Exam.Code:0916
Sub. Code: 6781

1079

B.E. (Computer Science and Engineering)
Fourth Semester

CS-401: Analysis and Design of Algorithms

Time allowed: 3 Hours

Max. Marks: 50

NOTE: Attempt five questions in all, including Question No. I which is compulsory and selecting two questions from each Section.

x-x-x

Q1. (a) What is the significance of asymptotic notation?
(b) State the control abstraction for Divide and Conquer algorithm designing strategy. [5 x 2]
(c) Compare greedy method and dynamic programming method.
(d) What is the Principle of Optimality?
(e) Explain the difference between explicit and implicit constraints using 8-queens problem.

Section – A

Q2. (a) Solve the recurrence $T(n) = T(\sqrt{n}) + 1$ using substitution method. Your solution should be asymptotically tight. [5]
(b) Solve the recurrence $T(n) = 4T(n/2) + n^2 \lg n$ using master method. [5]

Q3. (a) Write QuickSort algorithm based on divide and conquer strategy. Show how QuickSort sorts the array A = [13, 19, 9, 5, 12, 8, 2, 6, 21]. Also perform the space/time complexity analysis of QuickSort algorithm. [5]
(b) Write a short note on Strassen's Matrix Multiplication algorithm. How would you modify the same to multiply n x n matrices in which n is not an exact power of 2? Also perform the space/time complexity analysis of 1^{st} algorithm as well as the modified algorithm. [5]

Q4. (a) State the Greedy method based control abstraction for the subset paradigm. Apply it to solve the Knapsack problem. Find an optimal solution to the knapsack instance n=7, m = 15, $(p_1, p_2, \ldots, p_7)$ = (10, 5, 15, 7, 6, 18, 3) and $(w_1, w_2, \ldots, w_7)$ = (2, 3, 5, 7, 1, 4, 1). [5]
(b) Define a Spanning tree. Can Kruskal's algorithm return different spanning trees for the same input graph G? If yes, justify the same. Also show that for each minimum spanning tree T of G, there is a way to sort the edges of G in Kruskal's algorithm so that the algorithm returns T. [5]

Section – B

Q5. (a) Define the problem of Longest Common Subsequence. Also, design an algorithm to find the longest monotonically increasing subsequence of n numbers. Find the running time of the designed algorithm. [5]
(b) Write an algorithm AllPaths for All pairs shortest path problem using dynamic programming. Modify the function AllPaths so that a shortest path is output for each pair of vertices (i, j). Also perform the space/time complexity analysis of the AllPaths algorithm as well as the modified algorithm. [5]

P.T.O.

Q6. (a) Write an algorithm NQueens for n-queens problem using backtracking. Further redefine Place(k,i) function so that it either returns the next legitimate column or an illegal value. Modify the complete n – queens solution to accommodate above altered strategy. Will this alteration make the previous algorithm more efficient or not? Justify your observation. [5]

(b) Write the recursive backtracking algorithm SumOfSub for sum of subsets problem. Let m = 35, run SumOfSub on the data

1. w = {5, 7, 10, 12, 15, 18, 20}
2. w = {20, 18, 15, 12, 10, 7, 5}
3. w= {15, 7, 20, 5, 18, 10, 12}

Are there any visible differences in the computing time? [5]

Q7. (a) State your observation on the commonly believed relationship among P, NP, NP - Complete and NP – Hard problems. [4]

(b) Obtain a nondeterministic algorithm of complexity O(n) to determine whether there is a subset of n numbers a_i, 1<= i <= n, that sums to m. [3]

(c) Show that clique optimization problem reduces to clique decision problem. [3]

x-x-x