DATA STRUCTURE AND ALGORITHMS
(CSEN 2101)

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.

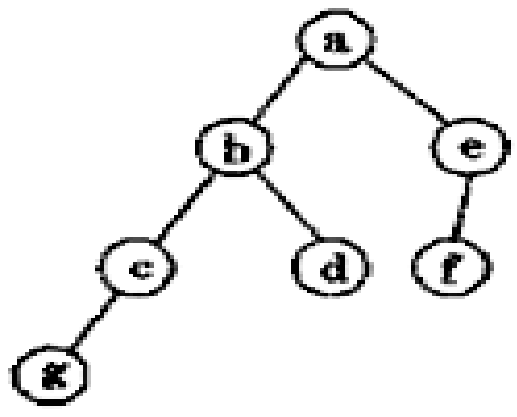
Candidates are required to give answer in their own words as far as practicable.

Group – A
(Multiple Choice Type Questions)

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

(i) A binary tree T has n leaf nodes then the number of nodes having degree 2 is
(a) n + 1 (b) n - 1 (c) 2^n (d) $\log_2 n$

(ii) Consider the balanced binary tree below:



How many nodes will become un-balanced if a node is inserted as a child of the node g?
(a) 1 (b) 3 (c) 7 (d) 8.

(iii) When inorder traversing a tree resulted E A C K F H D B G; the preorder traversal would return
(a) FAEKCDBHG (b) FAEKCDHGB (c) EAFKHDCBG (d) FEAKDCHBG.

(iv) Which of the following is essential for converting an infix expression to the postfix form efficiently?
(a) An operator stack (b) An operand stack
(c) An operator stack and an operand stack (d) A parse tree.

(v) Linked lists are not a suitable data structure for which of the following problems?
(a) Insertion sort (b) Radix sort (c) Binary Search (d) Bubble sort.

(vi) A binary search tree is generated by inserting the following integers: 50, 15, 62, 5, 20, 58, 91, 3, 8, 37, 60, and 24. The number of nodes in the left-subtree and right-subtree of the root respectively is
(a) (4, 7) (b) (7, 4) (c) (8, 3) (d) (3, 8).

(vii) The postfix form of the expression (A+ B)*(C*D- E)*F / G is?
(a) AB+ CD*E - FG /** (b) AB + CD* E - F **G /
(c) AB + CD* E - *F *G / (d) AB + CDE * - * F *G /

(viii) Let G be an undirected graph with n vertices and 25 edges such that each vertex has degree at least 3. What is the maximum possible value of n?
(a) 8 (b) 12 (c) 16 (d) 18.

(ix) $f(n) = 3n^2+10n$; $g(n) = 2n^3$; then
(a) $g(n) \in O(f(n))$ (b) $g(n) \in \Omega(f(n))$ (c) $g(n) \in \Theta(f(n))$ (d) all of these.

(x) A drawback of linear probing method is
(a) primary clustering (b) secondary clustering
(c) folding (d) none of these.

Group- B

2. (a) An unsorted array contains 2n+1 integers such taht all but one element occur twice each. Write a program/pseudo-code/algorithm that can find the element occuring exactly once in this array. What is its time complexity?
[(CO1, CO5)(Understand/HOCQ)]

(b) Write a program or pseudo code or algorithm to display node values in reverse order for a double linked list? [(CO4)(Understand/LOCQ)]

(c) A sorting method with time complexity O(n log n) spends exactly 1 millisecond to sort 1,000 data items. Assuming that time T(n) of sorting n items is directly proportional to n log n, that is, T(n) = cn log n, derive a formula for T(n), given the time T(N) for sorting N items, and estimate how long this method will take to sort 1,000,000 items. [(CO4)(Analyse/HOCQ)]

(4 + 1) + 3 + 4 = 12

3. (a) Can you implement an algorithm, which can perform "delete at front" of a circular linked list in constant time? Justify your answer with proper reasoning.
[(CO2, CO4)(Understand, Analyze/LOCQ)]

(b) Write a pseudo-code/C program to reverse a singly linked list.
[(CO2, CO3)(Apply/LOCQ)]

6 + 6 = 12

Group - C

4. (a) To implement a queue using Stack, show how to implement ENQUEUE and DEQUEUE operations using a sequence of given operations as PUSH and POP for a Stack?
[(CO2,CO6)(Remember/LOCQ)]

(b) Write a function "insert" that is tail recursive. It should take in a list "lst", an "item", and an "index", and insert the item into the list at the given index. Illustrate with an example. [(CO3, CO4)(Analyze/IOCQ)]

(c) Write a program or pseudo code or algorithm that should continue removing adjacent duplicates from the string given a string, till no duplicate is present in the result. For example input string is 'DBAABDAB'. The string left after the removal of all adjacent duplicates is 'AB'. [(CO3, CO5)(Evaluate/HOCQ)]

3 + (3 + 2) + 4 = 12

5. (a) Consider the following code snippet and illustrate in detail how f(5) is calculated?

```
int f(int n)
{
static int r=0;
if(n<=0) return 1;
if(n>3)
{
r=n;
return f(n-2)+2;
}
return f(n-1)+r;
}
```

[(CO4) (Analyze/LOCQ)]

(b) Write a program or pseudo code or algorithm to design a stack that returns a minimum element without using an auxiliary stack. [(CO3)(Analyze/IOCQ)]

(c) Write a program or pseudo code or algorithm to generate binary numbers between 1 to `n` using a queue. (CO4)(Analyze/LOCQ)]

4 + 4 + 4 = 12

Group - D

6. (a) What will be resultant Max-Heap when the elements: 32, 15, 20, 30, 12, 25, 16 are inserted one by one? [(CO2)(Understand/LOCQ)]

(b) Consider the following New-order strategy for traversing a binary tree:

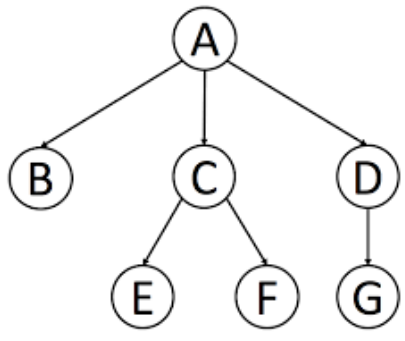
- Visit the root
- Visit the right sub-tree using New-order
- Visit the left sub-tree using New-order

Then what will be the New-order traversal of the expression tree corresponding to the reverse polish expression: 3 4 * 5 – 2 ^ 6 7 * 1 + - . [(CO6)(Evaluate/HOCQ)]

(c) The in-order traversal and pre-order traversal of a binary tree are “dbeafcg” and “abdecfg” respectively. What would be the post-order traversal for the same binary tree? [(CO3)(Analyze/IOCQ)]

4 + 5 + 3 = 12

7. (a) Provide a pseudo code to perform a Level order traversal of a rooted tree. What is the data structure that you have used in your algorithm? For example the level order traversal of the given tree would be: A, B, C, D, E, F, G



[(CO6) (Create/IOCQ)]

(b) Provide a pseudo code to count the number of leaf nodes of a binary tree. [(CO3) (Apply/IOCQ)]

6 + 6 = 12

Group - E

8. (a) When should we use counting sort over other sorting methods? Explain the algorithm for counting sort and explain its time complexity from the pseudo-code. [(CO4, CO6)(Evaluate, Analyze/IOCQ)]

   (b) When does interpolation search perform like a linear search? Explain with an example. [(CO1, C02)(Understand/LOCQ)]

   (c) Consider a hash table of size seven with starting index zero and a hash function (3x + 4) % 7 (x is the item that you want to insert in to hash table). Assuming the hash table is initially, empty. then what will be the content of the hash table after inserting 1, 3, 8, 10. [(CO1, C02)(Understand/LOCQ)]

6 + 3 + 3 = 12

9. (a) Say you have a data set of size n and you want to find the second largest element form the data set. Suggest an algorithm to find the second largest element from the data set in a single pass. Have you used any special data structure to implement it? [(CO4, CO5)(Analyze, Evaluate/HOCQ)]

   (b) In a closed hashing we are using an array of size 10 whose 0, 1, 2, 5, 6, 7 indexed positions are already filled and the hash function is (key+10) %10, then what will be the probability, that the next key which is going to be inserted should not be placed at 4th index . [(CO4, CO5)(Analyze, Evaluate/HOCQ)]

   (c) What is stable sort? [(CO1, CO2)(Understand/LOCQ)]

6 + 4 + 2 = 12

Cognition Level	LOCQ	IOCQ	HOCQ
Percentage distribution	40	31	29

Course Outcome (CO):

After the completion of the course students will be able to

CSEN2101.1 Understand and remember the basics of data structures and how time complexity analysis is applicable to different types of algorithms.

CSEN2101.2 Understand the significance and utility of different data structures and the context of their application. (For example, the queue in front of ticket counters uses first-in-first-out paradigm in a linear data structure)

CSEN2101.3Apply different types of data structures in algorithms and understand how the data structures can be useful in those algorithms.

CSEN2101.4 Analyse the behaviour of different data structures in algorithms. (For example, given an algorithm that uses a particular data structure, how to calculate its space and time complexity.)

CSEN2101.5 Evaluate solutions of a problem with different data structures and thereby understand how to select suitable data structures for a solution. (For example, what are the different ways to find the second largest number from a list of integers and which solution is the best.)

CSEN2101.6 Evaluate different types of solutions (e.g. sorting) to the same problem.

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