

Worksheet-1: Knapsack Algorithm, Prim Algorithm, Reverse Delete Algorithm, Dijkstra Algorithm.

Mode of submission: Hand-written in separate note book

Max.marks [05]

One line Answer:

- Formula for deciding fraction of last object placed in knapsack
- Prims' algorithm: Condition for "extract minimum" and "update key" steps.
- Reverse Delete algorithm: Suggest suitable strategy, which will find an alternate path to any vertex from a vertex. [Refer Graph searching algorithms]
- Any two differences between Prims and Reverse Delete algorithm.
- Dijkstra's algorithm: Condition for updating the distance for new node added in distance tree.
- Role of Heap memory in Prims' algorithm.
- One drawback of Dijkstra algorithm.

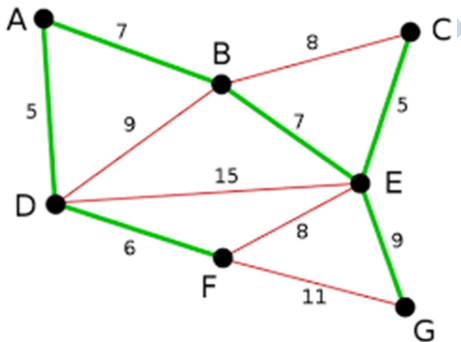
Solve the following numerical and demonstrate role of different data structures in solution.

- For the following scenario suggest suitable strategy to attain maximum throughput.
 CPU time: 15 seconds
 Details of process with throughput and time requirement

Process	1	2	3	4	5	6	7
Time	4	3	2	1	2	3	2
Throughput	12	16	5	7	8	12	10

If at run time process-1, throughput increases to 18, find total change in system throughput.

- Consider the following graph: Construct the data structures as shown in the figures.

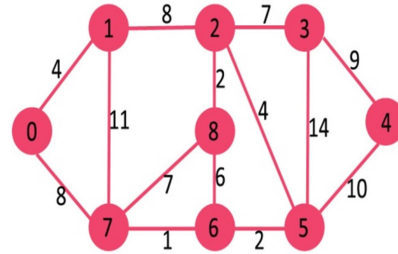


a	b	c	d	e	f	g	
-	a	-	-	-	d	-	AD

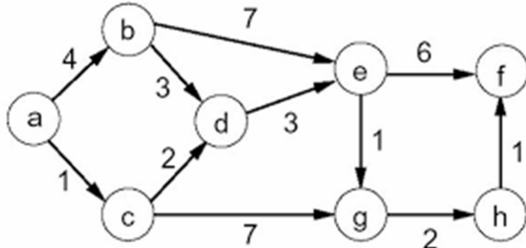
Fill the status of near array at each step.

Row	Column	Value

c. Find the storage requirement of Reverse Delete Algorithm for the following graph. The storage requirement depends upon: total number edges in the graph, total number of edges in the minimum cost spanning tree. Both graphs can be stored as: Matrix or Adjacency matrix [requirement of nodes].



Draw the MCST, executing reverse delete algorithm.

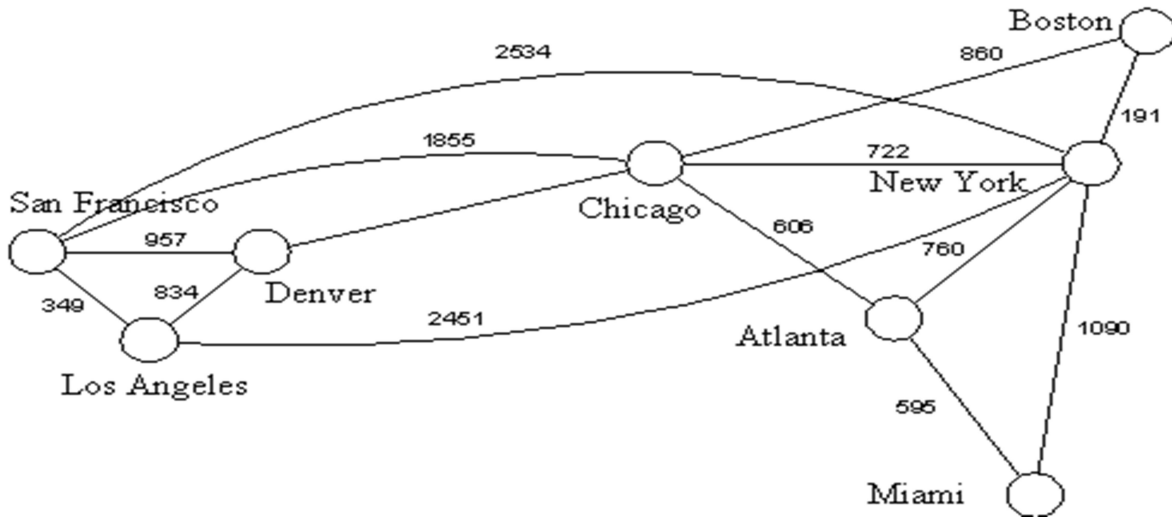


d. Draw the Prim-Tree: Step by step execution and find out total cost of spanning tree. Use Heap structure. Comment on time complexity of prim's algorithm, if heap structure representation is used for storage.

For example:



e. Implement Single Source Shortest Path algorithm for following map with distances: Find the contents of three arrays: distance, parent and selection in each step of execution.



Assume source: San Francisco