# DATA STRUCTURE AND BASIC ALGORITHMS
## (CSEN 2004)

**Time Allotted : 3 hrs** **Full Marks : 70**

***Figures out of the right margin indicate full marks.***

***Candidates are required to answer Group A and any 5 (five) from Group B to E, taking at least one from each group.***

***Candidates are required to give answer in their own words as far as practicable.***

## Group – A
## (Multiple Choice Type Questions)

1. Choose the correct alternative for the following: **10 × 1 = 10**

(i) The data structure required to check whether an expression contains a balanced parenthesis is?
(a) Stack (b) Queue (c) Array (d) Tree.

(ii) If a normal queue is implemented using an array having a maximum size of MAX_SIZE, overflow occurs when:
(a) front == MAX_SIZE – 1 (b) rear == MAX_SIZE - 1
(c) (front + 1)%MAX_SIZE == rear (d) (rear + 1)%MAX_SIZE == front.

(iii) What is the value of the postfix expression 6 3 2 4 + – *?
(a) 1 (b) 40 (c) 74 (d) -18.

(iv) Given any 2D array of order r × c, what is the minimum possible order of its triplet array representation?
(a) (r*c - 1) × 3 (b) (r*c + 1) × 3 (c) (r*c) × 3 (d) 1 × 3.

(v) Supposing the values 6, 3, -3, 8, 1, 13, 4, -7 are inserted in the given sequence into an empty binary search tree, what will be the inorder sequence of the final tree?
(a) 13, 8, 6, 4, 3, 1, -7, -3 (b) -3, -7, 1, 3, 4, 6, 8, 13
(c) -3, -7, 6, 3, 8, 1, 13, 4 (d) 6, 3, -3, 8, 1, 13, 4, -7.

(vi) The values in a BST can be sorted in ascending order by using which of the following traversals
(a) Pre-Order traversal (b) Post-order traversal
(c) In-order traversal (d) Level order traversal.

(vii) Which sorting algorithm takes the least time to run if the input array is already sorted in the desired order?
(a) Bubble sort (b) Selection sort
(c) Insertion sort (d) Merge sort.

(viii) What is the worst-case time complexity of searching for an element in a circular linked list?
(a) O(n) (b) O(nlogn) (c) O(1) (d) $O(n^2)$.

(ix) What is the load factor for an open addressing technique?
(a) 1 (b) 0.5 (c) 1.5 (d) 0.

(x) The minimum number of nodes required to connected all vertices of a graph having n nodes is
(a) n – 1 (b) n + 1 (c) $n^2$ (d) 2*n.

## Group - B

2. (a) Define Big-Oh notation. Use it to denote the time complexity of an algorithm whose running time is given as: $f(n) = 5n^3 + 3n^2 + 1$. *[(CO1)(Remember/LOCQ)]*
(b) What is an abstract data type? Explain with an example. *[(CO1)(Remember/LOCQ)]*
(c) Write an algorithm to insert an item into a circular lined list. *[(CO5)(Apply/IOCQ)]*

**(1 + 2) + (1 + 2) + 6 = 12**

3. (a) Compute the triplet array representation of the following 2D matrix:

1 0 0 0 3 0
0 0 8 0 0 0
0 2 7 0 0 0

Is it beneficial to store this matrix as a triplet array? Justify your answer. *[(CO3)(Understand/LOCQ)]*
(b) Explain the difference between best case and worst case running time of an algorithm with a suitable example. *[(CO1)(Remember/LOCQ)]*
(c) Design an algorithm to count the member of elements in a singly linked list. *[(CO6)(Design/HOCQ)]*
(d) Compare and contrast singly linked list and doubly linked list. *[(CO4)(Analyze/IOCQ)]*

**(3 + 1) + 3 + 3 + 2 = 12**

## Group - C

4. (a) Convert the following infix expression into postfix using appropriate data structure:

(a + b)/(c – d*2) + e

Show the contents of your data structure as they get modified. *[(CO5)(Understand/LOCQ)]*
(b) Write an algorithm to push an element into a stack. Your algorithm should handle overflow or underflow errors if applicable. *[(CO4)(Remember/LOCQ)]*
(c) Write a recursive function to compute the nth Fibonacci number. Draw the recursion tree for n = 5. *[(CO4)(Understand/LOCQ)]*

**6 + 3 + 3 = 12**

5. (a) What is tail recursion? How is it different from normal recursion? Explain with suitable example. *[(CO3)(Understand/LOCQ)]*

(b) Write the algorithm to enqueue an element into a circular queue whose maximum capacity is denoted by MAX_SIZE. Your algorithm must guard against any overflow or underflow errors if applicable. *[(CO4)(Remember/LOCQ)]*

(c) What value does the following function return when we pass the number 1 as argument? Give proper explanation.

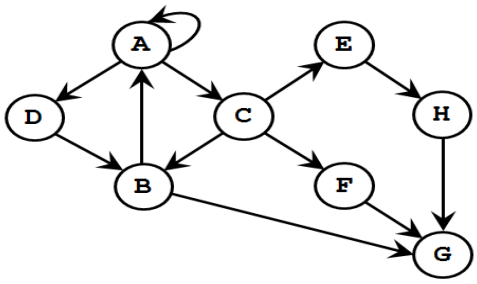
```
int fun (int n)
{
        if (n >100)
                return n;
        else
                return n + fun(n*4);
}
```

*[(CO4)(Analyze/IOCQ)]*

**(2 + 2) + 4 + 4 = 12**

## Group - D

6. (a) Given the following directed graph, compute the in-degree and out-degree of each vertex. Now, perform DFS traversal of the graph to generate a DFS tree, taking A as the starting node. If more than one vertex can be visited at any given time, use alphabetical order to make the choice.




*[(CO2)(Understand/IOCQ)]*

(b) What are the two ways of representing a graph? Use these methods to represent the graph of the previous question. *[(CO2)(Remember/LOCQ)]*

**(2 + 6) + (2 + 2) = 12**

7. (a) Create an AVL tree using the following keys: (Mention different rotation and balance factor in each step)

**H, I, J, B, A, E, C, F, D, G, K, L** *[(CO6)(Design/HOCQ)]*

(b) Design a BST for the following sequence of numbers (Show all intermediate trees).

17, 4, 20, 18, 37, 28, 49, 55, 6, 39

Design the tree if we delete (i) 18 (ii) 4 and (iii) 49 from the created tree one by one. Illustrate each step with suitable explanation. *[(CO2,CO5)(Analyze/IOCQ)]*

**5 + (3 + 1 + 1 + 2) = 12**

## Group - E

8. (a) Write the algorithm to search for a number in a list of integers using binary search. Assume that the list is already sorted. Does your algorithm perform better than linear search? Justify your answer. *[(CO3)(Remember/LOCQ)]*

(b) Perform insertion sort on the given list to arrange the values in ascending order. Show all the different iterations that the algorithm goes through.

17, 25, 30, 37, 34, 20, 12, 28 *[(CO5)(Understand/LOCQ)]*

**(4 + 2) + 6 = 12**

9. (a) Explain the different types of collision resolution techniques in hashing. Using linear probing, insert the keys 24, 49, 78, 76, 45, 3, 74, 33 into a hash table having size 10. *[(CO5)(Remember/LOCQ)]*

(b) Illustrate the selection sort algorithm on input [35, 22, 1, 65, 73, 42]. *[(CO2,CO5)(Analyze/IOCQ)]*

(c) What are the time complexities of performing Quicksort in the best case and average case scenarios? *[(CO2)(Remember/LOCQ)]*

**(3 + 3) + 4 + 2 = 12**

---

| *Cognition Level* | *LOCQ* | *IOCQ* | *HOCQ* |
|---|---|---|---|
| *Percentage distribution* | *59.38* | *32.29* | *8.33* |

**Course Outcome (CO):**

After the completion of the course students will be able to

1. Understand the data structures, their advantages and drawbacks
2. Identify the efficiency aspects of the graph and sorting algorithms covered in this course.
3. Learn about the data structures/methods/algorithms mentioned in the course with a comparative perspective
4. Describe problem statements and to design the solutions using programming language
5. Analyze and apply most appropriate data structure/method/algorithm in a program to enhance the efficiency
6. Develop and efficient program modifying an efficient one using the knowledge gathered from this course.

**LOCQ: Lower Order Cognitive Question; IOCQ: Intermediate Order Cognitive Question; HOCQ: Higher Order Cognitive Question.*