B.Sc.- II (CBCS Pattern) Semester-IV

USCST07 - Computer Science Paper-I - Algorithm & Data Structures

P. Pages : 2 **GUG/W/24/12002**

Time : Three Hours * 4 7 4 1 * Max. Marks : 50

Notes : 1. All questions are compulsory and carry equal marks.
2. Draw neat and labeled diagram wherever necessary.
3. Avoid vague answer and write specific answers related to questions.

Either:

1. a) Define Data Structure. Explain various operations performed on data structure. **5**

b) Write an algorithm for PUSH and POP an element into/from stack. **5**

OR

c) Write an algorithm to arrange all array values in descending order using bubble sort method. **5**

d) Convert following infix expression into postfix expression. **5**

i) $((A-B/C))\wedge(E*(F\wedge G))$

ii) $((P*Q/R)\wedge(S+T/U))\wedge V$

Either:

2. a) Write an algorithm to find factorial of a given number using recursion. **5**

b) Write recursive algorithm to find product of two positive numbers. **5**

OR

c) Write note on- **5**

i) Circular Queue

ii) Priority Queue

d) Write an algorithm of insert an element into queue. **5**

Either:

3. a) Write an algorithm to search an element ITEM into given Linked List. **5**

b) Write an algorithm for inserting element in linked list at first position. **5**

OR

c) Write an algorithm to delete an element from Linked list. **5**

d) Write an algorithm to insert an element ITEM into a sorted linked list. **5**

Either:

4. a) Explain with suitable figure- **5**

i) Complete Binary Tree

ii) Extended Binary Tree

b) Draw a binary tree T which preorder and in order traversal are as below. **5**
Preorder A B C D E F G
In order B A D C F E G

OR

c) Define Binary Tree. Explain the memory representation of binary tree. **5**

d) Write an algorithm to travels a tree using preorder traversal method. **5**

5. **All** questions are compulsory.

a) Write short note on Complexity of Algorithm. **2½**

b) Write short note on DEQUE. **2½**

c) Write a note on Double Linked List. **2½**

d) Explain- **2½**

i) Directed graph ii) Simple graph

****