**Reg. No. :** | | | | | | | | | | |

# Question Paper Code: 94826

B.E./B.Tech. DEGREE EXAMINATION, NOV 2024

Fourth Semester

Electrical and Electronics Engineering

19UIT426- Data Structure Using C

(Regulations 2019)

Duration: Three hours Maximum: 100 Marks

Answer All Questions

PART A - (10x 2 = 20 Marks)

| | | |
|---|---|---|
| 1. | What are benefits of ADT? | CO1- U |
| 2. | When doubly linked list can be represented as circular linked list? | CO1- U |
| 3. | Write an algorithm to implement the pop operation under array representation of stacks. | CO2- App |
| 4. | If the elements “A”, “B”, “C” and “D” are placed in a queue and are deleted one at a time, in what order will they be removed? | CO2- App |
| 5. | Define depth and height of a node. | CO1- U |
| 6. | Define internal nodes. | CO1- U |
| 7. | What are the applications of graph data structure? | CO1- U |
| 8. | What is topological sorting in a graph? | CO1- U |
| 9. | Define bubble sort | CO1- U |
| 10. | How the insertion sort is done with the array? | CO1- U |

PART – B (5 x 16= 80Marks)

| | | | | |
|---|---|---|---|---|
| 11. | (a) | Describe the various operations of the list ADT with examples. | CO1-U | (16) |
| | | Or | | |
| | (b) | Describe the steps involved in search operation into a doubly and circular linked list with visualization. | CO1-U | (16) |

12. (a) Explain how to evaluate arithmetic expressions using stacks CO1-U (16)

Or

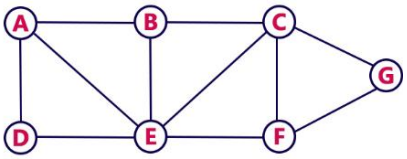
(b) Describe the applications of Stacks CO1-U (16)

13. (a) Construct a Binary Search Tree (BST) for the following sequence of numbers- CO2-App (16)
50, 70, 60, 20, 90, 10, 40, 100

Or

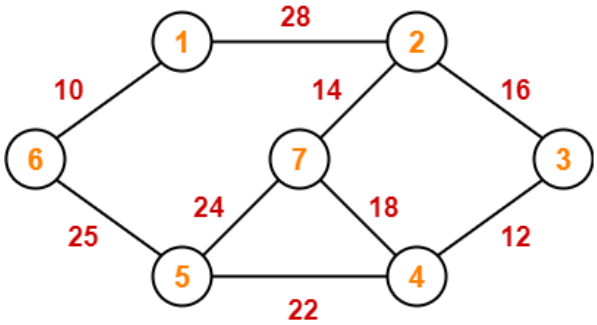
(b) Construct AVL Tree for the following sequence of numbers- CO2-App (16)
50 , 20 , 60 , 10 , 8 , 15 , 32 , 46 , 11 , 48

14. (a) Consider the following example graph to perform BFS traversal. CO2-App (16)




Or

(b) Construct the minimum spanning tree (MST) for the given graph using Kruskal's Algorithm. CO2-App (16)




15. (a) Write an algorithm to implement Bubble sort with suitable example. CO1- U (16)

Or

(b) Write an algorithm to implement insertion sort with suitable example. CO1- U (16)