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

Question Paper Code: 33202

B.E. / B.Tech. DEGREE EXAMINATION, NOV 2018

Third Semester

Computer Science and Engineering

01UCS302 - DATA STRUCTURES

(Regulation 2013)

Duration: Three hours Maximum: 100 Marks

Answer ALL Questions.

PART A - (10 x 2 = 20 Marks)

1. What is abstract data type? Give any two examples.
2. Mention the applications of stack.
3. What are the various tasks performed while traversing a binary tree?
4. What is meant by a thread in a threaded binary tree?
5. What is the need for balancing a tree?
6. Mention the applications of B-tree.
7. What is meant by primary clustering?
8. What is the need for path compression?
9. What is the significance of minimum spanning tree?
10. What is an articulation point in a graph?

PART - B (5 x 16 = 80 Marks)

11. (a) (i) Explain how a stack can be used to evaluate a postfix expression with an implementation in *C* language using an example. (8)

(ii) Write an algorithm to merge two sorted linked lists into a single sorted list. (8)

Or

(b) What do you mean by doubly linked list? Write an algorithm for inserting and deleting an element from doubly linked list. Illustrate with example. (16)

12. (a) In a binary tree,

(i) how do you compute the number of leaf nodes

(ii) how do you swap the left and right children of every node? Explain the algorithms with an example. (16)

Or

(b) Write an algorithm to insert an item into a binary search tree and trace the algorithm with the items : 6, 2, 8, 1, 4, 3, 5. (16)

13. (a) Write a procedure to implement single and double rotations while inserting nodes in an AVL tree with example. (16)

Or

(b) Explain the binary heap operations with examples. (16)

14. (a) Explain the concept of open addressing and rehashing? What is separate chaining? Illustrate the concept with suitable examples. (16)

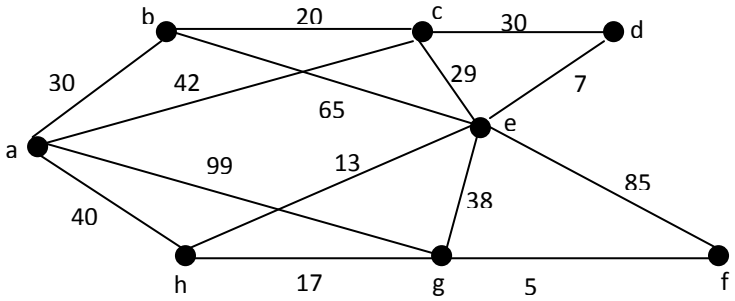
Or

(b) Explain the smart union algorithm with example. (16)

15. (a) Explain with an example for breadth first and depth first search traversal of a graph. (16)

Or

(b) Write an algorithm to find the shortest path using Dijkstra's algorithm. Find the shortest path from 'a' to 'd' in the graph given below.



(16)