PARVATHANENI BRAHMAYYA SIDDHARTHA COLLEGE OF ARTS & SCIENCE
Autonomous
Siddhartha Nagar, Vijayawada–520010
Re-accredited at 'A+' by the NAAC

Course Code	23AIMAL231						
Title of the Course	**Data structures using C**						
Offered to: (Programme/s)	B.Sc. Honours (Artificial Intelligence)						
L	**4**	**T**	**0**	**P**	**0**	**C**	**3**
Year of Introduction:	**2024-25**		Semester:				3
Course Category:	**Major**		**Course Relates to:**		Local, Regional, National, Global		
Year of Revision:	**-**		**Percentage:**		-		
Type of the Course:	SKILL DEVELOPMENT						
Crosscutting Issues of the Course :	-						
Pre-requisites, if any	**Algorithms logic**						

Course Description:

This course covers fundamental concepts in data structures and algorithms. Topics include C program structure, control structures, array data structures, algorithm analysis, sorting algorithms, stacks, queues, dynamic arrays, linked lists, trees, and graphs. Students will learn implementation, properties, functions, and traversal techniques, emphasizing efficiency and practical problem-solving.

Course Aims and Objectives:

S.NO	COURSE OBJECTIVES
1	To choose an appropriate data structure as applied to a specified problem.
2	To use various techniques for representation of the data in the real world.
3	To understand applications using data structures.
4	To develop graph data structures, traversal algorithm.
5	To test the logical ability for solving problems.

Course Outcomes

At the end of the course, the student will be able to…

CO NO	COURSE OUTCOME	BTL	PO	PSO
CO1	Remember C program structure, control structures, and data types.	**K1**	**PO1, PO2, PO4**	**PSO1, PSO2**
CO2	Understand arrays, Big O notation, time and space complexity.	**K2**	**PO1, PO2, PO6**	**PSO1, PSO2**
CO3	Apply sorting algorithms using arrays in practical scenarios.	**K3**	**PO2,PO3, PO4**	**PSO1, PSO2**
CO4	Analyze and compare stacks, queues, dynamic arrays, linked lists.	**K4**	**PO3,PO5, PO6**	**PSO1, PSO2**
CO5	Evaluate tree and graph data structures, traversal algorithm efficiency.	**K5**	**PO4, PO6, PO7**	**PSO1, PSO2**

For BTL: K1: Remember; K2: Understand; K3: Apply; K4: Analyze; K5: Evaluate; K6: Create

CO-PO MATRIX									
CO NO	**PO1**	**PO2**	**PO3**	**PO4**	**PO5**	**PO6**	**PO7**	**PSO1**	**PSO2**
CO1	3	2		1				2	1
CO2	3	2				1		3	2
CO3		3	2	1				3	3
CO4			3		2	1		3	2
CO5				3		2	1	3	3

Use the codes 3,2,1 for High, Moderate and Low correlation Between CO-PO-PSO respectively

Course Structure:

Unit 1: Basic Concepts and Introduction to Data Structures (12hours)

- Introduction to C program Structure
- Control Statements– IF,NESTED-IF,WHILE, DO-WHILE, FOR
- **Array Data Structure**
  - o   Single and Multi-Dimensional Arrays
- **Basics of Algorithm Analysis**
  - o   Big O Notation
  - o   Time and Space Complexity Notation
- **Data Structures Overview**
- **Primitive and Non-Primitive Data Structures**

Examples/Applications/Case Studies:

- Write a sample program using if, while, for, and do-while

Exercises/Projects:

- Write a C program to read an array of 10 integers and count total number of even or odd elements.
- Develop a C program to read two arrays of size 5 and store sum of these arrays into third array.

Specific Resources: (web)

1. GeeksforGeeks - Data Structures (https://www.geeksforgeeks.org/data-structures/)

2. Khan Academy - Algorithms (https://www.khanacademy.org/computing/computer-science/algorithms)

Unit 2: Sorting Algorithms   (12hours)

- Implementation Using Arrays
- Bubble Sort
- Selection Sort
- Insertion Sort
- Quick Sort
- Merge Sort

Examples/Applications/Case Studies:

- Write a C program to implement the Bubble sort.
- Write a C program to implement the Insertion sort.

Exercises/Projects:

- Write a C program to implement the Selection sort .

Specific Resources: (web)

1. Geeks for Geeks - Sorting Algorithms (https://www.geeksforgeeks.org/sorting-algorithms/)

2. Sorting Algorithms ([https:// www.Khanacademy. org/computing/ computerscience/ algorithms #sorting-algorithms](https://www.Khanacademy.org/computing/computerscience/algorithms#sorting-algorithms))

Unit 3: Searching, Stack and Queue Data Structures   (12hours)

- **Searching Techniques**
    - o Linear Search
    - o Binary Search
- **Stack Data Structures**
    - o Properties and Functions
    - o Applications of Stack – Infix-Postfix
- **Queue Data Structures**
    - o Properties and Functions
    - o Circular Queue

Examples/Applications/Case Studies:

1. Write a C program to implement the Binary Search.
2. Write a C program to implement the stack operations.

Exercises/Projects:

1. Write a C program to implement the queue operations

Specific Resources: (web)

1. GeeksforGeeks - Stack Data Structure (https://www.geeksforgeeks.org/stack-data-structure/)

2. GeeksforGeeks - Queue Data Structure (https://www.geeksforgeeks.org/queue-data-structure/)

Unit 4: Dynamic Arrays and Linked Lists   (12hours)

- Introduction to Pointers
- **Dynamic Memory Allocation**
- **Linked Lists**
    - Singly Linked List
    - Doubly Linked List

Examples/Applications/Case Studies:

- Write a C program to implement Single Linked List.

Exercises/Projects:

- Write a C program to implement the Double Linked List.

Specific Resources: (web)

1. GeeksforGeeks - Linked List Data Structure (https://www.geeksforgeeks.org/data-structures/linked-list/)

2. Tutorialspoint - Dynamic Memory Allocation in C (https://www. tutorialspoint.com /cprogramming/c_dynamic_memory_allocation.htm)

Unit 5: Trees and Graphs   (12hours)

- **Binary Trees**
    - Arrays and Linked Representation
    - Tree Traversals: Preorder, Inorder, Postorder
- Binary Search Tree and Operations: traversing insertion and deletion.
- **Graphs**
    - Directed and Undirected Graphs
    - Adjacency Lists and Matrices
    - Path and Graph Traversals: Breadth-First Search (BFS) and Depth-First Search (DFS)

Examples/Applications/Case Studies:

- Write a C program to implement the Binary Search Tree operations.

Exercises/Projects:

- Write a C program to implement the BFS and DFS Graph traversals.

Specific Resources: (web)

1. GeeksforGeeks - Tree Data Structure (https://www.geeksforgeeks.org/binary-tree-data-structure/)

2. GeeksforGeeks - Graph Data Structure (https://www.geeksforgeeks.org/graph-data-structure-and-algorithms/)

Text Books:

1. "Data Structures and Algorithms in Java" by Michael T. Goodrich, Roberto Tamassia, Michael H. Goldwasser , Third edition.
2. "Data Structures Using C and C++" by Yedidyah Langsam, Moshe Augenstein, Aaron M. Tenenbaum.
3. Reema Thareja, Data Structures Using C, Oxford University Press Publishers,2023.

Course Code & Title of the Course:	23AIMAL231 **Data Structures using C**
Offered to:	B.Sc. Honours (Artificial Intelligence)
Category:	**SEMESTER: 3**
Max. Marks	**70**
Max.Time	**3 Hrs**

Section A: Short Answer Questions (20 Marks)

Answer All questions. Each question carries 4 Marks.

Q1 (a)Write a C program to find the roots of a Quadratic Equation for all cases?   (K3)

OR

(b)   What is the difference between static and dynamic memory allocation? (K1)

Q2   (a)Write the passes to perform Selection Sort algorithm for 23,12,24,56,2,15. (K3)

OR

(b)   Write an algorithm to perform Bubble Sort. (K2)

Q3   (a)Write the steps how data is stored in a Queue Data structure 21,33,12,4,56.? (K3)

OR

(b)   Explain the push and pop operations in a stack data structure.(K2)

Q4   (a)What are the advantages of using a doubly linked list over a singly linked list? (K2)

OR

(b)   How do insertion and deletion operations differ in linked lists?(K1)

Q5   (a)   Apply preorder traversal to visit nodes of a binary tree.(K4)

OR

(b)   Analyze the impact of node deletion on a binary search tree's structure.(K4)

Section B: Long Answer Questions (50 Marks)
Answer All questions. Each question carries 10 Marks.

Q6   (a)Define primitive data structures and non-primitive data structures. Give Examples. (K2)

OR

(b)   Write a C program that to find the given number is prime or not.(K2)

Q7   (a)   Describe the Quick Sort algorithm, including how the pivot element is chosen and how partitioning works.   (K3)

OR

(b)   Implement Merge Sort for given numbers 23,13,56,32,11,22,57. (K3)

Q8   (a)   Define Queue Data Structure. Write the functions for enqueue and dequeue . (K2)

OR

(b)   Given an infix expression, convert it to postfix notation using a stack. Explain the conversion process.(K2)

Q9   (a)   Implement a C function to insert a node at the end of a singly linked list. (K4)

OR

(b)   Compare the advantages of doubly linked lists over singly linked list. (K4)

Q10   (a)   Analyze the results of different tree traversal methods (preorder, inorder, postorder) on a given binary tree. (K5)

OR

(b)   Evaluate the strengths and weaknesses of BFS and DFS in terms of space complexity.(K5)