Ahmedabad University

Course	CSE205 Data Structures	Semester	Monsoon Semester 2024
Faculty Name(s)	Amit Nanavati	Contact	amit.nanavati@ahduni.edu.in
School	SEAS	Credits	4
GER Category:	Not Applicable	Teaching Pedagogy Enable:NO	P/NP Course: Can not be taken as P/NP

Schedule	Section 1	10:00 am to 11:00 am	Fri	01-08-24 to 26-11-24
		11:00 am to 12:00 pm	Fri	01-08-24 to 26-11-24
		01:00 pm to 02:30 pm	Mon	01-08-24 to 26-11-24
		01:00 pm to 02:30 pm	Wed	01-08-24 to 26-11-24
	Section 2	03:00 pm to 04:00 pm	Mon	01-08-24 to 26-11-24
		04:00 pm to 05:00 pm	Mon	01-08-24 to 26-11-24
		05:30 pm to 07:00 pm	Tue	01-08-24 to 26-11-24
		05:30 pm to 07:00 pm	Thu	01-08-24 to 26-11-24

Prerequisite	CSE100 Fundamentals of Computer Programming & CSE101 Object Oriented Programming Lab/CSE108 Object Oriented Programming Lab & MAT101 Discrete Mathematics

Antirequisite	Not Applicable
Corequisite	Not Applicable
Course Description	The course covers basic data structures and techniques for design and analysis of data structures with a rich set of applications in research and industry. The course provides a thorough introduction to the analysis of the complexity of algorithms. It shows how to use these analysis for algorithms using the basic data structures like Lists, Stacks, Queues, Binary Search Trees, Heaps and Balanced Search Trees for storing data, sorting and searching problems. We will visit Sorting and Hashing. We will also introduce tools and techniques for computational analysis of these basic data structures. It covers also some more advanced problems tree algorithms.
Course Objectives	• Rigorous introduction of data structures like lists, trees, heaps, and graphs. • Mathematical foundations for analysis of algorithms on these structures. • Hands-on experience in writing programs using the structures.
Learning Outcomes	The student should be able to • use the data structures: stacks, queues, trees, heaps and hashing in solving practical problems • choose appropriate tools based on the knowledge of complexity of a given problem • modify known algorithms for a given problem • implement a solution in a programming language in a proper way
Pedagogy	The slides for the lectures are provided successively. The students should maintain their own notes for the lectures and lab sessions. When a problem is explained and worked out with examples and exercises, there is a short quiz checking the understanding of the concepts. We will be using popular coding platforms such as Leetcode/Hackerrank etc. to do labwork.
Expectation From Students	Students should maintain their own notes from the lectures and the projects.

Assessment/Evaluation	• Mid-Semester Examination: ○ Viva - 10% ○ Written - 15% • End Semester Examination: ○ Final Viva - 10% ○ Written - 30% • Other Components: ○ Coding Quiz - 30% ○ Labwork - 5%
Attendance Policy	As per Ahmedabad University Policy.
Project / Assignment Details	There are midterms, coding quizzes and vivas during the course. There are several lab assignments. We will balance concepts and practice.
Course Material	Reference Book • Data Structures and Algorithms in Java, Adam Drozdek, 3rd Edition, Cengage Learning, ISBN: 978-81-315-0655-4, Year: 2007, • Data Structures and Algorithms Using Python, Rance D. Necaise, ISBN: 978-0-470-61829-5, • Data Structures and Algorithms in Java, Adam Drozdek, 3rd Edition, Cengage Learning, ISBN: 978-81-315-0655-4, Year: 2007, • Data Structures and Algorithms Using Python, Rance D. Necaise, ISBN: 978-0-470-61829-5, • Data Structures and Algorithms in Java, Adam Drozdek, 3rd Edition, Cengage Learning, ISBN: 978-81-315-0655-4, Year: 2007, • Data Structures and Algorithms Using Python, Rance D. Necaise, ISBN: 978-0-470-61829-5, •
Additional Information	

Session Plan

NO.	TOPIC TITLE	TOPIC & SUBTOPIC DETAILS	READINGS,CASES,ETC.	ACTIVITIES	IMPORTANT DATES
1	Introduction	Overview, motivation for studying data structures, concept of algorithm	Chapter 1, 2 from textbook, slides	Before slides presentation, discussion about formal requirements of the course	
2	Introduction	Introducing concept of running time, recursion tree, pseudocode	Chapter 1, 2 from the textbook, slides	Solving problems, first programming exercises	
3	Lab session: Solving problems	Recalling basic concepts of Java. Programming exercises using Java: arrays, conditionals, loops. Solving problems by writing a program in Java.	chapter 1 from reference book, case study: random access file	programming exercises	
4	Growth of functions	Asymptotic notation, definitions, examples	Chapter 3 from the textbook, slides	Explanations, discussion of the concepts, solving problems	
5	Growth of functions	Solving problems and exercises using Java programs (e.g. to compute values of recursions)	Chapter 3 from the textbook	problems solved by writing java program whenever possible	
6	Lab session: One-dimensional array	Seraching and sorting algorithms in Java. Insertion, deletion, seraching: linear search, binary search sorting: bubble sort, insertion sort, mergesort	chapter 9.1, 9.3.4 from the reference textbook, case study: adding polynomials	programming exercises, discussing running time, short quiz	
7	Solving recurrences	Introduction to different methods of estimating running time for recursive algorithms. Defining recurrences. Maximum subarray algorithm. Algorithms for matrix multiplication.	Chapter 4 from textbook, slides	After presenting slides, solving problems, possibly small programming tasks	
8	Solving recurrences	Different methods for solving recurrencies. Substitution. Recursion tree method. Master method	Chapter 4 from textbook, slides	Solving problems, exercises	

9	Lab session: Two-dimensional array	Programming exercises: add two matrices, multiply matrices, maximum subarray algorithm, Strassen algorithm, magic square matrix	chapter 3.6 from the reference book, and possibly chapter 8.1 from the reference book: graph representation (as a matrix)	programming exercises	
10	Probabilistic analysis of algorithms	Basic notions from probability theory. Discrete probability distribution. Discrete random variable. Expected value of a random variable. Indicator random variables. Randomized algorithms.	Appendix C2, C3, chapter 5 from the textbook, slides	Students should take notes, implementation of random number generator	
11	Probabilistic analysis	Solving problems and exercises	Appendix C2, C3, chapter 5 from the textbook, slides	Explanations, solving problems, short quiz	
12	Lab session: Representing graphs and trees	Programming exercises in java: implementing tree in an array, tree travelsals, BFS, DFS, etc.	chapter 6.4 in reference book, case study: expression trees	programming exercises	
13	Heaps, Heapsort	Concept of a heap, properties, kinds of heaps, algorithms for heaps, (MAX-HEAPIFY, Building a heap) . Complexity analysis.	Chapter 6 from the textbook, slides		
14	Heaps, priority queues	Different kinds of priority queues, operations on queues, applications	Chapter 6 from the textbook, exercises, problems, implementation	solving problems,	
15	Lab session: Programming in Java	Heapsort, priority queues, Williams' and Floyd's algorithm to create a heap	chapter 4.4 (priority queues in java.util) and 6.9 from reference book,		
16	Quicksort	Advantages (sorting in place), explaining the algorithm, pseudocode, analysis	Chapter 7 from the textbook, slides		
17	Quicksort	solving problems, exercises	Chapter 7 from the textbook, explanations, proofs	programming exercises	

18	Lab session: Programming in Java	Quicksort, different versions of the algorithm	chapter 9.8, programming assignments, case study: polynomial multiplication	short quiz	
19	Sorting in linear time	Concept of decision tree, Counting sort, Radix sort, Bucket sort	Chapter 8 from the textbook, slides		
20	Sorting in lienar time	Solving problems and exrcises	Chapter 8		
21	Lab session: Programming in Java	Counting sort, Radix sort, Bucket sort Randomized versions of the algorithms,	Chapter 8, Implementation of different versions of the algorithms		
22	Medians and order statistics	Finding minimum, maximum, concept of order statistic, algorithm SELECT	Chapter 9 from the textbook, slides		
23	Medians and order statistics	Solving problems and exercises	Chapter 9 solving problems		
24	Lab session: Programming in Java	computing medians and order statistics	Chapter 9, implementation of the algorithms	short quiz	
25	Stacks, queues, linked lists	Dictionary algorithms, policy for stacks, and queues specific operations, different kinds of linked lists, operations on lists	Chapter 10 from the textbook, slides		
26	Stacks, queues, linked lists	dynamic memory allocation	Chapter 10, solving problems		
27	Lab session: Programming in Java	Using stacks, queues, linked lists, dynamic memory allocation	chapter 3 from the reference,case study: a library		
28	Hash tables	Direct address tables Hash functions, open addressing ways to solve collisions	Chapter 11 from the textbook, slides		
29	Hash tables	Solving problems and exercises	Chapter 11		

30	Lab session: Programming in Java	Using hash tables, studying different hashing functions in practice, hashing in java.util	chapter 10 from the reference	short quiz	
31	Binary search trees	Concepts, definitions, basic algorithms	Chapter 12 from the textbook, slides		
32	Binary search trees	Analysis of algorithms, problems, exercises	Chapter 12		
33	Lab session: Programming in Java	Binary search trees, implementation in java	chapter 6.2 from reference text		
34	Red-black trees	Motivation, concepts, algorithms (insertion, deletion, etc)	Chapter 13 from the textbook, slides		
35	Red-black trees	Algorithms continued. Exercises.			
36	Lab session: Programming in Java	Implementation of red-black trees	Chapter 13	programming exercises, short quiz	
37	AVL trees	Definitions, properties, operations, Comparison with RB trees	chapter 6 from the reference, case study: computing word frequencies		
38	Self-adjusting trees, Splaying	Problems and exercises case study: Computing word frequencies	chapter 6 from the reference book,		
39	Lab session: Programming in Java	Trees in java.util	chapter 7 from reference book		
40	Projects, tests	reviewing projects, tests, questions			
41	Reflections and reviews				
42	Reflections and reviews				

43	End-semester exam				