

April 2018

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. What is meant by lower bound, upper bound and range of an array?
2. Write any one application of data structure.
3. What are the operations that can be performed on a queue?
4. What are the three fields in DLL?
5. Define path.
6. What is a link?
7. Define sorting.
8. List the different types of hash functions.

PART – B

9. Write short notes on operations on data structures.
10. What is recursion? Write recursive function for factorial.
11. Explain the operations in stack.
12. State the advantages and disadvantages of linked organisation.
13. Define (i) Successor and Predecessor (ii) In degree and Out degree.
14. Explain linked list representation of binary tree.
15. Explain the classification of sorting.
16. Briefly explain linear search.

[Turn over.....

PART - C

17. (a) (i) Explain non-primitive data types.
(ii) Write short notes on different approaches to designing an algorithm.

(Or)

- (b) Give the algorithm for inserting into linear array. Write a 'C' program to insert a new element at index position 'k' in an array.

18. (a) (i) Explain the primitive operations in queue.
(ii) Write algorithm to reverse a list using stack.

(Or)

- (b) Give algorithm to convert infix expression to prefix expression. Explain with an example.

19. (a) How is SLL represented in memory? Give algorithm to insert at the end of the SLL.

(Or)

- (b) How is a free node allocated in memory? Give algorithm to delete intermediate node in SLL.

20. (a) Explain DFS and BFS in graphs.

(Or)

- (b) Explain the two standard methods of storing graphs in computers.

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

(Or)

- (b) Explain the methods of collision resolution.
