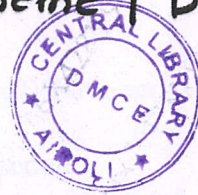


E(COMP.) / Sem-III / R-20 / C Scheme / DS.

(3 Hours)



Total Marks: 80

N.B: (1) Question No. 1 is compulsory

(2) Attempt any three questions out of the remaining five questions

(3) Figures to the right indicate full marks

(4) Make suitable assumptions wherever necessary with proper justifications

- Q1 A Differentiate between arrays and linked lists. [05]  
 B Define data structure and Explain types of data structures. [05]  
 C Write a program in C to reverse a string using stack. [05]  
 D Explain the concept of a queue ADT and describe a real-world application where queues are useful. [05]
- Q2 A Write a program to perform following operations on the Doubly linked list [10]  
 a) Create the list  
 b) Insert element at the beginning  
 c) Insert element at the end  
 d) Delete last element of the list  
 B Explain the process of deleting a node from a Binary Search Tree(BST). Illustrate the cases of deletion with examples. [10]
- Q3 A Write a program in C to convert a given infix expression into a postfix expression. [10]  
 B Explain the concept of Huffman coding and describe the steps for constructing a Huffman tree. Given the following characters with frequencies, construct the Huffman tree and generate the Huffman codes: A:5, B:9, C:12, D:13, E:16 [10]
- Q4 A Construct an AVL tree by inserting the following elements in sequence, showing the tree after each insertion and rotation: 30, 20, 10, 25, 40, 50, 5, 6, 11, 12. [10]  
 B Write a program in C to implement stack ADT using an array. [10]
- Q5 A Explain the key differences between a singly linked list, a doubly linked list, and a circular linked list. Use diagrams to show the structure of each type and discuss the advantages and disadvantages of each. [10]  
 B Write a program in C to implement circular queue using linked list. [10]
- Q6 A Explain Depth First search and Breadth First search graph traversal techniques with example. [10]  
 B A hash table has 10 slots, and the following keys are inserted in this order: 21, 33, 40, 98, 51, 64, 10, 75, 4, 86. Using linear probing as the collision resolution technique, show the final arrangement of keys in the hash table. Use the hash function  $h(x) = x \% 10$ . [10]