#### **B. TECH. SEM IV CE**

**SUB: DESIGN AND ANALYSIS OF ALGORITHM**

**CLASS ASSIGNMENT 1**

#### Following is a supplementary assignment based on the topics discussed in the class. Please solve the problems and submit it on or before: 14th March 2016. For any difficulty or doubts contact us at our office (Lab 2/Lab7) during working hours.

#### Q: 1 What is principle of optimality? Illustrate with example.

#### Q: 2 Can we use graph with a negative weight edges as an input to Floyd algorithm to find all

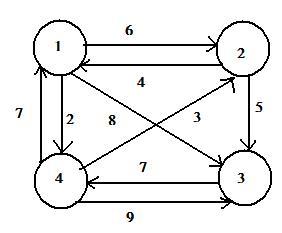
#### pair shortest path? Can a graph have negative cycle as an input? Justify your answers.

Q:3 Find any one Longest Common Subsequence for the strings “GAGCTGC” and “TGATC” using dynamic programming approach.

Q: 4 Solve the following 0/1 knapsack problem using dynamic approach and find items included in the knapsack. Assume maximum weight holding capacity of knapsack is 22.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| *Item No.* | 1 | 2 | 3 | 4 |
| *Weight* | 15 | 4 | 10 | 3 |
| *Profit* | 30 | 12 | 25 | 8 |

Q: 5 Find the shortest path between every pair of vertices for the graph shown below using Floyd-Warshall algorithm.



Q: 6 Let *(p1,....,p5) = (20, 15, 10, 5, 1)* and *(d1,...d5) = (2, 2, 1, 3, 3)*, *pi=profit , di= deadline* for *i=1 to 5*. Design Greedy strategy to find optimal solution of job sequencing so that profit earned is maximum and all the jobs in the sequence are completed by its deadline.

Q: 7 Prove that fractional Knapsack's greedy algorithm always gives optimal solution.

Q: 8 Write an algorithm which finds longest common subsequence using dynamic programming method. What is the time complexity of the algorithm?

Q: 9 Apply Kruskal's algorithm on graph shown in Graph G1. Explain each step briefly.

1 2

2

3

1

2

5

4 6 4

5

4

3

6 8 7

4

7

6

3

(Graph G1)

Q: 10 Write kruskal’s algorithm for MST.

Q: 11 Write knapsack problem’s solution algorithm using greedy strategy.

Q: 12 Solve Traveling Salesperson problem for 5 cities using branch and bound method given distance matrix as below:

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
|  | **A** | **B** | **C** | **D** | **E** |
| **A** | 0 | 10 | 15 | 10 | 3 |
| **B** | 5 | 0 | 9 | 2 | 9 |
| **C** | 6 | 13 | 0 | 7 | 2 |
| **D** | 8 | 3 | 4 | 0 | 1 |
| **E** | 4 | 9 | 6 | 10 | 0 |

Q: 13 What is "optimal substructure property'" in a dynamic programming problem? Explain with example.

Q: 14 General idea of greedy solution for Minimum Spanning Tree is to add edges in increasing order of weights. Is it possible to find MST by eliminating edges in decreasing order of weights? Will the algorithm be greedy and always give optimal solution?

Q: 15 Show that the sorting problem is polynomial time reducible to convex hull problem.

Q: 16 Define P and NP class of problem. Give example of problems which are in P, in NP, in both and in none of them.

Q: 17 Write non deterministic algorithm for TSP problem.

Q: 18 Write a dynamic programming algorithm to solve Chained Matrix Multiplication problem. Explain the working of the problem with an example.

Q: 19 Solve the following 0/1 knapsack problem using Branch and Bound approach. Find items included in the knapsack. Assume maximum weight holding capacity of knapsack is 22.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| *Item No.* | 1 | 2 | 3 | 4 |
| *Weight* | 15 | 4 | 10 | 3 |
| *Profit* | 30 | 12 | 25 | 8 |

Q: 20 Write a back tracking algorithm to solve knapsack problem described in question 19. Draw the state space tree.