Savitribai Phule Pune University Second Year of Computer Science and Engineering - 2024 Pattern		
PCC-201-CSE : Data Structures		
Teaching Scheme	**Credits**	**Examination Scheme**
Theory : 03 Hours/Week	03	CCE : 30 Marks End-Semester: 70 Marks

Prerequisite Courses:

1. Programming and Problem Solving
2. Fundamentals of Programming Languages

Companion Course : Data Structures Lab

Course Objectives: The course aims to:

1. To understand importance of data structures in context of writing efficient programs
2. To be able to implement linear and non-linear data structures

Course Outcomes: Upon successful completion of this course, students will be able to:

- CO1: **Understand** and **Analyze** various types of data structures and algorithms
- CO2: **Apply** various sorting and searching algorithms for given problem
- CO3: Make **Use** of Stacks and Queues to **solve** the given problem
- CO4: **Analyze** different hashing techniques and collision resolution strategies.
- CO5: **Demonstrate** basic operations on trees and graphs

Course Contents

Unit I - Introduction to Data Structures and Algorithms (09 Hours)

Introduction: Introduction to Data Structures: Abstract Data Types (ADT), Linear and Non-linear, Static and Dynamic, Persistent and Ephemeral data structures

Algorithms: Space complexity, Time complexity, Asymptotic notation- Big-O, Theta and Omega, finding complexity using step count method, Analysis of programming constructs-Linear, Quadratic, Cubic, Logarithmic.

Algorithmic Strategies: Introduction to algorithm design strategies- Divide and Conquer, and Greedy strategy

Case Study:E-commerce Product Sorting using Divide and Conquer strategy Google Calendar application using Greedy strategy

Unit II - Linear Data Structures, searching and sorting (09 Hours)

Overview of Array, Array as an Abstract Data Type, Operations on Array, Storage Representation, Multidimensional Arrays[2D, nD], Sparse matrix representation using 2D Searching: Sequential Search/Linear Search, Binary Search, Fibonacci Search, and Indexed Sequential Search.

Sorting: Concepts- Stability, Efficiency, and Number of Passes, Internal and External Sorting, Bubble sort, Insertion Sort, Selection Sort , Quick Sort, Merge sort

Case Study : Social Network Adjacency Matrix Representing friendship connections among millions of users.

Unit III - Stacks, Queues and Linked Lists (09 Hours)

Stacks: Stack operations, Multiple Stacks, Applications of Stack for Expression Conversion [infix, prefix and postfix], Postfix expression evaluation

Queues: Queue Operations, Circular Queue, Priority Queue and its advantages and applications

Linked list: Introduction of Linked Lists, Primitive Operations on Linked List- Create, Traverse, Search, Insert, Delete, Sort, and Concatenate. Types of Linked List: Singly linked, linear and Circular Linked Lists, Doubly Linked List,

Case study: Implementation of Stack and Queue operations using Linked lists

Unit IV - Hashing (09 Hours)

Hash Table : Concepts-hash table, hash function, basic operations, bucket, collision, probe, synonym, overflow, open hashing, closed hashing, perfect hash function, load density, full table, load factor, rehashing, properties of good hash function, Collision resolution strategies- open addressing and chaining, Hash table overflow- open addressing and chaining, extendible hashing, closed addressing and separate chaining

Case study : Dictionary Application using Hash Tables, Description: Implement a dictionary where words and meanings are stored and retrieved using hashing with collision resolution

Unit V - Graphs and Trees (09 Hours)

Graphs: Basic Concepts, Storage representation, Adjacency matrix, adjacency list, Traversals-depth first and breadth first, Minimum spanning Tree, Greedy algorithms for computing minimum spanning tree- Prims and Kruskal Algorithms

Trees: General tree and its representation: sequential and linked organization, Binary tree- properties, converting tree to binary tree, binary tree traversals (recursive and non-recursive) - inorder, preorder, post order, Operations on binary tree. Binary Search Tree (BST) and its operation

Case study: GPS/Navigation system that models a city map as a weighted graph and applies core graph algorithms ZIP/GZIP file compression using frequency-based encoding. using Huffman tree

Learning Resources

Text Books:

1. Data structures and algorithms in python by Michael T. Goodrich, ISBN-13: 978- 1118290279, ISBN-10: 1118290275, Publisher: Wiley; 1st edition (March 18, 2013).

2. Problem Solving with Algorithms and Data Structures Using Python by Bradley N Miller and David L. Ranum. ISBN-13: 978-1590282571, ISBN-10: 1590282574, Publisher: Franklin, Beedle & Associates; 2nd edition (August 22, 2011).

Reference Books:

1. Hands-On Data Structures and Algorithms with Python: Write complex and powerful code using the latest features of Python 3.7, 2nd Edition by Dr. Basant Agarwal, Benjamin Baka. ISBN: 9781788991933, 2018.

2. Core Python Programming -R. Nageswara Rao, ISBN-10: 9789351199427, ISBN-13: 978-9351199427, Willy; 1st edition (January 1, 2016).

MOOC / NPTEL/YouTube Links: -

1. Programming, Data Structures and Algorithms using Python By Prof. Madhavan Mukund, Chennai Mathematical Institute, https://archive.nptel.ac.in/courses/106/106/106106145/

YouTube/Video Links:

1. https://nptel.ac.in/courses/106102064
2. https://onlinecourses.swayam2.ac.in/cec19_cs04/preview