MCA / PGDCA 2ND SEMESTER EXAMINATION 2013

PAPER : VI

(DATA STRUCTURE THROUGH 'C' LANGUAGE)

(THEORY)

Time : 2 Hrs. Full Marks : 50

1. Answer the folowing questions : $2\times5 = 10$

   (a) How are two-dimentional arrays strored?

   (b) What do you mean by time complexity of an algorithm?

   (c) How Linked list is different from an array?

   (d) How many nodes can have in a complete binary tree of depth 'd'?

   (e) What is the prefix form of an intix expression A+B-C*D?

2. Answer any three questions from the following $4\times3 = 12$

   (a) What do you mean by traversing an array? Write down the difference btween an array and structure.

   (b) Define a node of a linked list containing an integer and a painter to next node. What is the main advantage of a

circular linked list over a linear linked list?

(c) Differentiate between LIFO and FIFO data structure. Give examples.

(d) Define a 'binary tree', and a 'Complete binary tree'. What are the different tree traversal techniques?

(e) Define the following terms with example: Graph, Degree of a Vertex, Cyclic Graph and Connected Graph.

3. Answer any three from the following questions $6\times3 = 18$

(a) What are the various applications of Stack? Write a 'C' program to implement any one of them.

(b) What is the Binary search technique works. Explain by writing algorithm with an example.

(c) Draw a BST of height '3' with the following elements:
{1,4,5,10,16,17,21}

(d) By using the quick sort algorithm sort the following list of elements:
{13,21,19,67,23,11,12,4}
Clearly mention each step.

(e) The inorder and preorder traversal of a tree 'T' yeilds the following sequences of nodes:
Inorder: 3 8 4 9 5 6 2 7
Preorder: 6 8 3 4 5 9 2 7
Draw the tree 'T'.

4. Answer any one question from the following : $10\times1 = 10$

(a) Write Algorithms for the following single liked list operations:

(i) Creating a list

(ii) Inserting at the begin and at the end position

(iii) Deleting a node from the end of the list.

(b) Write an algorithm for inserting an element into a binary search tree (BST) and deleting an element from a binary search tree.

** ** **