April 2019

Time – Three hours
(Maximum Marks: 75)

[N.B: (1) Q.No. 8 in PART – A and Q.No. 16 in PART – B are compulsory. Answer any FOUR questions from the remaining in each PART – A and PART – B

(2) Answer division (a) or division (b) of each question in PART – C.

(3) Each question carries 2 marks in PART – A, 3 marks in Part – B and 10 marks in PART – C.]

PART – A

1. Give any two primitive data structures and any two non-primitive data structures.

2. Define traversing on array and strings.

3. Define stack and queue.

4. Give any two applications of stack.

5. What are the two fields in a SLL?

6. Draw a binary tree.

7. Define searching.

8. Define isolated node and loop.

PART – B

9. Explain complexity.

10. Write short notes on priority queue.

11. Write down the steps to delete the last node of SLL.

12. List the differences between linked list and sequential list.

13. Describe any 3 types of graph.

14. What are hash tables?

15. Show a pictorial example of merge sort.

16. Write the recursive algorithm to find the factorial of a number.

[Turn over.....

PART – C

17. (a) (i) Discuss about row major order and column major order.
        (ii) With examples explain any four string functions.

    (Or)

    (b) (i) Explain about pointers and 2D array.
        (ii) Explain top-down approach.

18. (a) (i) Write the algorithm to convert infix expression to postfix expression.
        (ii) Explain implementation of stack using arrays.

    (Or)

    (b) (i) Write algorithm to evaluate postfix expression. illustrate with example.
        (ii) Explain the operations in queues.

19. (a) (i) Describe about doubly linked list.
        (ii) Write down the advantages and disadvantages of linked list.

    (Or)

    (b) (i) What is circular linked list?
        (ii) Explain traversing and searching of a linked list.

20. (a) (i) Define binary tree traversal and explain any one traversal with an example.
        (ii) Explain how to create a binary search tree for a given set of values.

    (Or)

    (b) (i) Explain linear representation of binary tree.
        (ii) Explain adjacency list representation with example.

21. (a) Explain linear search. Write a 'C' program for the same.

    (Or)

    (b) Explain quick sort with example.