VR17 Reg. No: | | | | | | | | | | |

VELAGAPUDI RAMAKRISHNA

SIDDHARTHA ENGINEERING COLLEGE

(AUTONOMOUS)

II/IV B.Tech. DEGREE EXAMINATION, NOVEMBER, 2018

Third Semester

INFORMATION TECHNOLOGY

17IT3303 DATA STRUCTURES

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

Part-A is compulsory

Answer One Question from each Unit of Part - B

Answer to any single question or its part shall be written at one place only

PART-A

10 x 1 = 10M

1. a. Define space complexity.
   b. What is data abstraction?
   c. Define algorithm.
   d. Write the ADT for queue.
   e. Define circular queue.
   f. Define binary search tree.
   g. What do you mean by balance factor in AVL tree?
   h. What is worst case time complexity of heap sort?
   i. List the properties of a good hash function.
   j. Define left skewed tree and give an example.

PART-B

4 x 15 = 60M

UNIT-I

2. a. Write the binary search implementation in C language and also mention the time complexity of it. **8M**

   b. Write the algorithm for infix to postfix conversion. **7M**

(or)

3. a. Write the ADT for stack and write a C program to implement stack operations. **8M**

   b. Discuss about time complexity. **7M**

UNIT-II

4. a. Write a C program to implement doubly linked list with creation and deletion operations. **8M**

   b. Define queue. Discuss various types of queues and the operations that can be performed. **7M**

(or)

5. a. Define linked list. Discuss various types of linked list. **7M**

   b. Write a C program to add two polynomials using linked list. **8M**

UNIT-III

6. a. Write the recursive binary tree traversal algorithms. **9M**

   b. Discuss various ways of representation of trees. **6M**

(or)

7. a. Write an algorithm for creation and search operations of binary search tree. **7M**

   b. Construct an AVL tree for the list : 1, 7, 2, 9, 4, 6, 3, 10, 5, 11, 13, 17, 12. **8M**

UNIT-IV

8. Write the heap sort algorithm to sort a set of integers. **15M**

(or)

9. a. Explain the need of extendible hashing. **6M**

   b. Explain separate chaining collision resolution technique with an example. **9M**

* * *