Total No. of Questions : 8]

SEAT No. :

P-485

[Total No. of Pages : 3

[6003]-705

T.E. (Information Technology)

DESIGN AND ANALYSIS OF ALGORITHM

(2019 Pattern) (Semester - I) (314445(A)) (Elective - I)

Time : 2½ Hours] *[Max. Marks : 70*

Instructions to the candidates :

1) *Answer Q.1 or Q.2, Q.3 or Q.4, Q.5 or Q.6, Q.7or Q.8*
2) *Neat diagrams must be drawn wherever necessary.*
3) *Figures to the right side indicate full marks.*
4) *Assume suitable data, if necessary.*

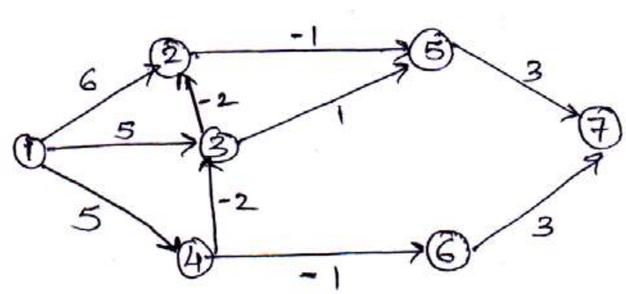
Q1) a) Explain the coin change making problem with suitable example. **[9]**

b) Solve the following instance of Knapsack problem by dynamic programming approach:

n = 6; M = 165 and (pl,p2,p3,p4,p5,p6) = (wl,w2,w3,w4,w5,w6) = (100,50,20,10,7,3). **[9]**

OR

Q2) a) Use Bellman Ford algorithm for finding the shortest path for the graph. **[9]**



b) What is dynamic programing? Is this the optimization technique? What are the drawback of dynamic programing. **[9]**

P.T.O.

Q3) a) Explain 8-Queen problem and explain the following terms with respect to 8-Queens problem. **[8]**

i) State space tree

ii) Live node

iii) Static tree

iv) Solution state

v) Answer state

b) State the principal of backtracking and write backtracking algorithm for graph coloring **[9]**

OR

Q4) a) What is Backtracking? Write an algorithm for backtracking solution to the 0/1 knapsack problem. **[8]**

b) Let W= {5, 10, 12, 13, 15, 18} and M = 30. Find all possible subsets of W that sum to M. Draw the portion of state space tree. **[9]**

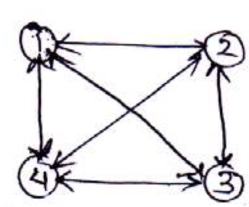
Q5) a) What is the difference between backtracking and branch & bound? Illustrate using the 0/1 knapsack problem. **[9]**

b) Write an algorithm for Least cost(LC) branch & bound. **[9]**

OR

Q6) What is traveling salesperson problem? Find solution to the following TSP using branch & bond. **[18]**

$$\begin{bmatrix} \infty & 10 & 15 & 20 \\ 5 & \infty & 9 & 10 \\ 6 & 13 & \infty & 12 \\ 8 & 8 & 9 & \infty \end{bmatrix}$$



Q7) a) Explain NP-Hard, NP-Complete, Decision problem & Polynomial time algorithm. **[9]**

b) Prove that clique problem is NP complete. **[8]**

OR

Q8) a) Prove that satisfiability problem is NP complete. **[9]**

b) Prove that vertex cover problem is NP complete. **[8]**

✷ ✷ ✷