(or)

8. a. Explain binary search with an example. **6M**

   b. Explain various rotations of AVL trees maintaining balance factor while insertion takes place. **9M**

* * *

CS/IT 3004

II/IV B.Tech. DEGREE EXAMINATION, APRIL, 2017

Third Semester

DATA STRUCTURES

Time: 3hours ***Max. Marks: 70***

Part-A is compulsory

Answer One Question from each Unit of Part-B

PART-A

10 x 1 = 10M

a. List the different types of Data structures.

b. What is the advantage of doubly linked list over a singly linked list?

c. Define time complexity.

d. Define complete binary tree.

e. Define the terms: degree of a node and leaf node

f. Differentiate linear from non linear data structure.

g. List the application of queues.

h. Define Warshal's algorithm.

i. What is an AVL tree?

j. What is heap sort?

PART-B

4 x 15 = 60M

UNIT-I

1. a. Define Stack. Write a C program to implement Stack using Linked List. **8M**

   b. Write an algorithm to evaluate Postfix expression. **7M**

(or)

2. a. What is a circular queue? Write an algorithm to implement circular queues using array for enque(), deque() and display() operations. **9M**

   b. Write the pseudo code to insert the element after a position in doubly linked list. **6M**

UNIT-II

3. a. Differentiate between Binary tree and BST with an example. **6M**

   b. Write recursive algorithms for postorder, preorder and inorder traversals of a binary tree. **9M**

(or)

4. a. Explain the Binary Tree Representations. **7M**

   b. Write an algorithm to perform insertion operation in a Binary Search Tree. **8M**

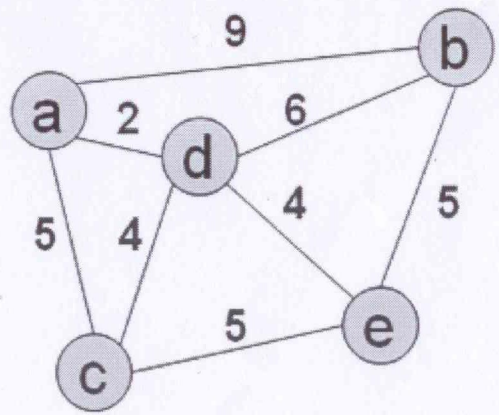
UNIT-III

5. a. Explain Breadth First Search with an example. **6M**

   b. Define Graph and explain how graphs can be represented in adjacency matrix and adjacency list by illustrating with examples. **9M**

(or)

6. a. What is Kruskal's algorithm? Find the minimum spanning tree for the following graph using Kruskal's algorithm. **8M**



   b. Explain Dijkstra's algorithm with an example. **7M**

UNIT-IV

7. a. Write an algorithm for merge sort. **7M**

   b. Trace the merge sort algorithm for the following list of numbers 20, 30, 10, 40, 500, 60, 80, 90, 200, 250 **8M**