Course Code: 25MC9104 **R25**

SIDDHARTH INSTITUTE OF ENGINEERING & TECHNOLOGY::PUTTUR
(AUTONOMOUS)
Siddharth Nagar, Narayanavanam Road – 517583

QUESTION BANK (DESCRIPTIVE)

Subject with Code: Data Structures using C(25MC9104) **Course & Branch**: MCA

Year & Sem: I-MCA & I-Sem **Regulation:** R25

UNIT –I
INTRODUCTION TO C LANGUAGE

1	**a)**	List the basic elements of the C programming language with examples.	**[L1][CO1]**	**[6M]**
	b)	Describe how variables are used in C programs and explain their scope and lifetime.	**[L2][CO1]**	**[6M]**
2		Discuss all basic data types in C with examples, sizes, and range of values.	**[L2][CO1]**	**[12M]**
3	**a)**	Explain the role of operators in C. Discuss arithmetic, relational operators with example.	**[L2][CO1]**	**[6M]**
	b)	Compare and contrast logical operators with bitwise operators in C.	**[L4][CO1]**	**[6M]**
4	**a)**	Write short notes on if-else statement with a simple example.	**[L2][CO1]**	**[6M]**
	b)	Compare the use of if-else-if ladder and switch-case statements.	**[L4][CO1]**	**[6M]**
5	**a)**	Discuss the loop control structures in C.	**[L2][CO1]**	**[6M]**
	b)	Write a C program to reverse a given number using a while loop.	**[L3][CO1]**	**[6M]**
6		Why do we use function prototypes in C? Demonstrate with an example.	**[L4][CO2]**	**[12M]**
7	**a)**	Explain the fundamental concept of pointers in C programming	**[L2][CO2]**	**[6M]**
	b)	Define storage classes in C. Write short notes on `auto` and `static` storage classes.	**[L2][CO2]**	**[6M]**
8	**a)**	What is a string ?How are strings represented in C?	**[L4][CO2]**	**[6M]**
	b)	List and explain the standard library functions used for string manipulation in C.	**[L1][CO2]**	**[6M]**
9	**a)**	Describe how structures are declared, initialized, and accessed in C.	**[L2][CO2]**	**[6M]**
	b)	Write a program to find the student with the highest marks using an array of structures.	**[L1][CO2]**	**[6M]**
10	**a)**	How are arrays declared and initialized in C?	**[L3][CO2]**	**[6M]**
	b)	How can a C program accept input from the command line? Illustrate with a program.	**[L3][CO2]**	**[6M]**

Course Code: 25MC9104 **R25**

UNIT –II

DATA STRUCTURES, STACKS AND QUEUES

1	**a)**	Define a data structure. Differentiate between linear and non-linear data structures.	**[L1][CO3]**	**[6M]**
	b)	Explain the concept of a stack and list its real-life applications.	**[L2][CO3]**	**[6M]**
2	**a)**	Write an algorithm to implement a stack using arrays.	**[L4][CO3]**	**[6M]**
	b)	Develop a C Program to implement Stack using array.	**[L6][CO3]**	**[6M]**
3	**a)**	What are the fundamental operations of a stack? Illustrate with examples.	**[L1][CO3]**	**[6M]**
	b)	Explain the Array representation of a stack with diagram.	**[L2][CO3]**	**[6M]**
4	**a)**	Illustrate the linked list representation of a queue with diagram.	**[L1][CO3]**	**[6M]**
	b)	Write an algorithm to implement a queue using linked list.	**[L4][CO3]**	**[6M]**
5	**a)**	Write short notes on infix, prefix, and postfix notations.	**[L3][CO3]**	**[6M]**
	b)	Convert the infix expression (A + B) * (C - D) to postfix forms.	**[L1][CO3]**	**[6M]**
6		How can stacks be used to compute the value of a postfix expression? Illustrate with an example.	**[L2][CO3]**	**[12M]**
7	**a)**	Explain the operations of a queue with suitable examples.	**[L2][CO3]**	**[6M]**
	b)	Write C programs that implement Queue using pointers.	**[L4][CO3]**	**[6M]**
8		Write an essay on the conversion of an infix expression to postfix notation. Explain step by step with an example.	**[L3][CO3]**	**[12M]**
9	**a)**	Given the infix expression A + B * C, what is the equivalent postfix expression?	**[L2][CO3]**	**[6M]**
	b)	Convert the prefix expression + A * B C to its equivalent infix expression.	**[L2][CO3]**	**[6M]**
10	**a)**	Explain how searching is performed in a linear queue. Write the algorithm.	**[L2][CO3]**	**[6M]**
	b)	write a C program to perform a searching operation on a queue using an array.	**[L3][CO3]**	**[6M]**

Course Code: 25MC9104 **R25**

UNIT –III

LINKED LISTS

1	**a)**	Describe the structure of a node in a singly linked list	**[L1][CO3]**	**[6M]**
	b)	Discuss the advantages and disadvantages of using linked lists over arrays.	**[L2][CO3]**	**[6M]**
2		Explain the process of inserting a node at the beginning, at the end, and at a specific location in a singly linked list, with example diagrams.	**[L2][CO3]**	**[12M]**
3	**a)**	Discuss deletion operations in a singly linked list. What are the challenges in deleting the last node?	**[L2][CO3]**	**[6M]**
	b)	How is an element searched in a singly linked list? Compare the efficiency with array-based linear search.	**[L2][CO3]**	**[6M]**
4	**a)**	Discuss the implementation of a stack using a singly linked list.	**[L2][CO3]**	**[6M]**
	b)	Describe the implementation of a queue using a linked list with examples.	**[L2][CO3]**	**[6M]**
5	**a)**	Explain how polynomials can be represented using singly linked lists.	**[L2][CO3]**	**[6M]**
	b)	Illustrate the procedure for adding polynomials represented with node-based structures.	**[L3][CO3]**	**[6M]**
6	**a)**	Explain the structure and properties of a circularly linked list.	**[L2][CO3]**	**[6M]**
	b)	Outline the steps to perform insertion in a circular linked list at the front, rear, and a given position.	**[L2][CO3]**	**[6M]**
7	**a)**	Discuss insertion operations in a doubly linked list with suitable examples.	**[L2][CO3]**	**[6M]**
	b)	Illustrate the steps for removing nodes from a doubly linked list at various positions with examples.	**[L3][CO3]**	**[6M]**
8		Explain the process of traversing and inserting nodes in a circular linked list.	**[L2][CO3]**	**[12M]**
9	**a)**	Describe the concept of circular linked lists and how they differ from singly linked lists.	**[L2][CO3]**	**[6M]**
	b)	Discuss the applications and advantages of circular linked lists in real-time systems	**[L2][CO3]**	**[6M]**
10		Explain the concept of circular doubly linked lists and describe their practical applications.	**[L2][CO3]**	**[12M]**

Course Code: 25MC9104 **R25**

UNIT –IV

TREES

1	**a)**	Define the following tree terms with examples: root, leaf, parent, child.	**[L1][CO4]**	**[6M]**
	b)	Describe sequential and linked representation of a tree.	**[L2][CO4]**	**[6M]**
2	**a)**	Illustrate the definition and properties of a binary tree with an example diagram.	**[L3][CO4]**	**[6M]**
	b)	Draw a binary tree for the following preorder and inorder sequences: Preorder: A B D E C F Inorder: D B E A C F	**[L1][CO4]**	**[6M]**
3	**a)**	Explain the different types of binary trees with example.	**[L2][CO4]**	**[6M]**
	b)	Discuss the array representation and linked-list representation of binary trees.	**[L2][CO4]**	**[6M]**
4		Write an essay on binary tree traversals: preorder, inorder, and postorder, with examples.	**[L3][CO4]**	**[12M]**
5		Explain the different operations that can be performed on a binary tree and how each operation works.	**[L2][CO4]**	**[6M]**
6		Discuss the Breadth-First Search algorithm with an example and its applications.	**[L2][CO4]**	**[12M]**
7		Explain the Depth-First Search algorithm with an example and its applications.	**[L2][CO4]**	**[12M]**
8	**a)**	Differentiate between adjacency matrix and adjacency list representation of graphs.	**[L2][CO4]**	**[6M]**
	b)	Explain the difference between directed and undirected graphs with examples.	**[L2][CO4]**	**[6M]**
9	**a)**	What is a connected component in a graph?	**[L1][CO4]**	**[6M]**
	b)	Illustrate the idea of connected components in a graph, including examples and diagrams.	**[L3][CO4]**	**[6M]**
10	**a)**	What is a spanning tree of a graph?	**[L1][CO4]**	**[4M]**
	b)	Describe the process of constructing a spanning tree	**[L2][CO4]**	**[8M]**

Course Code: 25MC9104 **R25**

UNIT –V

SEARCHING AND SORTING

1		Explain the concept of searching in data structures and its importance.	**[L2][CO5]**	**[12M]**
2	**a)**	Describe the step-by-step procedure of linear search algorithm.	**[L2][CO5]**	**[6M]**
	b)	Explain with example how linear search algorithm searches for an element in an array.	**[L2][CO5]**	**[6M]**
3		Illustrate how binary search works on sorted arrays with a step-by-step example.	**[L3][CO5]**	**[12M]**
4	**a)**	Explain the bubble sort algorithm with a step-by-step example.	**[L2][CO5]**	**[6M]**
	b)	Write a C program that implements the bubble sorting methods to sort a given list of integers in ascending order.	**[L4][CO5]**	**[6M]**
5		Trace the execution of selection sort on a given array of integers.	**[L3][CO5]**	**[12M]**
6	**a)**	Show step-by-step how insertion sort arranges an array into order.	**[L1][CO5]**	**[6M]**
	b)	Write a C program that implements the insertion sorting.	**[L4][CO5]**	**[6M]**
7		Describe the quick sort algorithm using the divide-and-conquer approach with an example.	**[L2][CO5]**	**[12M]**
8	**a)**	Explain the merge sort algorithm with a step-by-step example.	**[L1][CO5]**	**[6M]**
	b)	Write a C program that implements the merge sorting method.	**[L4][CO5]**	**[6M]**
9	**a)**	Explain the heap sort algorithm and the concept of a heap.	**[L1][CO5]**	**[6M]**
	b)	Write a C program that implements the heap sorting methods to sort a given list of integers in ascending order.	**[L4][CO5]**	**[6M]**
10		Write C programs that use recursive function to perform the Linear search searching operations.	**[L4][CO5]**	**[12M]**

Prepared by:

A Radha
Assistant Professor
MCA