Reg. No. : ....

Name : ....

Fourth Semester B.Tech. Degree Examination, May 2013
(2008 Scheme)
08.405 : DATA STRUCTURES AND ALGORITHMS (RF)

Time : 3 Hours

Max. Marks : 100

PART – A

Answer **all** questions

1. Define priority queue. Give any one application.

2. What is meant by structured programming ?

3. What is the use of frequency count in analyzing algorithms ?

4. Give the prefix and postfix representations for the following infix expression :
   $(a + b) * c - d/f + g + h.$

5. Explain how a binary tree is represented using one dimensional array.

6. List any two applications of graph data structure.

7. Write a short note on garbage collection.

8. The Ackerman function, for all non-negative values of m and n is recursively defined as :

$$A(m, n) = \begin{cases} n + 1 & , \text{ if } m = 0 \\ A(m - 1, 1) & , \text{ if } m \neq 0 \text{ but } n = 0 \\ A(m-1, A(m, n-1)) & , \text{ otherwise} \end{cases}$$

   What is the value of A (2, 2) ?

9. What is hashing ?

10. What is the maximum number of edges in a n-node undirected graph without self loop ?

(10×4=40 Marks)

PART – B

Module – I

11. a) Explain the implementation of two stacks using one array, efficiently.

    b) Explain the algorithm for inserting a node into a doubly linked list.

OR

12. a) Explain the representation of polynomial using linked list. Use it for adding two polynomials.

    b) Explain stepwise refinement techniques.

Module – II

13. a) Write a non-recursive algorithm for inorder traversal.

    b) Explain any two methods of representing graphs, with examples.

OR

14. a) Write an algorithm to delete a node form a given binary search tree. Give examples.

    b) Explain storage compaction.

Module – III

15. a) Explain quick sort algorithm. Derive its best case and worst case time complexity.

    b) Explain collision resolution techniques.

OR

16. a) Explain the working of heap sort algorithm for the following input :

    35, 37, 61, 84, 73, 25, 53, 46, 39, 93.

    b) Explain :
       i) insertion sort algorithm and
       ii) binary search algorithm. **(3×20=60 Marks**