

SCIS 313: Data Structures and Algorithm Analysis

Spelman College, Computer Science Department
Spring 2026

Course Information

Meeting Time: Tuesday & Thursday, 9:25 AM
Location: Tapley 227
Credit Hours: 4 credits
Prerequisites: CIS 215; Co- or prerequisite: Calculus I

Instructor Information

Instructor: Dr. Antonio Khalil Moretti
Email: antoniomoretti@spelman.edu
Office: Tapley 218
Office Hours: Tuesday 11:00 AM – 12:00 PM
(also welcome to stop by T/Th 10:40 AM – 1:00 PM)

Course Description

Data Structures and Algorithm Analysis covers basic concepts of data structure techniques and associated storage allocation and management algorithms. The course examines methods for processing a variety of data structures including lists, trees, strings, arrays, and stacks. The course includes advanced data structures and design and analysis of algorithms. Topics include balanced trees, sorting, heaps, selection, string matching, graph algorithms, and algebraic problems. Algorithm design techniques include divide-and-conquer and dynamic programming, as well as breadth-first and depth-first search. The course also includes methods for solving common summations and recurrences as well as an introduction to NP and computability.

Learning Outcomes

At the end of this course, students will be able to:

- Design and analyze data structures and algorithms for general problem solving using computers
- Analyze the performance of algorithms using asymptotic notation
- Compare and select among algorithmic approaches such as divide-and-conquer, greedy algorithms, and dynamic programming

- Apply algorithm design methods to practical problems
- Demonstrate an understanding of the relative performance of different categories of problems
- Compare and contrast varied approaches for applying algorithms to problem domains
- Explain features of various algorithms
- Demonstrate increased technical writing and oral presentation skills

Required Textbook

Weiss, Mark Allen. *Data Structures and Algorithm Analysis in C++*. Latest Edition.

Recommended References

- Shaffer, Clifford A. *Data Structures and Algorithm Analysis* (Edition 3.2 C++ version).
- Cormen, Thomas H., et al. *Introduction to Algorithms*. MIT Press.
- Sedgewick. *Algorithms in C++*. Addison-Wesley.

Course Topics

Weeks 1–7: Foundations

- Empirical analysis and timing experiments
- Big-O notation and asymptotic analysis
- Analyzing iterative code and loops
- Mathematical foundations (logarithms, summations, proof techniques)
- Recursion and recurrence relations
- Searching and sorting algorithms
- Data structures review (arrays, lists, stacks, queues, trees, heaps, hash tables)

Weeks 8–13: Advanced Topics

- Amortized analysis
- Balanced trees (AVL trees)
- Graph algorithms (BFS, DFS, MST)
- Greedy algorithms
- Dynamic programming
- Complexity theory (P, NP, NP-completeness)

Week 14: Review & Integration

Week 15: Final Exam

Grading Breakdown

Component	Weight
Exams (3–4 total)	65%
Exam 1 (Week 6)	15%
Exam 2 (Week 9)	15%
Exam 3 (Week 12)	15%
Final Exam (Week 15)	20%
Projects (3–4 presentations)	25%
Homework (weekly)	10%

Grading Scale: 90–100 A, 80–89 B, 70–79 C, 60–69 D, <60 F. Plus and minus grades will be assigned within these 10-point ranges.

Course Components

Homework (10%)

Homework problems will be assigned weekly. Students are expected to attempt all problems and submit screenshots of their work along with written solutions. Homework is completion-based and is graded on effort and submission. **Students should be prepared to present homework solutions in class each week.** Students may decline to be recorded during their presentations if they prefer.

Projects (25%)

Students will complete 3–4 group presentations (2–3 students per group) on advanced topics including sorting algorithms, tree structures, graph algorithms, or NP-complete problems. Each group will become class “experts” on their assigned topic and will present their findings to the class. Detailed instructions will be provided for each project.

Exams (65%)

There will be 3–4 exams throughout the semester, including a comprehensive final exam. Exams will test understanding of algorithms, data structures, analysis techniques, and problem-solving skills.

Course Policies

Class Recording

All class sessions will be recorded via Zoom for educational purposes. Students are required to attend class in person. Recordings are provided as a supplementary resource and do not substitute for attendance. **Students may opt out of being recorded when presenting homework problems**, but all students must still be prepared to present weekly.

Attendance

Students are required to be present and on time for each class and are responsible for all material covered in class whether present or absent. If a student has missed two or more weeks of class (6 or more class meetings), the student may be withdrawn without further notice.

Late Policy

No makeup tests will be given except by prior agreement with the instructor or a dean's excuse. Programming assignments or projects turned in late will be penalized 10 percentage points per day, with a maximum penalty of 50 points. Assignments may not be turned in for credit more than one week late.

Inclement Weather or Emergency Policy

If a class is not held at the assigned time, all work scheduled for that day is automatically rescheduled for the next scheduled class period. Any projects, assignments, or homework due on a missed day become due at the next scheduled class period.

Use of Generative AI

This course assumes that work submitted by students will be created by the students themselves, working individually or in groups as directed. It is considered a violation of the Academic Integrity Policy if a student has another person/entity complete work for them. This includes purchasing work from a company, hiring a person to complete an assignment, and/or using generative AI tools (such as ChatGPT). **Generative AI may be used for programming assignments only when explicitly directed by the instructor as part of the assignment.**

Academic Integrity

At the heart of Spelman College's mission is academic excellence, along with the development of intellectual, ethical, and leadership qualities. All members of the academic community are expected to uphold a commitment to high ethical standards. Students are expected to read and abide by the Spelman College Code of Conduct and behave as mature and responsible members of the academic community.

Common violations include:

- **Cheating:** Using or attempting to use unauthorized assistance, materials, or study aids in examinations or academic work (including unauthorized use of AI tools, using cheat sheets, soliciting exam information from other students, etc.)

- **Plagiarism:** Using the ideas, data, or language of another without specific and proper acknowledgment (including submitting downloaded/purchased papers, copying from the internet without citation, etc.)

Consequences:

- Cheating on an examination: Zero on the exam and possibly an F in the course
- Cheating on homework/projects: Zero on the assignment and a one-letter grade penalty for the course
- All cases will be forwarded to the Office of Undergraduate Studies for additional disciplinary action

Disability Statement

Spelman College is committed to ensuring the full participation of all students in its programs. If you have a documented disability (or think you may have a disability) and need a reasonable accommodation to participate in class, complete course requirements, or benefit from the College's programs or services, contact the Office of Disability Services (ODS) as soon as possible. To receive academic accommodation, you must be appropriately registered with ODS. For further information, contact ODS at 404-270-5289, located in MacVicar Hall, Room 106.

Excused Absences

The Dean of Undergraduate Studies will issue official excuses for students who have missed two consecutive class sessions (T/Th). Students must request an excused absence within one week of the absence. All requests must be accompanied by valid documentation.

Excused absences include:

- Student illness or injury
- Critical illness/death in the immediate family
- Short-term military obligation
- Jury duty or court subpoena
- Religious holidays

Do not schedule any event, interview, or plant trip in conflict with any examination for this course. Such absences will not be approved or excused.

COVID-19 Health Guidance

If you test positive for COVID-19 or experience respiratory viral symptoms:

- Stay home and away from others
- Seek medical treatment if needed
- Resume normal activities when symptoms are improving and you are fever-free (without medication) for at least 24 hours
- Notify Student Health Services when absent due to COVID-19

For the College's full statement, visit:

<https://spelmancollege.activehosted.com/index.php?action=social&chash=7e9e346dc5fd268b49bf41852262&s=bd42c316b9c6c8ff903f2c741cb31530>

Tentative Course Schedule

Week	Topic	Description	Reading
1	Empirical Analysis & Timing Code	Why algorithm analysis matters, timing experiments, growth rates	Weiss Ch. 2.1–2.3
2	Big-O Notation	Asymptotic notation (O , Ω , Θ), formal definitions, proving bounds	Weiss Ch. 2.4
3	Analyzing Iterative Code	Analyzing loops, nested loops, common patterns	Weiss Ch. 2.4
4	Mathematical Foundations	Logarithms, exponents, summations, series, proof techniques	Weiss Ch. 1.4–1.7, Appendix
5	Recursion & Recurrence Relations	Recursive algorithms, Master Theorem, substitution method	Weiss Ch. 2.4
6	Searching & Sorting	Binary search, merge sort, quick sort, complexity analysis	Weiss Ch. 7
	Exam 1		
7	Data Structures Review	Arrays, lists, stacks, queues, trees, heaps, hash tables	Weiss Ch. 3–5
8	Amortized Analysis	Dynamic arrays, aggregate method, accounting & potential methods	Weiss Ch. 3.5
9	Balanced Trees (AVL)	Why balancing matters, rotations (LL, RR, LR, RL), guaranteed $O(\log n)$	Weiss Ch. 4.4
	Exam 2		
10	Graphs	Representations (matrix vs list), BFS & DFS, applications	Weiss Ch. 9
11	Greedy Algorithms	Greedy strategy, activity selection, minimum spanning trees	Weiss Ch. 9.5–9.6
12	Dynamic Programming	Optimal substructure, overlapping subproblems, classic problems	Weiss Ch. 10
	Exam 3		
13	Complexity Theory	Problem classes (P, NP), NP-completeness, decidability overview	Weiss Ch. 9.7
14	Review & Integration	Comprehensive review, choosing the right approach, practice problems	Review all
15	Final Exam	Comprehensive	

This schedule is subject to modification. Any changes will be announced in class and posted on the course website.