CSCI 3230 DATA STRUCTURES

Summer 2022

Instructor:	Dr. Weitian Tong Assistant Professor in Computer Science
Email:	wtong@georgiasouthern.edu
Website:	www.weitiantong.com
Office:	IT 2305
Office hours:	By appointment
Lecture class:	Online Course, Lecture, Entirely at a distance
Important Dates:	Please refer to Summer 2022 University Calendar for details. • May 16, Class begins and Attendance Verification • May 16 – 18, Drop/Add • June 1, Last day to withdraw without academic penalty • June 15, Last day of classes • June 16, Final Exams
CRN and Credits:	CRN 52208, 3.0 credit hours
Prerequisite(s):	A minimum grade of "C" in CSCI 1302 and MATH 2130 or CSCI 2625.
Course Content:	This course introduces basic data structures and their application using the Java programming language. We will introduce • Analysis of Algorithms: mathematical framework for evaluating the efficiency of algorithms and data structures • Recursion • Linear data structures such as lists, stacks, queues. • Maps and hash tables • Binary trees, heap and search trees • Sorting algorithm • The storage and exploration of graphs
Textbook:	There is **no** required textbook.
Materials:	• All course materials (such as syllabus, slides, and homework descriptions) are available on Folio. • The actual course schedule will be updated timely on our course website: "www.weitiantong.com/teaching/DS-course"
Assessments:	• Written Homework (**5 times**): 50%; • Coding Quiz (about **14 times**): 50%;
Grading Scale:	A: [90, 100]; B: [80, 90); C: [70, 80); D: [60, 70); F: [0, 60).

Course objectives: Proficiency in data structures (including containers, trees, heaps, priority queues, hash tables and graphs) and algorithms for manipulating data (including sorts and tree traversals). Develop the ability to implement those data structures and analytic understanding of the time and space trade offs of different data structures and different implementations. Write larger programs that require interacting with a data structure.

Learning Outcomes: At the end of the course, a successful student should be able to:

- Be able to perform complexity analysis on any algorithm covered in class.
- Be able to identify a preferred algorithm and data structure using complexity depending on the problem being solved and the nature of the data.
- Be able to implement recursive solutions and trace any recursive algorithm.
- Be able to implement standard operations (as appropriate, these are insert, delete, search) in all data structures: stack, queue, tree (BST, AVL, B/B+), heap, hashtable, graphs.
- Be able to implement a doubly linked list and a multi-list data structures.
- Be able to implement a stack, in array and in linked list, in order to solve problems such as building expression trees, paragraph matching, conversion to postfix.
- Be able to implement straight queue and circular queue, in linked list and dynamic structure, in order to solve problems such as buffering input data, simulation.
- Be able to implement sorting algorithms on arrays, dynamic (in memory structures) and externally: insertion, selection, shell, heap, quick, merge, radix.
- Be able to implement hashing and hash functions (open addressing, closed addressing) various collision resolutions (linear probe, quadratic probe, double hashing) in solution of a problems such as building a symbol table.
- Be able to implement binary search trees on arrays and on dynamic structures.
- Be able to implement graphs using basic graph representation, such as adjacency matrix, adjacency list, edge list, etc.
- Understand at least one dynamically balanced binary search tree algorithm (AVL, DSW, ...)
- Understand basic graph search algorithms, such as BFS and DFS.

Course Policy:

- **Academic Honesty**: Consultation with fellow students is encouraged. However, directly copying another student's work (past or present) defeats the purpose of the assignments and exams and is an honor code violation. Unless otherwise noted, you are expected to complete all assignments **individually**. Violations of the Georgia Southern University academic honesty policy (including cheating and plagiarism) are taken very seriously. Any violation of this policy will become part of the student's permanent educational record. More information on the academic dishonesty policy and procedure can be found at: Academic Dishonesty
- **Late Assignments, make-up work/tests**: **No** late work will be accepted and there is **no** make-up work available in this class, unless the absence is due to a situation that is beyond your control and can be **verified by written documentation.**
- **Attendance**: As this is an online course (Entirely at a distance) that is delivered 100% through distance education technology, all course materials will be posted on Folio and/or the course website. Detailed guidelines will be provided each day. Please follow the guidelines to complete every assigned task.
- **Grading disagreement**: Requests for re-evaluation of assignments and quizzes must be made by email **as soon as possible** after the grades are available.

- **Communication**: **Important news will be announced on Folio**. To ensure effective communication, you are required to check Folio or your university email (if the announcements are pushed to your email) for all communications related to this class. It is your responsibility to check your Folio account and university email daily, as many important communications will be done via email.
- **Distribution of course materials**: Lecture notes, assignments/quizzes/exams, as well as their sample solutions, are prepared for students in this course **only**. Any other use, distribution, or posting in places outside this course are prohibited without written permission from the instructor.
- **Illnesses**: We want you to take appropriate precautions for your health as well as the well-being of your classmates. If you become ill during the term, please contact me immediately. We will work through what you will need to do, to either continue working in class or make up work that might have been missed during your absence. If you have an illness that would result in an extended absence, you will need to contact the Dean of Students office. In the event of serious illness, injury, or extenuating circumstances, the DOS office will notify professors at your request. If you need to self-report either a confirmed or suspected positive COVID-19 diagnosis, have received self-quarantine requirements, or have symptoms with pending test results, please complete the CARES Center COVID-19 self-reporting form (through the MyGeorgiaSouthern portal under “COVID-19 Information & Resources”). You may also reach the CARES Center by using the MyGS mobile app, calling 912-478-CARE (M-F 8am-5pm), or emailing covidsupport@georgiasouthern.edu. The CARES Center should not be used for medical advice. If you need medical advice, you need to call your health provider or 911.
- **Disability-related Accommodations**: Georgia Southern University is committed to providing reasonable accommodations to students with documented disabilities as required under federal law. Disabilities may include ADD or AD/HD, autism spectrum disorders, brain injuries, chronic medical conditions, communication disorders, hearing impairment, learning disabilities, mobility impairment, psychological disorders and visual impairment. The purpose of disability accommodation is to provide equal access to the academic material and equal access to demonstrate mastery of the material. If you have a disability and need accommodations, please contact the Student Accessibility Resource Center (SARC). You will need to meet with a SARC staff member who can help you gather documentation of your disability or refer you to an appropriate resource for assessment. Once documentation of the disability is approved, SARC staff will provide you with an accommodation letter detailing the approved accommodations which you should present to me so we can discuss and implement your accommodations. Disability accommodations work best starting at the beginning of the semester, but can be approved and started at any point in the semester. Accommodations start at the time the accommodation letter is presented to faculty within reasonable timelines; accommodations are not given retroactively. SARC on the Statesboro campus is located on the second floor of Cone Hall and the phone number is (912) 478-1566. SARC for Savannah and Liberty campuses is located on the second floor of Memorial College Center and the phone number is (912) 344-2572.
- **Equal Opportunity & Title IX**: Please refer to “Equal Opportunity & Title IX” for more information.
  - **Reporting**: Georgia Southern University does not discriminate on the basis of sex, race, color, sexual orientation, gender identity or expression, national origin, religion, age, veteran status, political affiliation, or disability. While students are encouraged to share with faculty any issues or concerns they may be having, please be aware

there are reporting requirements which are a part of the job requirements at Georgia Southern University. For example, if you disclose an issue of sexual misconduct, the information will be kept as private as possible but faculty and staff are required to bring it to the attention of the institution's Title IX Coordinator/Director of Equal Opportunity and Title IX.

- **Pregnant and Parenting Students**: Georgia Southern University does not discriminate against students who are pregnant, give birth, experience a false pregnancy, termination of pregnancy, or recovery therefrom. Students should work with their faculty as soon as possible to arrange appropriate accommodations based on this status; delays in reporting may impact available accomodations. Students will be treated similarly to other similarly situated students. Absences from class may be excused due to pregnancy or childbirth for as long as medically necessary. At the conclusion of medical leave, the student will be allowed to return to the same academic status. Medical certification may be requested from the student. Both students and faculty are able to consult with the Title IX Coordinator regarding any questions or issues which arise.

Enjoy the class!

CSCI 3230 Data Structures

Basic Info

Instructor: **Dr. Weitian Tong** Assistant Professor at Department of Computer Science
Office: CIT 2305
Email: wtong@georgiasouthern.edu
Lecture time: N/A; Online Course, Entirely at a distance
Office hours: By appointment

Course Schedule (timely updated!)

-

Date	Topics	Useful Resources
Module 1:	• Class 1: ◦ Attendance Verification ◦ Check the Syllabus on Folio ◦ Quickly explore the resources in the right column → ◦ Introduction: ▪ Video: Data Types vs. Abstract Data Types	**Visualizing data structures and algorithms:** • Visualgo • Data Structure Visualizations • Data Structures and Algorithms Visualizations **Search for coding examples and hands-on practice:** • GeeksforGeeks • LeetCode

	▪ Video: Data Structure Definition & Advantages ▪ Video: Types of Data Structures • Module 2: Java background (refresh your memory) ◦ Classes and Objects ◦ Interfaces ◦ Strings, Arrays ◦ Overloading, Overriding ◦ Video: Object Oriented Programming ◦ Video: Comparable Generics ◦ Video: Generic Programming • Class 3: Analysis of Algorithms ◦ Video: Basics of Asymptotic Analysis: Part 1; Part 2; Part 3; Part 4; Part 5; • Class 4: Analysis of Algorithms ◦ Video: Guidelines for Asymptotic Analysis: Part 1; Part 2;	• How to Code Without Plagiarizing **Free online textbook:** • An Introduction to the Analysis of Algorithms • Algorithms, 4th Edition • Tutorialspoint **YouTube Channels:** • Neso Academy: Data Structure • Data Structures by Dr. Rob Edwards • HackerRank: Data Structure and Algorithms **Latex:** • Download Latex • Learn LaTeX in 30 minutes • Youtube: LaTex Tutorial for Beginners Full Course **Asymptotic Analysis:** More resources • Asymptotic Analysis

	◦ Video: Asymptotic Analysis Examples: Example 1; Example 2; Example 3; • Class 5: ◦ Recursion ◦ Divide and Conquer ◦ Coding Quiz 1: Recursion	• Asymptotic Notations • Big-O-cheatsheet • Video: Introduction to Asymptotic Analysis **Fibonacci:** a recursion example • Fibonacci • Coursera Video: Fibonacci I • Coursera Video: Fibonacci II • Coursera Video: Fibonacci II
H1 (Asymptotic Analysis and Recursion)		
Module 2:	• Class 1: Linear DS ◦ Array ◦ ArrayList ◦ Array vs ArrayList in Java Video • Class 2: Linear DS: ◦ Singly Linked List Video ◦ Doubly Linked List Video ◦ Coding Quiz 2: LinkedList • Class 3: Linear DS: ◦ Stack Video ◦ Queue Video ◦ Coding Quiz 3: Stack	**Visualizations:** • ArrayList • LinkedList • Array-based stack, List-based stack • Array-based queue, List-based queue • Hash Table • Hash Table: Linear Probing vs Quadratic Probing vs Double Probing vs Separate Chaining

	• Class 4: Maps and Hash Tables ◦ Hashing ◦ Separate Chaining; Open Addressing; Double Hashing • Class 5: Trees ◦ General Tree Definitions and Terminology ◦ Binary Tree ◦ Tree Traversals (Inorder, Preorder and Postorder) ◦ Binary Tree Properties	**Tree:** • Binary Tree questions • Tree traversal questions
H2 (Linear Data Structures and Hash Tables)		
Module 3:	• Class 1: ◦ PriorityQueue ◦ Heap ◦ Coding Quiz 4: Heap It is a bonus quiz! • Class 2: Search Trees ◦ Binary Search Tree: Search and Insertion; Delete • Class 3: ◦ Coding Quiz 5: Search Tree • Class 4: ◦ AVL tree: Insertion , Delete ◦ Coding Quiz 6: AVL Tree • Class 5:	**Visualizations:** • Heap • Heap on Visualgo • Binary search tree and AVL tree on Visualgo • Binary search tree • AVL Tree • Splay Tree • Red-Black Tree

	◦ Red-Black Tree ◦ Coding Quiz 7 Red-Black Tree It is a bonus quiz!	
H3 (Trees)		
Module 4:	• Class 1: Comparison Sorts ◦ Sorting Terminology ◦ Selection Sort, Bubble Sort, Insertion Sort ◦ Coding Quiz 8: Insertion Sort • Class 2: Comparison Sorts ◦ Merge Sort ◦ HeapSort ◦ Coding Quiz 9: Merge Sort • Class 3: Comparison Sorts ◦ QuickSort ◦ Lower bound for comparison sorts ◦ Coding Quiz 10: Quick Sort • Class 4: Linear Sorts ◦ Sorting without comparison of elements ◦ Bucket Sort ◦ Counting Sort ◦ Radix Sort ◦ Coding Quiz 11: Counting Sort • Class 5: Graph and graph representations	**Visualization:** • Sorting algorithms on Visualgo • Comparison Sorting Algorithms • Bucket Sort • Counting Sort • Radix Sort • Graph representations on Visualgo

	◦ Graph Introduction ◦ Graph and its representations	
H4 (Sorting)		
Module 5:	• Class 1: Graph Algorithms ◦ Depth First Search ◦ Video: Depth First Search (Algorithm, Examples, and Code) ◦ Coding Quiz 12: DFS • Class 2: Graph Algorithms ◦ Breadth First Search ◦ Coding Quiz 13: BFS • Class 3: Graph Algorithms ◦ Shortest Path: Dijkstra's algorithm ◦ Minimum Spanning Tree ▪ Prim's algorithm, Kruskal's algorithm ◦ Coding Quiz 14: Dijkstra's algorithm	**Visualization:** • DFS and BFS on Visualgo • Minimum Spanning Tree on Visualgo • Shortest on Visualgo • Depth First Search • Breadth First Search • Dijkstra's algorithm • Prim's algorithm • Kruskal's algorithm
H5 (Graphs)		