Total No. of Questions—**8**] [Total No. of Printed Pages—**3**

Seat No.	

[5352]-563

S.E. (Computer Engineering) (I Sem.) EXAMINATION, 2018
DATA STRUCTURE & 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*) Draw neat diagrams wherever necessary.
(*iii*) Assume suitable data, if necessary.

1. (*a*) Define and explain the following terms : [3]
(*i*) Data
(*ii*) Data structure
(*iii*) Algorithm.

(*b*) Give pseudo C/C++ code to reverse the string. [3]

(*c*) Explain the divide and conquer strategy with suitable example. Comment on its time complexity. [6]

Or

2. (*a*) Define and explain the following terms : [4]
(*i*) Sequential organization
(*ii*) Linear data structure
(*iii*) Ordered list
(*iv*) Sparse matrix.

P.T.O.

(*b*) Explain polynomial representation using an array with suitable example. [2]

(*c*) Explain the Asymptotic notation Big O, Omega and Theta with suitable example. [6]

3. (*a*) Write a pseudo C/C++ code to insert node into a singly linked list. [3]

(*b*) Explain Generalised linked list with suitable example. [3]

(*c*) Explain evaluation of postfix expression using stack with suitable example. [6]

Or

4. (*a*) Give pseudo C/C++ code to implement the following operations on linked stack : [4]

(*i*) Create

(*ii*) Push data.

(*b*) Explain the stepwise conversion using stack for the given infix expression to the postfix expression : [2]

A * B + C * D.

(*c*) Write pseudo C/C++ code for polynomial addition using singly linked list. [6]

5. (*a*) Define the following terms with example : [6]

(*i*) Linear queue

(*ii*) Circular queue

(*iii*) Priority queue.

(*b*) Write pseudo C/C++ code to implement priority queue operations. [7]

Or

6. (*a*) Explain linear queue and circular queue with suitable example. Give the advantages of circular queue over linear queue. [6]

(*b*) Write pseudo C/C++ code to implement linked queue. [7]

7. (*a*) Sort the following numbers using insertion sort :

55, 85, 45, 11, 34, 05, 89, 99, 67.

Discuss its time complexity and space complexity. [6]

(*b*) Explain sequential search and binary search with appropriate example. Comment on their data organization, time complexity and space complexity. [7]

Or

8. (*a*) Explain Merge sort using the following example :

18, 13, 12, 22, 15, 24, 10, 16, 19, 14, 30.

Discuss its time and space complexity. [6]

(*b*) Write a pseudo C/C++ code to sort the data using bucket sort in ascending order. [7]