

COURSE STRUCTURE

Course Code	CSE00030				
Course Category	Program Core				
Course Title	Algorithms and Data Structures Concepts				
Teaching Scheme	Lectures	Tutorials	Laboratory / Practical	Project	Total
Weekly load hours	3	-	4	-	7
Credits	3	-	2	-	5
Assessment Schema Code	TL9				

Prerequisites:

- Programming in C

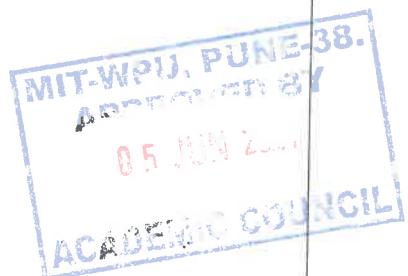
Course Objectives:

1. To gain the knowledge of Various types of Data Structure
2. To introduce various techniques for representation of the data in the real world.
3. To choose appropriate data structure as applied to specified problem definition.
4. To design various applications based on different data structures.
5. Use of programming language for application development.

Course Outcomes:

After completion of this course students will be able to:

1. Understand various categories of data structures
2. Compare and analyse different searching and sorting algorithms
3. Implement various categories of Linked List.
4. Implement the concept of stack and queue and develop applications using it.
5. Implement the concept of tree and graph and its applications.



Course Contents:

Unit 1: Introduction to Data Structure

Concept and need of Data Structure, Abstract Data Type, Classification of data Structure, Operations on data structures, Algorithm Different Approaches to design an algorithm (i)Top Down(ii)Bottom Up

Dean
Faculty of Engineering and Technology



, Control Structures used in Algorithm, Analysis of algorithms, Time and Space Complexity-Worst Case, Average Case, Best Case and Amortized , Asymptotic Notations-Big O Notation(O),Omega Notation(Ω),Theta Notation(θ)

Unit 2: Searching & Sorting

Introduction to searching, Searching Techniques (Algorithm & Examples)-Linear Search, Binary Search

Introduction to sorting, Sorting Techniques (Algorithm & Examples)-Bubble Sort, Insertion Sort, Selection Sort, Merge Sort, Radix Sort, Heap Sort, Shell Sort, Comparison of Sorting Algorithm

Unit 3: Linked List

Introduction-Basic Terminologies-Node, Null Pointer, Address, Empty List etc., Linked List Versus Arrays, Operations on Linked List- Creation, Insertion, Deletion, Traversing, Searching

Types of Linked List-Singly Linked List-Traversing a Linked List, searching for a value, Inserting a new Node, Deleting a node, Double Linked List-Inserting a new Node, Deleting a node, Circular Linked List (only concept)

Unit 4: Stacks & Queue

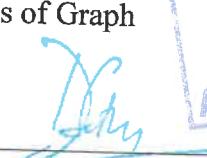
Introduction to stack-Representation of stacks in memory using array, Primitive Operations of stack, Stack Overflow, Stack underflow, Application of stacks-Recursion, Polish Notations-Introduction, Conversion of Infix Expression into postfix Expression (Algorithm & Examples), Conversion of Infix Expression into prefix Expression (Algorithm & Examples), Evaluation of prefix expression (Algorithm & Examples), Evaluation of Postfix Expression (Algorithm & Examples)

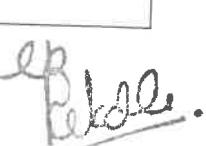
Introduction to Queues-Representation of Queue in memory using Array, Primitive Operations on Queue, Types of Queues-Linear Queue, Circular Queue, Priority Queue, Applications of Queue

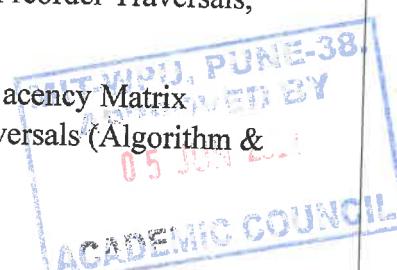
Unit 5: Tree & Graph

Introduction to Tree-Basic Terminology-Tree, Degree of a node, Degree of a tree, Root, Sub-trees, Leaf node, Path, Ancestor Node, Descendent Node, Level of a node, Depth/Height of tree, In-degree, Out-Degree. Types of Trees (Definition and example)-General Trees, Forest, Binary Trees, Binary Search Trees, Expression Trees. Tree Traversals (Algorithm & Examples)-Preorder Traversals, Inorder Traversals, Postorder Traversal, Applications of tree

Introduction to Graph-Graph Terminologies- Representation of a graph-Adjacency Matrix representation of graph, Adjacency List representation of graph. Graph Traversals (Algorithm & examples)-Breadth First Search, Depth First Search, Applications of Graph


Dean
Faculty of Engineering and Technology


ACADEMIC COUNCIL



Laboratory Exercises / Practical: Detailed list of experiment. 1 to 12 (Every experiment consist of more than 3 subprograms)

1. Write a program to implement arrays operations:

- i. Find an average of 10 numbers using an array.
- ii. Display the following pattern.

```
*  
##  
***  
####
```

- iii. Find First Repeating Element
- iv. Find the greatest and the smallest element in the array
- v. Squaring the odd positioned elements.

2. Write 'C' program to Search Particular Data from given array using:

- i. Search data using Linear search: Consider the following list to perform Linear search (56, 36, 89, 57, 01, 00, 67, 59)
 - 1. Search the item 01 from the above list and write the item is found or not with procedure.
 - 2. Search the item 55 from the above list and write the item is found or not with procedure.
- ii. Search data using Binary search.
- iii. Compare Linear search vs Binary search.
- iv. State Limitations of Linear search in terms of Time Complexity.

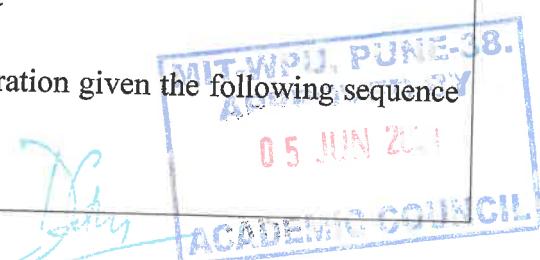
3. Write 'C' program to sort data from given array using – Bubble sort & Selection Sort:

- i. Sort elements in ascending order using Bubble Sort.
- ii. Sort elements in descending order using Selection Sort.
- iii. Find the number of comparisons required in bubble sort method of the following list having 5 numbers: 100, 200, 300, 400, 500
- iv. Sort the given array in ascending order using Selection sort method and show diagrammatic representation of every iteration of for loop: 500, -20, 30, 14, 50

4. Write 'C' program to sort data from given array using – Insertion sort & Radix Sort:

- i. Sort element in ascending order using Insertion Sort
- ii. Sort element in ascending order using Radix Sort
- iii. What is the output of insertion sort after the 2nd iteration given the following sequence of numbers: 7, 3, 5, 1, 9, 8, 4, 6

5. Singly linked List



Dean
 Faculty of Engineering and Technology

*EP
Patil*



- i. Write a menu driven C program that implements singly linked list for the following operations: Create, Insert node at beginning of a, Insert node at middle, Insert node at end
- ii. Write a menu driven C program that implements singly linked list for the following operations: Delete First Node, Delete node at middle, Delete node at end
- iii. Write a menu driven C program that implements singly linked list for the following operations: Create, Display, Concatenate, Count No. of nodes, Reverse, Search a node.

6. Doubly Linked List

- i. Write a menu driven C program that implements doubly linked list for the following operations: Create, Display, Insert a node, Delete a node
- ii. Write a menu driven C program that implements doubly linked list for the following operations: Search a node, Count No. of nodes, Reverse a node

7. Write a 'C' program to perform.

- i. Primitive operations on Stack-PUSH, POP, Is Empty, Is full.
- ii. Stack size-8 for push(10), push(20), pop, push(25), push(50), push(70), pop, pop, push(100), pop and draw final output.

8. Write 'C' program for expression conversion using stack:

- i. Convert Infix to Postfix or Prefix Expression
- ii. Convert infix expression into prefix using Stack: Infix Expression:
 $A + (B * C - (D / E ^ F)) * G) * H$
- iii. Write a C program to evaluate Infix or Postfix Expression.
- iv. Evaluate prefix expression using Stack: prefix Expression:
 $+ - * + 1 \ 2 \ / 4 \ 2 \ 1 \ \$ 4 \ 2$

9. Write 'C' program to perform primitive operation on Linear Queue.

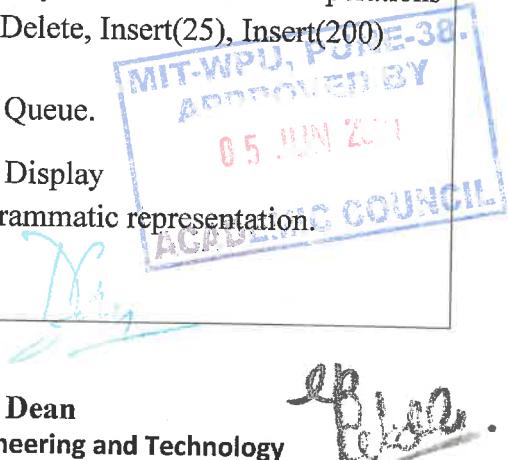
- i. Perform primitive operation on Linear Queue-Insert, Delete, Display
- ii. Perform following operations on Linear queue in array size 10 with this operations
Insert(10), Insert(50), Delete, Insert(100), Insert(20), Delete, Insert(25), Insert(200)

10. Write a 'C' program to perform primitive operations on Circular Queue.

- i. Primitive operation on Circular Queue-Insert, Delete, Display
- ii. What are the advantages of circular queue, show diagrammatic representation.

11. Binary Tree

Dean
Faculty of Engineering and Technology



- i. Write 'C' program to implement following operations on Binary Tree: Creation, Display, Deletion, Search
- ii. Write a menu driven 'C' program to implement Tree Traversal on Binary Tree.

12. Graph

- i. Write 'C' program to implement following operations on Graph: Creation, Display.
- ii. Write 'C' program to represent a graph using following ways
 - o Adjacency Matrix
 - o Adjacency list

Learning Resources:

Text Books/ Reference Books:

- Reema Thareja, "Data Structures using C" - Oxford University, Second Edition, ISBN -0-19-809930-4
- Treamblay, Sorenson, "An introduction to data structures with applications" Tata McGraw Hill, Second Edition, ISBN: 9780070651579
- Horowitz, S. Sahani, S. Anderson-Freed, "Fundamentals of Data Structures in C", Universities Press, ISBN ISBN 0929306406, Publication year 2008
- Yedidyah Langsam, Moshe J Augenstein, Aron M Tenenbaum, "Data Structures using C and C++", Pearson Education, ISBN 81-317-0328-2

Supplementary Reading:

Web Resources:

Weblinks:

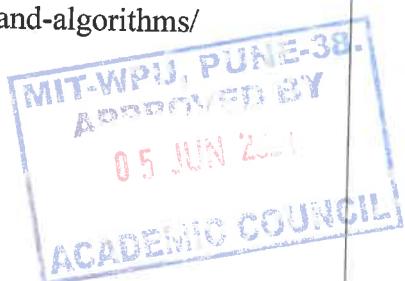
- https://www.tutorialspoint.com/data_structures_algorithms
- <https://www.studytonight.com/data-structures>
- <https://www.khanacademy.org/computing/computer-science/algorithms>

MOOCs: Online courses for self-learning

- <https://www.udemy.com/learning-data-structures-and-algorithms/>
- <https://nptel.ac.in/courses/106103069/>

Pedagogy:

- Group activity
- Power point presentations
- Videos
- Demonstrations
- Systematic use of group work and project-based learning.



M
July

LB
Pekde

Dean
Faculty of Engineering and Technology