

## FACULTY OF INFORMATICS

B.E. 2/4 (IT) I-Semester (Supplementary) Examination, June/July, 2011

## DATA STRUCTURES

Time : Three Hours]

[Maximum Marks : 75

Note :— Answer *all* questions from Part A. Answer any *five* questions from Part B.

## PART—A (Marks : 25)

1. Differentiate between an AVL Tree and a Binary search tree. 2
2. What are the applications of stacks ? 2
3. Convert the following postfix expression to infix and prefix form :  
ab + de/\*. 2
4. Define theta notation. Express  $f(x) = 10x^2 + 4x + 2$  in theta notation. 3
5. Define performance of a program. What are the components of space complexity ? 3
6. What are the advantages of linked list over arrays ? 3
7. Define a leftist tree. 2
8. Define a heap. Create a max heap with the following keys :  
2, 14, 3, 12, 15, 10. 3
9. Explain representation of binary trees using arrays. 3
10. What are advantages of doubly linked list over singly linked list ? 2

## PART—B (Marks : 10×5=50)

11. (a) Write a C++ program for selection sort. 4
- (b) Explain the following :
  - (i) Exceptions. 2
  - (ii) The standard templates library. 4
12. (a) Explain the procedure for converting infix to postfix expression using an example. 6
- (b) Explain the linked representation of queues and explain its operations. 4
13. (a) Define a B-tree. Describe how insertion and deletion operations are carried out in a B-tree. 8
- (b) List down the applications of trees. 2
14. (a) How to represent multiple lists in a single array ? 4
- (b) Write a procedure to insert and delete an element from a single linked list. 6

15. (a) Discuss with examples various graph search methods. 6  
(b) Define a priority queue. List down the applications of priority queues. 4
16. (a) Define a binary search tree. Write a procedure to insert an element into a binary search tree. 6  
(b) Create a binary search tree with the following keys. Perform inorder, postorder and preorder traversals on it and write the results :  
20, 10, 30, 5, 15, 25, 35, 2, 6. 4
17. Write short notes on following :—  
(a) Testing and Debugging. 2  
(b) AVL Tree Rotations. 4  
(c) Asymptotic Notations. 4