**Department of Computer Science and Engineering**

**Practice Example: Set – 1: Unit: 3: Greedy Algorithms**

**Submission date: 15-07-2019**

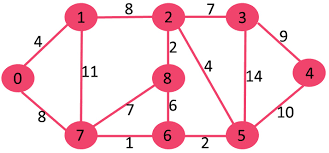
**Knapsack Algorithm**

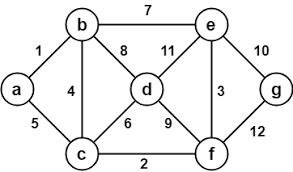
Example: 1

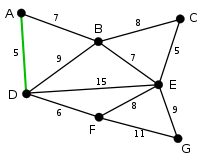
Capacity: 19

Number of objects: 9

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Index** | **1** | **2** | **3** | **4** | **5** | **6** | **7** | **8** | **9** |
| Profit | 12 | 7 | 15 | 4 | 9 | 11 | 5 | 10 | 3 |
| Weight | 3 | 4 | 2 | 3 | 2 | 3 | 5 | 2 | 3 |







**USE above three graphs and solve the problems given below.**

**Prims Algorithm:**

Example: 1

Convert the graph into Cost matrix [n x n]

Generate near array at every stage.

Refer the algorithm and find the value of variable “j” at every stage

Design the matrix of for Minimum cost spanning tree.

**Kruskal Algorithm**

Example: 1

Convert the graph into cost matrix [n x n]

Prepare edge list in ascending order of weight.

Construct the table to design spanning tree of minimum cost.

Find out how many edges are not considered in execution.

Why an edge is rejected.

What is graph contraction and graph expansion. How it is used in Kruskal Algorithm

**Reverse Delete Algorithm:**

Example 1:

Convert the graph into cost matrix [n x n]

Prepare edge list in descending order of weight.

Demonstrate the deletion of edge from graph with alternate path rule.

Find out how many edges are not considered in execution.

State whether the design logic uses contraction or expansion.