

Total No. of Questions—8]

[Total No. of Printed Pages—3

Seat No.	
-------------	--

**[5459]-183**

**S.E. (Computer Engineering) (I Sem.) EXAMINATION, 2018**

**DATA STRUCTURES AND ALGORITHMS**

**(2015 PATTERN)**

**Time : Two Hours**

**Maximum Marks : 50**

**N.B. :—** (i) Attempt Q. No. 1 or Q. No. 2, Q. No. 3 or Q. No. 4,  
Q. No. 5 or Q. No. 6, Q. No. 7 or Q. No. 8.

(ii) Neat diagrams must be drawn wherever necessary.

(iii) Assume suitable data, if necessary.

1. (a) Explain static and dynamic data structures with examples. [4]
- (b) What is recurrence relation ? Explain with example. [2]
- (c) Explain the algorithmic strategy of divide and conquer. Explain its application in binary search. [6]

*Or*

2. (a) Write pseudo C/C++ code for reversing a string and state its time complexity. [4]
- (b) Explain the need for fast transpose of sparse matrix. Comment on its time complexity. [2]
- (c) Explain two-dimensional arrays with row and column major implementation. Explain address calculation in both cases with example. [6]

P.T.O.

3. (a) Represent the following using GLL : [3]

$(p, q(r, s(u, v), w) (x, y))$

- (b) Explain the algorithm for evaluation of a postfix expression with an example. [3]
- (c) Write pseudo C/C++ code to delete a node from a doubly linked list. [6]

*Or*

4. (a) What is backtracking ? Explain the use of stack in backtracking. [4]
- (b) Compare sequential and linked organisation of data. [2]
- (c) Write pseudo C/C++ code to perform addition of two polynomials using arrays. [6]

5. (a) Define the following with example : [6]
- (1) Multi-queue
- (2) Dequeue
- (3) Circular queue.
- (b) Explain circular queue using linked list. Write pseudo C code for enqueue operation. [7]

*Or*

6. (a) Write pseudo C/C++ code to perform insert and delete operation on linear queue. [6]
- (b) Explain priority queue. Write ADT for priority queue and state its applications. [7]

7. (a) What is heap ? Explain heap sort with suitable example. State its complexity. [6]

(b) Sort the following numbers using quick sort : [7]

25, 82, 17, 23, 38, 7, 64, 86, 21

State its time complexity and space complexity.

*Or*

8. (a) Write pseudo C/C++ code to perform shell sort. State its time complexity. [6]

(b) Explain linear search with example. State its time complexity and compare it with binary search. [7]