabus.

Programming Projects

Throughout the semester, you will be assigned programming projects. Most projects will involve preparatory lab activities before you begin coding. These assignments will require dedicated programming time outside of scheduled lab hours.

The point value for each programming assignment may vary depending on its difficulty and workload. To receive full credit, all projects must be submitted through Canvas (https://elearning.salemstate.edu/) and then tested and reviewed with the instructor during your lab hours.

The project submission must following Project Submission Requirements (which can be found at Canvas).

AI Policy and Coding Assignment Grading

Online sources and generative AI tools (such as ChatGPT) may be used in the study of course topics and concepts. The primary approach to learning programming involves reading the assigned text chosen from the textbook, focusing the class lectures, doing lab exercises, and completing the coding assignments following the examples provided in the class. Online sources and AI tools should be used **only as supplementary aids**.

Simply using solutions (i.e., programs) found online or generated by AI tools as your submissions for coding assignments (labs, projects, and etc.) are considered plagiarism.

Please also note that online "solutions", particularly those generated by AI tools, may use "advanced programming techniques" that have ***not*** been covered in the course by the assignment deadline. Students are ***only*** allowed to use programming techniques that have been presented up to that point.

For each programming project and some labs, students must:

1. Submit program code to Canvas by the deadline.
2. Have the instructor test your code during lab hours.
3. Correctly answer the instructor's questions during the testing process.
4. Successfully modify your code to address a similar coding problem *within a couple of minutes on site* in the lab.

Due to the extensive testing process, timely submission of your solutions is crucial. Thoroughly understand your Java code. If you cannot adequately explain your code, make necessary modifications, and answer related questions during the in-lab testing, your project grade will be *significantly impacted, potentially resulting in a score of* ***zero***.

Tests (mini-tests/quizzes and final examination)

Mini-tests (quizzes) will be administered throughout the semester, along with a comprehensive final exam. The weight of each assessment in determining your final grade is outlined in the grading policies above.

Mini-tests will consist of questions in multiple-choice, true/false, fill-in-the-blank, and short-answer formats, as well as programming problems. These tests must be completed during designated class/lab periods or by the deadlines online at Canvas. Programming mini-tests will resemble smaller-scale versions of lab exercises and projects.

The final exam will be a comprehensive evaluation of all course material.

Makeup Examinations

Make-up exams/mini-tests are generally *not permitted unless* there is documented proof of an emergency. If you need to reschedule a test, you must arrange this with the instructor within one week of the original test date. The final exam make-up will be done in the university designated Make-up Exam Period.

Policy on Late Work

- Late submission of assessed assignments (including labs and projects) will incur the following penalties:
  - A **20%** deduction if submitted within 24 hours after the deadline.
  - A **40%** deduction if submitted more than 24 hours and up to 48 hours late.

  - A **60%** deduction if submitted more than 48 hours and up to 72 hours late.
  - An **80%** deduction if submitted more than 72 hours and up to 96 hours late.
  - A **100%** deduction (grade of zero) if submitted more than 96 hours late.
  - In addition, **no** assignments (including labs and projects) will be accepted *after the final examination*. Any work submitted after that time will receive a grade of **zero**.
- In the event of an emergency that prevents you from completing or submitting your assignments/labs/projects on time, you must send an *email* request for an extension. The instructor will respond with either the number of days granted for the extension or a new deadline. You should keep this email as proof of the extension approval. Please note that sending a request email does ***not*** automatically guarantee an extension.
- Please **double-check your submissions**, as they are typically graded after the deadlines. To ensure successful and correct submissions, download and review your submitted materials. For programming projects, download the programs and verify that the code compiles and runs correctly.

Study Groups

While I strongly encourage study groups, each student must submit their answers in their own words or solutions. If two submissions are highly similar, neither will receive credit.

When working on your programming projects, you may discuss project topics, algorithms, and methodologies with others. However, the coding must be entirely your own work. If two code submissions are identical or very similar, neither will receive credit, and further action may be taken, such as reporting the incident to the department or university.

Collaboration is encouraged for discussing project topics, algorithms, and methodologies. However, all code must be your original work. Identical or highly similar code submissions will result in ***zero*** credit for both parties and may lead to further disciplinary action.

Academic Integrity

Salem State University assumes that all students come to the University with serious educational intent and expects them to be mature, responsible individuals who will exhibit high standards of honesty and personal conduct in their academic life. All members of the Salem State University academic community have a responsibility to ensure that scholastic honesty and academic integrity are safeguarded and maintained. Cheating and plagiarism are unfair, demoralizing, and demeaning to all of us. Cheating, plagiarism, and collusion in dishonest activities are serious acts that erode the University's educational role and cheapen and diminish the learning experience not only for the perpetrators, but also for the entire community. It is expected that Salem State University students will understand and subscribe to the ideal of academic integrity and that they will be willing to bear individual responsibility for their work. Materials (written or otherwise) submitted to fulfill academic requirements must represent a student's own efforts.

Equal Access Statement

Salem State University celebrates and welcomes diverse learners of all types. Salem State University is committed to providing equal access to the educational experience for all students and to providing all reasonable academic accommodations, aids and adjustments.

Students who qualify for accommodations, aids or adjustments who have not previously done so should provide documentation to and schedule an appointment with the Center for Accessible Academic Resources and obtain appropriate services at caar@salemstate.edu. Any student who has a documented need for accommodations, aids or adjustments should share their accommodation letter with their instructors. We encourage students to communicate with their instructors as early in the semester as possible.

Salem State University complies with Section 504 of The Rehabilitation Act and The Americans with Disabilities Act.

University Emergency Statement

In the event of a university declared critical emergency, Salem State University reserves the right to alter this course plan. Students should refer to Salem State for further information and updates. The course attendance policy stays in effect until there is a university declared critical emergency. In the event of an emergency, please refer to the alternative educational plans for this course located at Canvas (https://elearning.salemstate.edu/). Students should review the plans and gather all required materials before an emergency is declared.

Coursework Expectations and Schedule

Students enrolled in this four-credit course should plan to spend approximately three hours per week attending class and three hours in lab attendance or practice. Additionally, **a minimum of eight hours per week outside of class and lab** is required for course-related work, beyond the six hours spent in class and lab sessions.

Students are responsible for adhering to Salem State University's academic regulations, including those pertaining to academic integrity, as outlined in the college catalog. It is essential that students complete all course requirements and keep up with course content, even in absences. The following table outlines the course schedule (subject to some adjustments or minor changes as the class goes on), including the topics covered each week of the semester and the final examination time. Detailed course requirements and advancements can be found at Canvas (https://elearning.salemstate.edu/).

Week	Dates	Contents (textbook chapters and others)
1	1/20—1/23	CSC115-Reviews Java Primer (Ch 1)
2	1/26—1/30	OOD (Ch2) Fundamental Data Structures (Ch3)
3	2/2—2/6	Fundamental Data Structures (Ch3)
4	2/9—2/13	Algorithm Analysis (Ch 4)
5	2/16—2/20	Algorithm Analysis (Ch 4) Recursion (Ch 5)
6	2/23—2/27	Stacks/Queues/Deques (Ch 6)
7	3/2—3/6	List & Iterator ADT (Ch 7)
8	3/9—3/13	Trees (Ch 8)
9	3/16—3/20	**Spring Recess**
10	3/23—3/27	Trees (Ch 8) Priority Queues (Ch 9)
11	3/30—4/3	Maps/Hash/Set (Ch 10)
12	4/6—4/10	Search Trees (Ch 11)
13	4/13—4/17	Sorting/Selection (Ch12)
14	4/20—4/24	Text Processing (Ch 13) Graph (Ch 14)
15	4/27—4/29	Graph (Ch 14) **Reviews**
16-17	5/4—5/11	**Final Examination** **May 4 (Monday), 11:00am-1:00pm, MH210**
Notes: Detailed course contents are available on Canvas at https://elearning.salemstate.edu/; visit the instructor's homepage at http://weblab.salemstate.edu/~byi/ for more information.		

Please remember that if, for any reason, you decide to drop this course, you MUST do so officially through the Registrar's office. The last day to withdraw from a course this semester is **April 10th**.

Note: This syllabus represents the intended structure of the course for the semester. If changes are necessary, students will be notified in writing and via all regular class communication mechanisms (class discussion, emails, and/or the course link at Canvas https://elearning.salemstate.edu).