Code: DC08 **Subject: DATA STRUCTURES**
Time: 3 Hours **Max. Marks: 100**

JUNE 2011

NOTE: There are 9 Questions in all.

- **Question 1 is compulsory and carries 20 marks. Answer to Q.1 must be written in the space provided for it in the answer book supplied and nowhere else.**
- **The answer sheet for the Q.1 will be collected by the invigilator after 45 Minutes of the commencement of the examination.**
- **Out of the remaining EIGHT Questions answer any FIVE Questions. Each question carries 16 marks.**
- **Any required data not explicitly given, may be suitably assumed and stated.**

Q.1 Choose the correct or the best alternative in the following: (2×10)

a. A list of data items, usually words or bytes, with the accessing restriction that elements can be added or removed at one end of the list only, is known as:

(A) stack **(B)** memory
(C) linked list **(D)** heap

b. A complete binary tree with the property that the value at each node is at least as large as the values at its children is known as

(A) binary search tree **(B)** AVL tree
(C) complete balanced tree **(D)** heap

c. Consider that n elements are to be sorted. What is the worst case time complexity of Shell sort:

(A) O(n) **(B)** $O(n\log_2 n)$
(C) $O(n^{1.2})$ **(D)** O(n X n)

d. Which data structure is needed to convert infix notations to postfix notations:

(A) branch **(B)** queue
(C) tree **(D)** stack

e. Which of the following is a hash function:

(A) quadratic probing **(B)** chaining
(C) open addressing **(D)** folding

f. The prefix expression for the expression a*(b + c)/e-f is :

(A) /*a+bc-ef **(B)** -/*+abcef
(C) -/*a+bcef **(D)** none of these

g. In linked list representation, a node contains at least

(A) node address field, data field
(B) node number, data field
(C) next address field, information field
(D) none of these

h. Adjacency matrix for a digraph is:

(A) unimatrix **(B)** symmetric
(C) asymmetric matrix **(D)** multisymmetric

i. "n" elements of a queue are to be reversed using another queue . The number of "ADD" and "REMOVE" operations required to do so is:

(A) 2*n **(B)** 4*n
(C) n **(D)** no possibility

j. Sparse matrices have

(A) many zeroes entries **(B)** many non numeric entries
(C) higher dimension **(D)** none of these

Answer any FIVE Questions out of EIGHT Questions.
Each question carries 16 marks.

Q.2 a. Write note on the 'Time complexity' of the algorithm. Find out the time complexity of following algorithm 'sum'. **(8)**

```
algorithm  sum(a, n)
    {
    initially sum=0
    for i = 0 to n
    sum=sum+a[i]
    return sum;
    }
```

b. Define Abstract Data Types. By taking any suitable example, make a list of primary operations that may be defined on ADT. **(8)**

Q.3 a. What are the different ways of representing a polynomial using arrays? Write an algorithm to add two polynomials using arrays. **(8)**

b. What is a Linked List? How is it different from array? Write the different types of linked lists. **(8)**

Q.4 a. Implement a Singly Linked List in Stack (LIFO) manner. **(8)**

b. Write an algorithm to interchange the elements at the odd and even positions of an array with n elements. **(8)**

Q.5 a. Discuss Circular linked list with the help of a suitable block diagram. **(8)**

b. Write an algorithm to convert an infix expression to a post fix expression. Execute your algorithm on the following expression: $(A-B)*(D/E)$. Show the position of the stack at all the intermediate stages. **(8)**

Q.6 a. Write down the algorithm for Binary search. Discuss the complexity of it. **(8)**

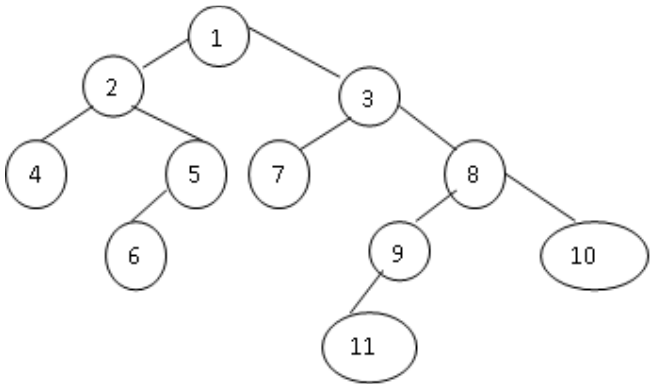
b. Consider the list of six elements as 66,44,2,22,18,16. Apply Selection Sort algorithm to sort this list and show the result of each pass. **(8)**

Q.7 a. Define Merging. Write a recursive algorithm to implement Merge sort. **(8)**

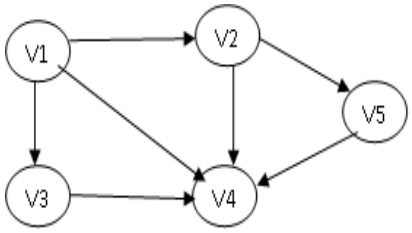
b. Define Hashing. Write Mid-Square method to implement Hashing. **(8)**

Q. 8 a. Explain the Linked representation of Binary Trees by taking any example. **(8)**

b. Find out the Pre-order, In-order and Post-order traversal of the following Tree: **(8)**



Q.9 a. For the following graph, find out the In-degree and Out-degree of all vertices. **(8)**



b. Write Breadth First Search Traversal algorithm for Graph. **(8)**