



TED (15) – 4133

Reg. No.

(REVISION — 2015)

Signature

**DIPLOMA EXAMINATION IN ENGINEERING/TECHNOLOGY/
MANAGEMENT/COMMERCIAL PRACTICE — APRIL, 2019**

DATA STRUCTURES

[Time : 3 hours

(Maximum marks : 100)

PART — A

(Maximum marks : 10)

Marks

I Answer *all* questions in one or two sentences. Each question carries 2 marks.

1. Define ADT.
2. Convert the expression $(A + B) * C / (D - E)$ in to prefix form.
3. Write memory management operation for nodes in C++.
4. What is a full binary tree ?
5. Define directed graph.

(5×2 = 10)

PART — B

(Maximum marks : 30)

II Answer any *five* of the following questions. Each question carries 6 marks.

1. Explain complexity of algorithms and Big Oh(O) notation.
2. How do you insert and delete elements in a queue ?
3. Describe list using array.
4. What do you mean by traversal of a binary tree ? Write the algorithm for post order traverse of a binary tree.
5. Explain with example Expression trees.
6. Write the algorithm for DFS of a graph with example.
7. Write the algorithm for binary search on a list of sorted elements.

(5×6 = 30)



PART — C

(Maximum marks : 60)

(Answer *one* full question from each unit. Each full question carries 15 marks.)

UNIT — I

- III (a) Explain about Array as an ADT. 8
(b) Explain Template and Classes in C++ 7

OR

- IV (a) Explain prefix, infix and postfix Expressions. 6
(b) Explain the procedure/algorithm for infix to postfix conversion using stack. 9

UNIT — II

- V Explain linked list ADT with makeEmpty(), printList(), find(), findkth(), insert(), delete() 15

OR

- VI (a) Explain the implementation of Stack with linked list. 9
(b) Write short note on circular and doubly linked list. 6

UNIT — III

- VII Explain implementation of binary trees with example. 15

OR

- VIII Explain with example binary search tree ADT and its traverse. 15

UNIT — IV

- IX (a) Explain with example adjacency matrix representation of graph. 8
(b) Write the algorithm for all-pairs shortest path. 7

OR

- X Explain Quick sort algorithm with example. 15
-