Total No. of Questions : 5]

SEAT No. :

PC-3245

[Total No. of Pages : 2

[6383]-1002

M.C.A. (Management)

IT - 12 : DATA STRUCTURES AND ALGORITHMS

(2024 Pattern) (Semester - I)

Time : 2½ Hours] *[Max. Marks : 50*

Instructions to the candidates :

1) ***All questions are compulsory.***
2) ***All questions carry equal marks.***

Q1) a) Write an algorithm to generate fibonaci series upto 5 number using array elements. **[6]**

b) Explain various 1D array applications. **[4]**

OR

c) Write an algorithm to insert elements at the last position of the list. **[6]**

d) Explain space matrix with example. **[4]**

Q2) a) Write an algorithm to print sum of alternate node elements in doubly linked list. **[6]**

b) Differentiate between Doubly linked list and Circular linked list. **[4]**

OR

c) Write an algorithm to reverse the singly linked list elements. **[6]**

d) Write an algorithm to delete node from beginning in singly linked list. **[4]**

Q3) a) Write an algorithm to perform all operations of Queue using linked list. (Insert, Delete, Peek and Traverse). **[6]**

b) Explain applications of stack with suitable example. **[4]**

OR

c) Convert the given Infix expression to post fix expression.
A+B* (C – D) / (E – F) using stack **[6]**

d) Explain Advantages of Queue over stack. **[4]**

P.T.O.

Q4) a) Create as AVL tree by inserting the values 48, 72, 36, 8, 79, 22, 84, 66. **[6]**

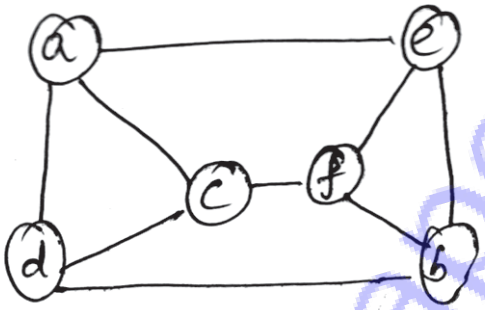
b) Draw a Binary Tree from the given traversal and print its pre-order traversal. **[4]**

Post-order = 4, 5, 2, 6, 3, 1

In-order = 4, 2, 5, 1, 3, 6

OR

c) Apply the BFS Algorithm to traverse the following graph's. **[6]**



d) Construct step-by-step Binary search tree for the following data: 12, 10, 17, 14, 15, 9, 11, 19, 22, 20, 6 **[4]**

Q5) a) Apply Binary search algorithm for the following data:

88, 12, 14, 26, 31, 41, 42, 53, 58, 71 Where search Key = 58 **[6]**

b) Explain Hashing with example. **[4]**

OR

c) Apply Quick Sort Algorithm to sort the given data :

5, 4, 2, 3, 6, 10, 8, 11, 7 **[6]**

d) Explain min heap and max heap with example. **[4]**

▽▽▽▽