20181CSE0621

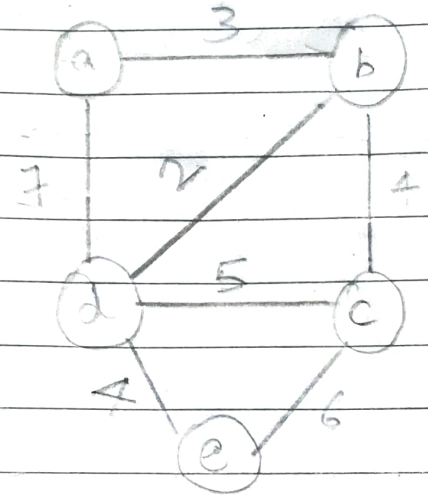
Sai Ram. K.

20181CSE0621

08/01/2021

Part-C

Q.1] Graph:-



↳ Dijkstra's Algorithm:-

Let start node & node x's distance be x.

Step1: Assign to every node a distance value & set it to 0 for initial node & ∞ to other node

Step2: Set initial node as current & mark other nodes as unvisited.

Step3: Consider current node & calculate distance of neighbors.

Step5: Set unvisited nodes with smallest distance from initial node as next & continue.

Step6: When all nodes are visited the algorithm ends.

20181CSE0621

Saifa

Date
Page

Algorithm :-

Let s be source node

dist[s] ← 0

for all v ∈ V - {s} do

dist[v] ← ∞

S ← ϕ

Q ← V

while (Q ≠ ϕ) do

u ← min dist (Q, dist)

S ← S ∪ {u}

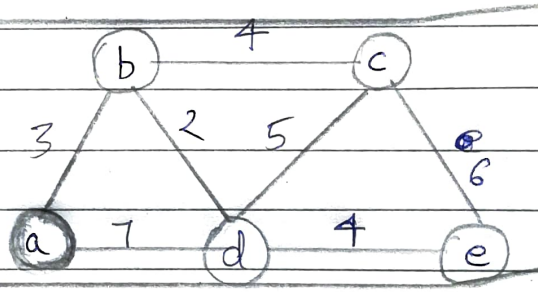
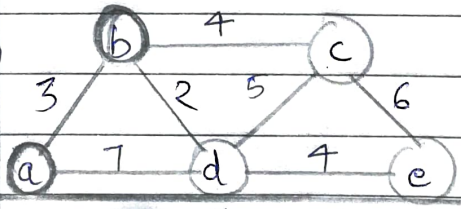
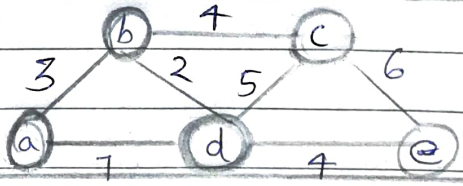
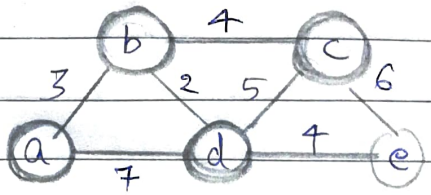
for all v ∈ neighbors[u] do

if (dist[v] > dist(u) + cost[u][v])

then dist[v] = dist[u] + cost[u][v]

return dist.

→ Finding shortest path :-

True Vertices	Remaining Vertices	
a(-,0)	b(a,3), c(-,∞) d(a,7), e(-,∞)	
b(a,3)	c(b,3+4), d(b,3+2) e(-,∞)	
d(b,5)	c(b,7) e(d,5+4)	
e(d,9)		

Hence, Shortest path is a -> b -> d -> c