Reg. No. : E N G G T R E E . C O M

Question Paper Code : 70055

B.E./B.Tech. DEGREE EXAMINATIONS, NOVEMBER/DECEMBER 2022.

Third Semester

Computer and Communication Engineering

CD 3291 — DATA STRUCTURES AND ALGORITHMS

(Common to Information Technology)

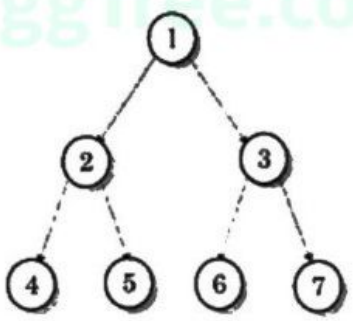
(Regulations 2021)

Time : Three hours Maximum : 100 marks

Answer ALL questions.

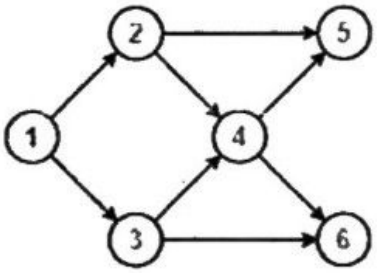
PART A — (10 × 2 = 20 marks)

1. Define abstract data type. What does it focus on?
2. Give the general recurrence for divide and conquer algorithms.
3. Define a circular linked list. What is its disadvantage?
4. Define a linear queue. What is its disadvantage?
5. Differentiate between internal and external sorting.
6. Define hash function. Give two examples.
7. Define a full (perfect) binary tree. Is the tree given below a full binary tree?



8. Construct a max heap with the following data 20, 1, 2, 40, 30.

9. Find the topological ordering of the following graph



10. Define complexity classes P and NP.

PART B — (5 × 13 = 65 marks)

11. (a) (i) What is the purpose of asymptotic notations? Explain the asymptotic notations. (6)

(ii) Consider the following recursive algorithm for computing the sum of the first n cubes. $S(n) = 1^3 + 2^3 + 3^3 + \ldots n^3$.

Algorithm S(n)

//Input : A positive integer n

//Output: The sum of the first n cubes

if n = 1 return 1

else return S(n – 1) + n * n * n

Identify the basic operation in the algorithm. Set up the recurrence relation for the number of times the algorithm's basic operation is executed and solve it using back substitution. (7)

Or

(b) Explain the features and characteristics of Object Oriented Programming with suitable examples.

12. (a) What are the advantages of linked list over arrays? Give the structure of a singly linked list. What is the disadvantage of a singly linked list? How is it overcome? Write an algorithm to reverse a singly linked list.

Or

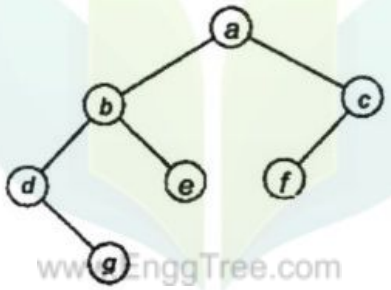
(b) Define a stack. Write an algorithm to convert an infix expression to a postfix expression and apply the same to convert ((A + B) – C * (D/E)) + F to postfix expression.

13. (a) Write an algorithm to perform quicksort and analyse the best case complexity of it. Apply it to sort 24, 56, 47, 35, 10, 90, 82, 31.

Or

(b) What is collision? How is it overcome using separate chaining? Using hash function key mod 7 hash the keys 50, 700, 76, 85, 92, 73, 101. Use separate chaining when collision occurs.

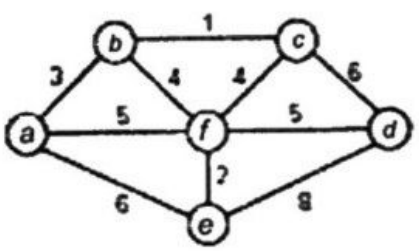
14. (a) How are NULL pointers in a binary tree removed? Write the pseudocode for binary tree traversals. Traverse the binary tree given below using inorder, preorder and postorder traversals and remove NULL pointers from it.



Or

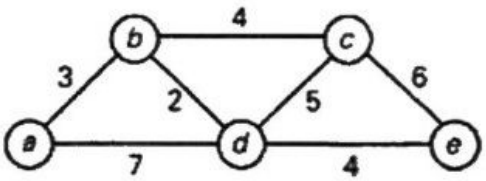
(b) List the properties of a m-way search tree. Construct a 3-way search tree using the keys : 7, 17, 1, 4, 3, 2 10, 15, 19, 21. From the constructed tree delete 19, 1, 4 and 7. Assume deletions are independent of one another.

15. (a) Define minimum spanning tree. Explain kruskals algorithm to find the minimum spanning tree of a graph. Apply the algorithm to find the minimum spanning tree of the given graph.



Or

(b) List the properties of greedy approach. Explain Dijkstra's algorithm to find the shortest path from a source vertex and analyse the time complexity of the approach. Apply the same on the graph given below to find the shortest path from vertex 'a'.



PART C — (1 × 15 = 15 marks)

16. (a) Comment on the statement: The minimum key in any AVL tree with $n > 10$ keys must be found in one of the last two levels. Give an example to prove or disprove. Discuss the generic representations of rotations undertaken on an AVL search tree during insertion operation. Demonstrate rotations when the following elements are inserted into an empty AVL search tree: 5,6,8,3,2,4,7

Or

(b) Consider the graph given below. Write down the adjacency matrix and adjacency lists specifying this graph. (Assume that the matrix rows and columns and vertices in the adjacency lists follow in the alphabetical order of the vertex labels). Write an algorithm for traversing a graph using depth first and breadth first order. Apply graph traversal algorithms to traverse the given graph from vertex 'a', resolving ties by the vertex alphabetical order. What is the complexity of the algorithm when the graph is represented using

(i) adjacency matrix

(ii) adjacency list?