U.S.N.										

B.M.S. College of Engineering, Bengaluru-560019

Autonomous Institute Affiliated to VTU

April 2024 Semester End Main Examinations

Programme: B.E. **Semester: III**
Branch: Computer Science and Engineering **Duration: 3 hrs.**
Course Code: 23CS3PCDST/19CS3PCDST **Max Marks: 100**
Course: Data Structures using C

Instructions: 1. Answer any FIVE full questions, choosing one full question from each unit.
2. Missing data, if any, may be suitably assumed.

Important Note: Completing your answers, compulsorily draw diagonal cross lines on the remaining blank pages. Revealing of identification, appeal to evaluator will be treated as malpractice.

		UNIT - I	CO	PO	Marks
1	a)	Summarize the fundamental classification of data structures based on the organization of elements and relationships.	CO1	PO1	**05**

	b)	Analyze the given line of code below and show the function call trace when user enters number=488.
		`int Fun(int num)`
		`{`
		`  static int val=0;`
		`  if(num>0)`
		`  {`
		`  val++;`
		`  Fun(num/10);`
		`  }`
		`  else`
		`  {`
		`  return val;`
		`  }`
		`}`
		`int main()`
		`{`
		`  int number;`
		`  int val=0;`
		`  printf("Enter a positive integer number: ");`
		`  scanf("%d",&number);`
		`  val=Fun(number);`
		`  printf("%d is: %d\n",number,val);`
		`  return 0;`
		`}`
			CO2	PO2	**05**

	c)	Convert the following infix expression into postfix expression: **((A + B) * (C – D) + E) / (F + G)** Also, write an algorithm for infix to postfix conversion.	CO2	PO2	**10**
		UNIT – II			
2	a)	Define and elaborate on the fundamental differences between a Linear Queue and a Circular Queue data structure with an example.	CO1	PO1	**05**

		b)	Consider a Circular Queue with a capacity of 5, initially containing the elements 10, 40, and 20. The front (F) is at position 2, and the rear (R) is at position 4. After the insertion of elements 50 and 60, determine the new values of F and R. Subsequently, explain the situation while inserting an element 30. Perform two deletions from the queue and insert the element 100. Show the sequence of steps along with a corresponding diagram, highlighting the values of F and R at each stage.	*CO2*	*PO2*	**05**
		c)	Write a program to implement Ascending Priority Queue. Also, list the applications of Priority Queue.	*CO3*	*PO3*	**10**
			OR			
	3	a)	Demonstrate the process of deleting a node from a singly linked list when the node value is given. Also, write a C function for the same.	*CO3*	*CO3*	**05**

		b)	Analyze the following C function that takes a Singly-Linked List as an input argument. The function pop the element from the list and returns the item. Complete the blank part of the code to perform the above operation.
			
			`typedef struct node`
			`{`
			`  int value;`
			`  struct node _____`
			`}Node;`
			
			`int function(Node *head)`
			`{`
			`  _____ *ptr;`
			`  if(head == NULL)`
			`  {`
			`  printf("\nList is empty");`
			`  }`
			`  else`
			`  {`
			`  _____`
			`  _____`
			`  _____`
			`  return ptr->value;`
			`  }`
			`}`
				CO2	*PO2*	**05**

		c)	A project manager oversees a team with members joining or leaving randomly. The manager needs a flexible system to update the list of team members, allowing additions or removals, along with the ability to print the current count of team members. Develop a C function to simulate this dynamic scenario using a Singly Linked List.	*CO3*	*PO3*	**10**
			UNIT - III			
	4	a)	Demonstrate the functionalities of the stack using a Doubly Linked List.	*CO2*	*PO2*	**10**
		b)	In a music playlist application, the list of songs is managed using a Circular Linked List. Each node in the circular linked list represents a song, containing information such as the song title,	*CO3*	*PO3*	**10**

			artist, and duration. The circular nature of the list ensures seamless playback. Write a C program to add or remove song at specific position in the circular linked list.			
			UNIT - IV			
	5	a)	Given the preorder traversal sequence of a Binary Search Tree: 30, 20, 10, 15, 25, 23, 39, 35, 42 Construct the binary search tree and determine its corresponding postorder traversal. Illustrate the step-by-step process of constructing the binary search tree.	*CO2*	*PO2*	**05**
		b)	Analyze and write a Binary Search Tree for the below given node values: 13, 3, 4, 12, 14, 10, 5, 1, 8, 2, 7, 9, 11, 6, 18 Write a Binary Search Tree after deleting node 8 and 10. Explain the steps clearly.	*CO2*	*PO2*	**05**
		c)	Imagine you are developing a library catalog system where each book is represented as a node in a Binary Search Tree. Each node contains information such as the book id, book title, author, and publication year. Your task is to implement the insertion of new books into the Binary Search Tree based on book id and provide a mechanism to print the preorder traversal of the catalog.	*CO3*	*PO3*	**10**
			OR			
	6	a)	List the applications of Breadth-First Search graph traversal technique. Also, write an algorithm for BFS graph traversal.	*CO1*	*PO1*	**10**
		b)	Write a C function to check whether given graph is connected or not. Also, explain the same with an example.	*CO3*	*PO3*	**10**
			UNIT - V			
	7	a)	Develop a C program that employs a hash function H: K -> L, specifically defined as H(K) = K mod n where 'n' represents size of the hash table. The objective is to implement a hashing technique that maps a given product code K to a corresponding memory address L. In cases where collisions occur, implement a resolution strategy using linear probing.	*CO3*	*PO3*	**10**
		b)	Demonstrate the various address collision resolution methods in hashing technique with advantages and disadvantages in each method.	*CO1*	*PO1*	**10**

B.M.S.C.E. - ODD SEM 2023-24

****