(Following Paper ID and Roll No. to be filled in your Answer Books)	
Paper ID : 214220	**Roll No.** ☐☐☐☐☐☐☐☐☐☐

M.C.A.

Theory Examination (Semester-II) 2015-16

DATA STRUCTURE USING 'C'

Time : 3 Hours   *Max. Marks : 100*

Note: Attempt questions from all Sections as per directions.

Section-A

Attempt *all parts* of this section. Answer in brief. (2×l0=20)

Q1. (a)   What are the data structures used to perform recursion?

(b)   State the advantages of using infix/postfix notations.

(c)   Difference between Abstract Data Type, Data Type and Data Structure.

(d)   What is the need for Priority queue?

(e)   What do you mean by structure property in a heap?

(f) What do you mean by the term "Percolate down"?

(g) What are the objectives of studying data structures?

(h) State the difference between stacks and linked lists

(i) What are the advantages of modularity?

(j) State the difference between primitive and non-primitive data types.

Section-B

Q2. Attempt *any five questions* from this section. (10×5=50)

(a) Evaluate the following prefix expression " ++ 26 + - 1324".

(b) Construct an expression tree for the expression A+(B-C)*D+(E*F)

(c) How many null nodes will a binary tree with 20 nodes have?

(d) Write and explain weighted and unweighted shortest path algorithm.

(e) Explain the minimum spanning tree algorithms with an example.

(2)

(f) Write a C program to perform Merge sort and analyze time complexity of the algorithm.

(g) Write an algorithm INSERT that takes a pointer to a sorted list and a pointer to a node and inserts the node into its correct position in the list.

(h) What do you mean by Base case, Recursive case. Binding Time, Run-Time Stack and Tail Recursion?

Section-C

Attempt *any two questions* from this section.   (15×2=30)

Q3. Write a program to allocate memory dynamically for strings, and store their addresses in array of pointers to strings.

Q4. How memory is freed using Boundary tag method in the context of Dynamic memory management?

Q5. What are expression trees? Represent the following expression using a tree. Comment on the result that you get when this tree is traversed in Preorder, Inorder and postorder (a-b) / ((c*d)+e).