DESIGN & ANALYSIS OF ALGORITHMS
(CSEN 2201)

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) Let *S* be a sorted array of n integers. Let t(n) denote the best case time taken by an algorithm to determine if there are two elements with sum less than 1000 in *S*. Which of the following statements is true?
(a) t(n) is O(1) (b) $n < t(n) < n\log_2 n$
(c) $n\log_2 n < t(n) < \binom{n}{2}$ (d) $t(n) = \binom{n}{2}$.

(ii) The maximum number of nodes present at height h of any n – element heap is –
(a) $\lfloor n/2^{h+1} \rfloor$ (b) $\lfloor n/2^{h} \rfloor$ (c) $\lceil n/2^{h+1} \rceil$ (d) None of (a), (b) & (c).

(iii) Which of the following set is NP-complete?
(a) Travelling salesman problem, Longest simple path problem
(b) Matrix-chain multiplication problem, Fraction knapsack problem
(c) Edge-cover problem, 2-CNF
(d) Finding all pair shortest path of a graph with negative edges.

(iv) The lower bound for comparison sort is
(a) $O(n \log_{10} n)$ (b) $O(n \log_2 n)$ (c) $O(n \log_e n)$ (d) all of (a), (b) & (c).

(v) Dijkstra's Algorithm cannot be applied on
(a) Directed and weighted graphs
(b) Graphs having negative weight function
(c) Unweighted graphs
(d) Undirected and unweighted graphs.

(vi) A connected, undirected graph G = (V, E) may have more than one minimum spanning trees (MSTs) if –
(a) the graph contains more than one edges with same weights
(b) if all the edges have unique weights
(c) if there exists a –ve edge reachable from the source in the graph
(d) None of the above.

(vii) Consider the following recurrence equation:

$$T(n) = \begin{cases} k & \text{for } n = 1 \\ 3T(n/2)+kn & \text{for } n > 1 \end{cases}$$

Which of the following statements is TRUE?

(a) $T(n) = O(n^{1.59})$ (b) O(1) (c) O(nlog n) (d) None of (a), (b) & (c).

(viii) Say, we need to multiply three matrices A1, A2 and A3, with dimensions P=[20,4,10,5]. What will be the minimum number of scalar multiplication needed?

(a) 600 (b) 120 (c) 800 (d) 200.

(ix) The number of comparisons required to find the minimum and maximum of any four distinct numbers stored in any arbitrary order is -

(a) 5 (b) 6 (c) 3 (d) 4.

(x) If the 2nd largest of n elements can be found in X number of comparisons, then X =

(a) 2(n – 1) (b) $n + \lceil \lg n \rceil - 2$ (c) $n + \lceil \lg n \rceil$ (d) $\lceil 3n/2 \rceil$.

Group - B

2. (a) You are taught in the class the MERGE procedure as a subroutine in the merge sort algorithm. Write the pseudo code of that procedure MERGE (A, p, q, r), where A is an array and p, q and r are indices into the array such that p<=q<r. This procedure will merge the two sorted sub-arrays A[p..q] and A[q+1..r] to form a single sorted array. *[(CSEN2201.2) (Remember, Understand/LOCQ)]*

(b) Apply the above procedure that you have written on the following two sorted sub-arrays to form a single sorted array:

2	4	5	7

1	2	3	6

You are supposed to show all the steps. *[(CSEN 2201.2) (Apply/IOCQ)]*

(c) Analyze the time complexity of the procedure that you have written in part (a) in worst case. *[(CSEN 2201.4) (Apply/IOCQ)]*

5 + 5 + 2 = 12

3. (a) (i) Analyse the time complexity of the following function, where A is an array of n elements and Max_Heapify(A, i) takes O(floor(log n)) time.

Build_Max_Heap(A) *[(CSEN2201.4)(Analyze/IOCQ)]*

1. heap_size(A) ← length(A)
2. for (i ←⌊length(A)-2/2⌋ down to 0)
3. Max_Heapify(A, i).

(ii) Solve the following recurrence using master theorem:

$T(n)=3T(n/4) +2 n^2$ if n > 1, T(1) = 2 *[(CSEN2201.3)(Apply/IOCQ)]*

(iii) Say you are given elements: 5,4,3,2. *[(CSEN2201.2)(Understand/LOCQ)]*

How many comparisons will be needed to sort these elements in ascending order using quick sort? Show all the steps.

(b) Use recursion tree method to guess the solution of the following recurrence:

$T(n)=4T(n/4) + n$ if n > 1, T(1) = 1

Verify that your guess is correct using substitution method.

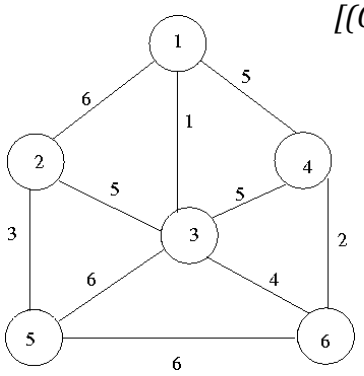
[(CSEN2201.3)(Apply/IOCQ)]

(3 + 3 + 3) + 3 = 12

Group - C

4. (a) Find out an optimal Huffman code for the following set of frequencies.
a: 15 b:20 c:25 d:35 e:5. *[(CSEN2201.3)(Apply/IOCQ)]*

(b) Find out the Minimum Spanning Tree of the following graph using Prim's Algorithm with source vertex '2'.
[(CSEN2201.3)(Apply/IOCQ)]



(c) If there is a negative weighted edge in a given graph, will Bellman Ford always be able to give a solution? *[(CSEN2201.1)(Remember/LOCQ)]*

5 + 5 + 2 = 12

5. (a) Define a Spanning tree of a graph with suitable example.
[(CSEN2201.2)(Remember,Understand/LOCQ)]

(b) Consider an undirected graph G (V, E) with the set of vertices V = {v1, v2. v3, v4, v5, v6, v7}. E is a set of edges of the form (u, v ,w) which denotes an undirected edge between u and v with weight w. E={(v1,v2,4), (v2,v3,2), (v2,v4,6), (v3,v1,3), (v3,v4,2), (v1,v5,4), (v4,v5,3), (v5,v6,1), (v5,v7,3), (v6,v3,4), (v2,v7,5), (v4,v7,3)}. Illustrate the steps of finding a minimum spanning tree using Kruskal's algorithm. Your illustration should have the application of UNION-FIND operations on disjoint data structures. *[(CSEN2201.3)(Apply/IOCQ)]*

(c) If you are given an undirected, un-weighted graph G = (V, E), then how can you find the shortest path from a given source vertex S to all the remaining vertices of G in O (V+E) time? Give brief reasons of your answer.
[(CSEN2201.4)(Analyze/IOCQ)]

2 + 7 + 3 = 12

Group - D

6. (a) Suppose you are asked to perform a sequence of n operations on a data structure in which the i^{th} operation costs i if i is an exact power of 2, and 1 otherwise. Use aggregate analysis to determine the amortized cost per operation. *[(CSEN2201.4)(Evaluate/HOCQ)]*

(b) Construct the string-matching automaton for the pattern P = aabab and illustrate its operation to find this pattern P on the text string T = aaababaabaababaab. *[(CSEN2201.6)(Create/HOCQ)]*

5 + (3 + 4) = 12

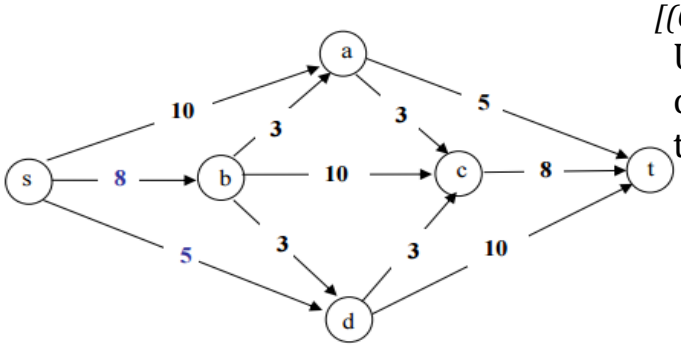
7. (a) Define the prefix function Π in the context of KMP pattern matching algorithm and also write it mathematically. *[(CSEN2201.1)(Remember/LOCQ)]*
Suppose we make the prefix function work on the pattern P: ababaabab. Then give the values of Π(ab), Π(aba), Π(abab), Π(ababa), Π(ababaa), Π(ababaab), and Π(ababaaba). *[(CSEN2201.3)(Apply/IOCQ)]*

(b) Show that the expected number of nodes in a Skip-List containing n items which was created by tossing an unbiased coin is 2n. *[(CSEN2201.1)(Remember/LOCQ)]*
Suppose you are tossing a biased coin such that P(H) = 2/3 and P(T) = 1/3. What is the expected number of tosses required to get a Head? Justify your answer.
[(CSEN2201.2)(Understand/LOCQ)]

(1.5 + (7 × 0.5)) + (4 + 3) = 12

Group - E

8. (a) A flow in G is a real valued function f: V x V → R that satisfies 3 properties. What are they and also state each of them in one sentence.
[(CSEN2201.2)(Remember,Understand/LOCQ)]

(b) 

(Edge capacities are show in bold face.)

Using Ford Fulkerson algorithm, find out the amount of maximum flow of the above flow network.
[(CSEN2201.3)(Apply/IOCQ)]

(c) Prove that flow of any s-t cut is equal to the flow of the network. (s = source, t = sink). *[(CSEN2201.2)(Understand/LOCQ)]*

3 + 6 + 3 = 12

9. (a) Explain the relation between the Clique Decision Problem, Independent Set Decision Problem, and Vertex Cover Decision Problem to show that if one of them is NP-hard, the other two has to be NP-hard. You may use a small example.
[(CSEN2201.5)(Assess/HOCQ)]

(b) Define the following terms in the context of optimization problems -
(i) Approximation scheme (AS), (ii) PTAS, (iii) FPTAS.
[(CSEN2201.1)(Remember/LOCQ)]

(c) Design a 2-approximation algorithm for the Vertex Cover Problem and state its complexity. Prove that the approximation ratio is indeed bounded by 2.
[(CSEN 2201.5)(Design/HOCQ)]

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

Cognition Level	*LOCQ*	*IOCQ*	*HOCQ*
Percentage distribution	*27.60*	*50.52*	*21.88*

Course Outcome (CO):

After the completion of the course students will be able to
CSEN2201.1. Remember time complexities of various existing algorithms in different situations.
CSEN2201.2. Understand the basic principles of different paradigms of designing algorithms.
CSEN2201.3. Apply mathematical principles to solve various problems.
CSEN2201.4. Analyze the complexities of various algorithms in worst case, best case and average case.
CSEN2201.5. Assess the computational hardness of a problem and learn how some of the well-known problems are proved to be NP-hard and also design approximation algorithms for some of them.
CSEN2201.6. Create/ Design a good algorithm for a new problem given to him/ her.
**LOCQ: Lower Order Cognitive Question; IOCQ: Intermediate Order Cognitive Question; HOCQ: Higher Order Cognitive Question.*