No. of Printed Pages : **3**

BICS-007

B.Tech. COMPUTER SCIENCE & ENGINEERING (BTCSVI)

Term-End Examination

December, 2013

BICS-007 : DATA STRUCTURES

Time : 3 hours *Maximum Marks : 70*

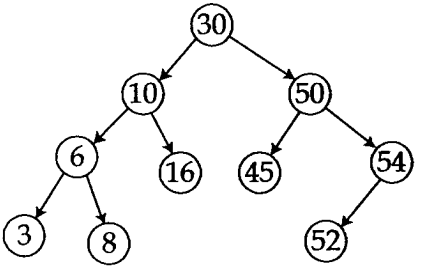
*Note : Attempt **any seven** questions. Assume suitable missing data, if any.*

1. (a) Define sparse matrix. Write a function to find the transpose of a sparse matrix. 5

   (b) Differentiate between static tree table and dynamic tree table. 5

2. (a) Write binary search algorithm and list its advantages and disadvantages over sequential search. 5

   (b) Consider the following binary tree and show what is printed by post, pre and in order traversal of the tree. 5



3. (a) Write any recursive constructive algorithm to find an Euler path in a Eulerian graph. 5

(b) Define algorithm. How do you measure the complexity of an algorithm ? List commonly used asymptotic notations. 5

4. (a) "Hamiltonian cycle is known to be NP-complete". Justify your answer. 5

(b) Write an algorithm to delete an item at position 'p' from a singly linked list. 5

5. (a) Write Heap sort algorithm and also find the average and worst case complexity for it. 5

(b) Why hashing is necessary ? Write down the typical operations of a hash table. 5

6. (a) Differentiate between QUEUE and STACK with suitable examples. 5

(b) Represent a doubly linked list and write its all possible operations and applications. 5

7. (a) Evaluate the following postfix expression using STACK as a underlying data structure. 5

10, 9, +, 4, 9, 3, /, *, 3, +, –

(b) Which of the sorting algorithm has best performance in terms of storage and time complexity ? Justify your answer. 5

8. (a) Consider the following array : 5

30 45 10 20 40 5 15.

Show the content of the array after applying bubble sort (after 1st pass)

(b) Write a function to delete an item from singly linked list. 5

9. (a) Define garbage collection with examples. 5

(b) Differentiate array and linked list. 5

10. Write short notes on **any two** : **5x2=10**

(a) Minimum spanning tree

(b) Applications of Queue

(c) Representation of sets : using bit vector.