Exp - 9

VAISHNAV.S
1BM18CS121

Question : Use distance vector algorithm to find distance between all points in the given topology.

Main :

class topology :

```
def __init__(self, array_of_nodes):
    self.node = array_of_nodes
    self.edges = []

def add_dist_connection(self, P1, P2, cost):
    self.edges.append((P1, P2, cost))
    self.edges.append((P2, P1, cost))

def distance_vector_routing(self):
    dist = collections.defaultdict(int)
    next_hop = {(node, node): node}
    for other_node in self.nodes:
        if other_node != node:
            dist[other_node] = 10000000
    for i in range(len(self.nodes) - 1):
        for edge in self.edges:
```

```
src, dest, cost = edge.
    if dist[src] + cost < dist[dest];
        dist[dest] = dist[src] + cost
            if (src == node;
                next_hop[dest] = dest .
    elif src in next_hop;
        next_hop[dest] = next_hop[src]
Sub print_routing_table(node, dist, next_h))
    Print()
dept print_routing_table(self, node dist next_hop);
        Print(f'Routing table for {node}:')
        Print('Dest  it Cost \t Next hop')
        for dest, cost in dist.items();
        Print(f'{dest} \t {cost} \t {next_hop}.
nodes = input("Enter the node:").Split()
t = topology(nodes)
edges = int(input("Enter the number of connections:"))
for _ in range(edges);
    src, dest, cost = input("Enter [src] [dest] [cost]:").Split()
t.add_direct_connection(src, dest, int(cost)).
```