B.TECH/CSE/4TH SEM/CSEN 2201/2021

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 <u>any 5 (five)</u> from Group B to E, taking <u>at least one</u> 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) An algorithm takes T(n) = log n! steps. What is the tightest bound for T(n) in big-Oh notation?
(a) O(nlog log n) (b) O(nlog n)
(c) O(n(log n) ^{n}) (d) O(n $^{\log n}$)

(ii) Worst case time complexity for inserting an element in a sorted array so that it stays sorted is
(a) O(1) (b) O(n) (c) O(n^2) (d) None of these.

(iii) In the KMP algorithm for pattern matching, the suffix function δ(x) is the __est __of the pattern p that is also a __ of x
(a) large, suffix, prefix (b) large, prefix, suffix
(c) small, prefix, suffix (d) small, suffix, prefix.

(iv) How would you classify the time complexity of an algorithm for the 0/1 Knapsack problem with n objects and capacity C which runs in O(nC) time? Assume that the inputs to the problem are the sizes and profits of the n objects and the capacity C.
(a) polynomial (b) exponential
(c) pseudo-polynomial (d) quadratic.

(v) Consider the Floyd Warshall algorithm for all pairs shortest path that is run on a graph G whose edge weights are given as a 2-d matrix. The final answer is written in matrix D. Then the algorithm can be used to detect negative weight cycles in G by
(a) Checking if any main diagonal entry of D assumes a negative value
(b) Checking if any entry above the main diagonal of D becomes negative
(c) Checking if any entry below the main diagonal of D becomes negative
(d) Checking if the determinant of D becomes negative.

(vi) Consider a k-bit binary counter initialized to all zeros. If the cost of an increment operation is measured by the number of bits flipped, what is the amortized cost per operation for n increments starting from all zeros?

(a) O(n) (b) O(nk)
(c) O(nlog n) (d) None of these.

(vii) Suppose we implement Quicksort in two ways:
A. Using Median element of the array as pivot after applying the O(n) median-of-medians algorithm.
B. Using the middle element at location (low + high)/2 as pivot. Here low and high are the lowest and highest indices of the part of the array on which Quicksort is invoked.
Which of the above results in O(nlog n) worst case complexity for Quicksort?

(a) A only (b) B only
(c) Both A and B (d) Neither A nor B.

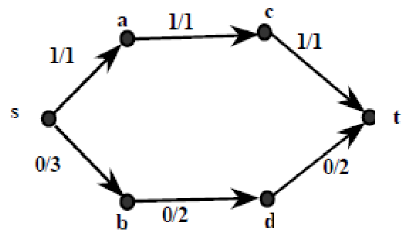
(viii) Consider the following code snippet:

```
int fun(int n)
{
  int count = 0;
  for(int i = 0; i < n; i++)
         for(int j = i; j > 0; j--)
                 count = count + 1;
  return count;
}
```

What is the time complexity of fun() in the worst case?

(a) θ (n) (b) θ (n^2) (c) θ (n logn) (d) θ (n loglogn)

(ix) What is the residual capacity of the augmenting path for the flow network shown below? Note that for each 'a/b' written beside it implies flow = 'a' and capacity = 'b'.



(a) 0.5 (b) 1 (c) 1.5 (d) 2

(x) A problem in NP is NP-complete if

(a) it can be reduced to the 3-SAT problem in polynomial time
(b) the 3-SAT problem can be reduced to it in polynomial time
(c) it can be reduced to any other problem in NP in polynomial time
(d) some problem in NP can be reduced to it in polynomial time.

Group – B

2. (a) Let T(n) = 1 if n < 3 and T(n)=3T(n/3) + √n otherwise. What is T(n) in big-Oh notation? Find it using the Master theorem or otherwise.

(b) For the input, 10, 90, 80, 200, 250, 300, 400, how many times you need to call max-heapify function to ensure that it is a max heap? Please also identify each call to max heapify.

(c) Prove that the average case time complexity of a binary search is O(n logn).

4 + 3 + 5 = 12

3. (a) You are provided with an input of an unsorted sequence S of n integers, where n>1. You would like to find both the second order statistic and the nth order statistic of S. Show that they can be found out in $(3\lfloor n/2 \rfloor + \lceil \lg n \rceil - 1)$ number of pair-wise comparisons.

(b) Let $X[1 .. n]$ and $Y [1 .. n]$ be two arrays, each containing n numbers already in sorted order. Give the outline of an $O(\lg n)$-time algorithm to find the median of all $2n$ elements in arrays X and Y.
Hint: This is not difficult. Just think how to keep reducing the search space. It is important that the algorithm remains O(lg n) and does not become O(n).

7 + 5 = 12

Group – C

4. (a) Show that the Kruskal's algorithm correctly provides us a minimal spanning tree.

(b) (i) What is the *prefix property* in coding and how does it help?
(ii) A networking company uses a compression technique to encode the message before transmitting over the network. Suppose the message contains the following characters with their frequency:

Character	Frequency
a	5
b	9
c	12
d	13
e	16
f	45

Each character in input message takes 1 byte. If the compression technique used is Huffman Coding, how many bits will be saved in the message?

(c) Given two sequence X[1..7] and Y[1..6], where X = ABCBDAB and Y = BDCABA. Find any one longest subsequence common to both and briefly show the steps.

4 + (2 + 3) + 3 = 12

5. (a) Matrices A, B and C, of dimensions 10 × 24, 24 × 6 and 6 × 17 respectively, are provided. You are told to ensure that when computing the matrix product A.B.C

the number of scalar multiplications will be a minimum. How would you parenthesize the product, and what would be the minimum number of scalar multiplications you would need to perform?

(b) Write the pseudo codes for Initialize-Single –Source(G,s) and Relax(u,v,w) functions used in Dijkstra's algorithm to find out the shortest paths from a source vertex(s) to all other vertices of a given graph G(V, E) . Assume that w is the non-negative weight of the edge(u, v). In the given graph G, there is a non-negative weight for each edge (u, v) ϵ E.
Let s $\rightsquigarrow$ u → v be a shortest path in the given graph G(V,E) for some vertices u, v ϵ V, where the notations used have the usual meanings, like there is a path from s to u, followed by a directed edge from u to v. Then after INITIALIZE SINGLE-SRC(G, s) and a sequence of relaxation steps that include the call RELAX(u, v, w), if d[u] = δ(s,u) (shortest path weight from s to u), at any time prior to the call, then d[v] = δ(s,v) at all times after the call. Prove this statement. Note that, d[x] maintains an upper bound of the shortest path weight from source s to vertex x.

(c) Let G=(V, E) be a connected, undirected graph with a real-valued weight function w defined on E. Let A be a subset of E that is included in some MST for G, let (S, V-S) be any cut of G the respects A, and let (u, v) be a light edge crossing (S, V-S). Then edge (u, v) is safe for A. Prove it.

2 + (3 + 4) + 3 = 12

Group – D

6. (a) Amortized analysis is different from average-case analysis as no __________ is involved in the former one but it is involved in the latter.

(b) Suppose that the procedure 'Compute-Prefix-Function(P)' is available to you. Write the pseudo-code for KMP matcher (T, P) for text T and pattern P using the above procedure.

(c) Show how to implement a queue with two ordinary stacks so that the amortized cost of each enqueue and dequeue operations remains O(1) and does not become O(n). Explain your amortized analysis.

1 + 5 + (3 + 3) = 12

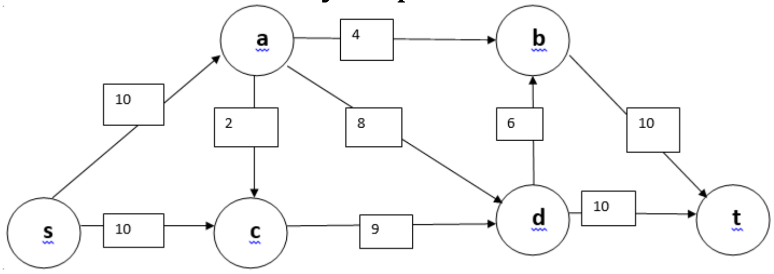
7. (a) Construct the prefix function for the pattern 'aadbaaba' that can be used by the Knuth-Morris-Pratt algorithm.

(b) Suppose we perform a sequence of n operations on a data structure in which the i^{th} operation cost is i, if i is an exact power of 2, and 1 otherwise. Use aggregate analysis to determine the integer k such that the amortized cost A per operation satisfies the following $k \le A < k + 1$.

6 + 6 = 12

Group – E

8. (a) Run Ford-Fulkerson algorithm on the above Flow network and find out the maximum flow value. Show every step in detail

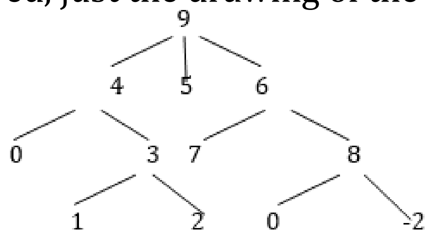


(b) Consider the execution of the algorithm for finding connected components in an undirected graph using the UNION-FIND data structure. If the graph has n vertices, e edges and k components, how many times in terms of n, e and k are the MAKE_SET, FIND_SET and UNION operations invoked?

(c) State which problems are solvable in polynomial time and which are NP-hard -
(i) Hamiltonian Cycle
(ii) Subset Sum Problem
(iii) Set Cover Problem
(iv) Edge Cover Problem.

6 + 4 + 2 = 12

9. (a) Draw the tree after the call of FIND_SET(8) using path compression heuristic. No explanation required, just the drawing of the tree.



(b) Show that the approximation algorithm 'APPROX-VERTEX-COVER' taught in class is a polynomial-time 2-approximation algorithm. State its time complexity.

(c) Reduce the CNF, (x v y v z') ^ (x v y' v w') ^ (x' v z v w') ^ (z' v y' v w') to an instance of a clique decision problem, where x, y, v, and w are Boolean variables. What is the size of the clique that you will be looking for? Please write all the steps and the terms used in your steps in detail.

2 + (4 + 1) + (2 + 1 + 2) = 12

Department & Section	Submission Link
CSE - A	https://classroom.google.com/u/0/w/OTM0Njk3NzA3MDFa/tc/MzcxNjU0OTU3ODUw
CSE - B	https://classroom.google.com/w/MzExMjM4MTA3MDAy/tc/Mzc0MTQxNDEzODI4
CSE - C	https://classroom.google.com/c/MzEyNTEyNTExMjQw/a/Mzc0MTM0NDE1OTU3/details
BACKLOG & IMPROVEMENT	https://classroom.google.com/c/Mzc0MTQxNDEzOTUw/a/Mzc0MTQyMDk3NzI4/details