Reg. No. : ☐☐☐☐☐☐☐☐☐☐☐☐☐

## Question Paper Code : 91339

B.E./B.Tech. DEGREE EXAMINATION, NOVEMBER/DECEMBER 2014.

Third Semester

Computer Science and Engineering

CS 2201/CS 33/080230007/10144 CS 302 — DATA STRUCTURES

(Regulation 2008/2010)

(Common to 10144 CS 302 – Data Structures for B.E. (Part-Time) Second Semester CSE – Regulation 2010)

Time : Three hours                                              Maximum : 100 marks

Answer ALL questions.

PART A — (10 × 2 = 20 marks)

1.  Write algorithms to insert and delete an element from a linked list.

2.  Convert the following infix expression into postfix : $(A + B)*(C + B)*(E / F)$.

3.  What are the advantages of threaded binary trees over ordinary binary tree?

4.  What is the need for tree representation?

5.  Define the height-balanced tree : "AVL".

6.  What is splay tree?

7.  Define hash function.

8.  Identify the different properties of an equivalence relation in a set.

9.  Write atleast two applications of graphs.

10. What is topological sorting?

PART B — (5 × 16 = 80 marks)

11. (a) Write algorithms to insert and delete elements from a linked list. Consider all cases. (16)

Or

(b) (i) Develop algorithms for inserting and deleting values from a queue. (8)

(ii) Develop algorithms for inserting and deleting values from a stack. (8)

12. (a) (i) Write in detail the various methods in which a binary tree can be represented. Discuss the advantage and disadvantage of each method. (10)

(ii) Write an iterative algorithm to traverse a tree in preorder. (6)

Or

(b) (i) Explain about threaded binary trees in detail. (8)

(ii) Explain by an algorithm to create and delete a particular node from a binary tree. (8)

13. (a) Explain in detail about heap sort algorithm with an example.

Or

(b) (i) Discuss how to insert an element in a AVL tree. (8)

(ii) Discuss briefly the various rotations in a splay tree with an example. (8)

14. (a) Explain about smart union algorithms in detail. (16)

Or

(b) Discuss the disjoint set find with path compression with suitable algorithm. (16)

15. (a) Apply depth first and breadth first search to the complete graph on four vertices. List the vertices in the order they would be visited. (16)

Or

(b) Explain about Prim's and Kruskal's algorithms in detail. (16)