YMCAUST -MAY 2019

B.Tech. , IV SEMESTER (Re-Appear)(Scheme 2017)

Analysis and Design of Algorithm(CE-204C)

Time: 3 Hours **Max. Marks:75**

Instructions:
1. *It is compulsory to answer all the questions (1.5 marks each) of Part -A in short.*
2. *Answer any four questions from Part -B in detail.*
3. *Different sub-parts of a question are to be attempted adjacent to each other.*

PART –A

Q1 (a) What is an algorithm? Describe its main features. (1.5)

(b) Derive the worst case complexity of quick sort? (1.5)

(c) Define union and find. (1.5)

(d) Define principle of optimality. In which technique is it used? (1.5)

(e) Which of the following sorting algorithm are stable: insertion sort, merge sort or quick sort? And how? (1.5)

(f) What is breadth first and depth first search? Give example. (1.5)

(g) What are implicit and explicit constraints in backtracking methods? (1.5)

(h) What is least cost search? (1.5)

(i) What is chromatic number? How is it determined? (1.5)

(j) What are deterministic and nondeterministic algorithms? (1.5)

PART –B

Q2 (a) Explain Strassen's matrix multiplication. (5)

(b) Explain Prim's and Kruskal's algorithms. (10)

Q3 (a) What is the key feature of dynamic programming that distinguishes it from divide and conquer technique. What are the main steps to solve a problem using dynamic programming. (5)

(b) Describe the algorithm for merge sort and derive its complexity. (10)

Q4 (a) Explain greedy knapsack algorithm. Give example. (5)

(b) Explain job sequencing with deadlines. Generate a solution using Job Scheduling for the following jobs (10)

Job	A	B	C	D	E
profit	100	19	27	25	15
deadline	2	1	2	1	3

Q5 (a) Explain n-queens algorithm using backtracking. (5)

(b) Solve the following knapsack problem using branch and bound (10)
N=4, (p1,p2,p3,p4) = (10,10,12,18), (w1,w2,w3,w4)=(2,4,6,9)

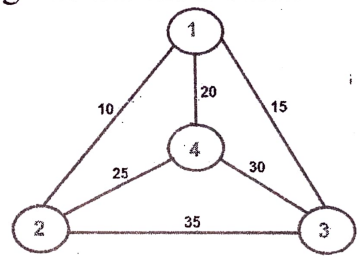
Q6 (a) Define Cook's theorem. Write an algorithm for non-deterministic quicksort. (5)

(b) Construct an optimal binary search tree for following. (10)

A B C D

p(1:4) =(0.1,0.2,0.4,0.3)

Q7 Solve the following TSP using Branch and Bound. (15)



****