PVP 14

Code: IT3T2

II B.Tech - I Semester–Regular/Supplementary Examinations
November 2016

CLASSIC DATA STRUCTURES
(INFORMATION TECHNOLOGY)

Duration: 3 hours Max. Marks: 70

PART – A

Answer ***all*** the questions. All questions carry equal marks

11x 2 = 22 M

1.

a) Define Data Structure and List the categories of Data Structures.
b) State the procedure to insert a node at beginning of the Single Linked List.
c) Evaluate the following postfix expression.

6 5 2 3 + 8 * + 3 + *

d) List the applications of a Stack Data structure.
e) State the limitations of linear queue.
f) Define Abstract Data Type.
g) Describe Omega (Ω) notation.
h) State the average and best case time complexities of Merge sort.
i) Draw the diagram to show the arrangement of nodes using Circular Double Linked list.
j) Define Data Abstraction.
k) Differentiate Binary Tree and BST.

PART – B

Answer any ***THREE*** questions. All questions carry equal marks.
3 x 16 = 48 M

2.

a) Write an algorithm to search elements using Binary Search technique. Explain the procedure to calculate binary search time complexity. 8 M

b) Explain the Quick sort technique with an algorithm. 8 M

3.

a) Write a C program to implement the following operations of a Single Linked List. 9 M

i) List Creation ii) Node Insertion

iii) Node Deletion iv) List Traversal

b) Explain the representation of polynomials using a Circular Linked List. 7 M

4.

a) Explain the procedure to convert the given infix expression to postfix expression using a Stack data structure. 8 M

b) Explain about Queue ADT and its operations. 8 M

5.

a) Define a Binary Tree and its properties. Discuss binary tree representations using arrays and linked lists. 7 M

b) Construct the Binary Search Tree using the following elements. Explain the procedure to perform search, insertion and deletion operations on BST. 9 M

13, 3, 4, 12, 14, 10, 5, 1, 8, 2, 7, 9, 11, 6, 18

6.

a) Define a Graph. Explain the various ways to represent graphs. 8 M

b) Write a BFS procedure to visit nodes of the following graph using Queue data structure. 8 M