

B.Tech I Year II Semester (R15) Supplementary Examinations December 2016

**DATA STRUCTURES**

(Common to CSE and IT)

Time: 3 hours

Max. Marks: 70

**PART – A**  
(Compulsory Question)

\*\*\*\*\*

- 1 Answer the following: (10 X 02 = 20 Marks)
  - (a) List out the areas in which data structures applied extensively.
  - (b) Differentiate array and linked list.
  - (c) Consider the following stack of characters, where stack is allocated N = 8 memory cells.  
STACK : A, C, D, F, K, \_, \_, \_.  
Describe the stack as the following operations takes place.
    - (i) POP (STACK, ITEM)
    - (ii) POP (STACK, ITEM)
    - (iii) PUSH (STACK, R)
    - (iv) PUSH (STACK, L)
  - (d) How do you test for an empty queue?
  - (e) There are 8, 15, 13, 14 nodes, were there in 4 different trees. Which of them could have formed a full binary tree?
  - (f) Write the applications of graph data structure.
  - (g) Why is quick sort better than other sorting algorithms?
  - (h) List the properties of heap sort.
  - (i) What is sentinel search?
  - (j) What is clustering in a hashing and list its types?

**PART – B**

(Answer all five units, 5 X 10 = 50 Marks)

**UNIT – I**

- 2 Explain in brief about multi-dimensional array with an example.

**OR**

- 3 Write the various operations of double linked list in detail.

**UNIT – II**

- 4 Explain the various stack operations and illustrate the procedure *Infix To Postfix* with the following arithmetic expression:  $(A + B) \wedge C - (D * E) / F$ .

**OR**

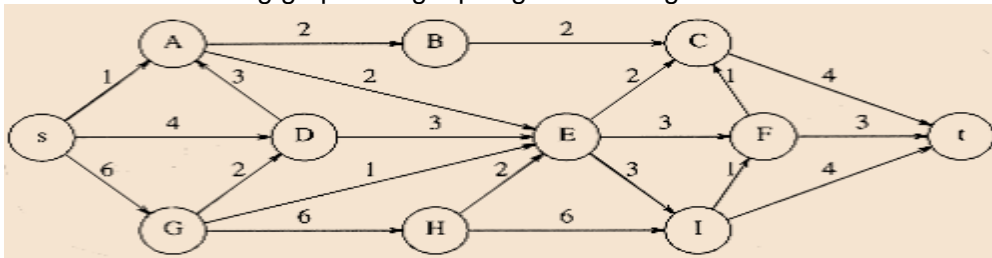
- 5 Write an algorithm and explain the various operations of Circular queue.

**UNIT – III**

- 6 Write a routine to perform a tree traversal with one example.

**OR**

- 7 Simulate the following graph using topological ordering.



**UNIT – IV**

- 8 Explain in brief about Two Way merge sort with an example.

**OR**

- 9 Explain quick sort with an example.

**UNIT – V**

- 10 Briefly explain about probability search and Ordered list search.

**OR**

- 11 Explain linear probing and quadratic probing.

\*\*\*\*\*