31/12/2020

COMPUTER NETWORKS

CN - LAB - II

M.S. SATHWICK KIRAN
1BM18CS050
5 - B

CODE

```
Class Topology:
    def __init__(self, array_of_points):
        self.nodes = array_of_points
        self.edges = []
    def add_direct_connection(self, p1, p2, cost):
        self.edges.append([p1, p2, cost])
        self.edges.append([p2, p1, cost])
    def distance_vector_routing(self):
        input collection
        for node in self.nodes:
            dist = collections.defaultdict(int)
            next_hop = {node: node}
            for other_node != node:
                if other_node != node:
                    dist[other_node] = 1000000000 = infinity

        for i in range(len(self.nodes)-1):
            for edge in self.edges:
                src, dest, cost = edge
                if dist[src] + cost < dist[dest]:
```

M.S. SATHWICK KIRAN
1BM18CS050

⇒ dist[dest] = dist[src] + cost

if src == node:
next_hop[dest] = dest
elif src in next_hop:
next_hop[dest] = next_hop[src]
self.print_routing_table(node, dist, next_hop)
print()
def print_routing_table(self, node, dist, next_hop):
print(f'Routing table for {node}:')
print('Dest \t Cost \t Next Hop')
for dest, cost in dist.items():
print(f'{dest} \t {cost} \t {next_hop[dest]}')

nodes = ['A', 'B', 'C', 'D', 'E']
t = Topology(nodes)

t.add_direct_connection('A', 'B', 2)
t.add_direct_connection('A', 'C', 5)
t.add_direct_connection('[redacted]')
t.add_direct_connection('[redacted]')
t.add_direct_connection('[redacted]')
t.add_direct_connection('[redacted]')

t.distance_vector_routing()

⇒ t.add_direct_connection('A', 'D', 3)
t.add_direct_connection('B', 'C', 2)
t.add_direct_connection('B', 'F', 3)
t.add_direct_connection('B', 'G', 4)
t.add_direct_connection('C', 'D', 6)
t.add_direct_connection('C', 'E', 4)
t.add_direct_connection('C', 'F', 4)
t.add_direct_connection('D', 'E', 3)
t.add_direct_connection('E', 'F', 2)
t.add_direct_connection('F', 'G', 5)

t.distance_vector_routing()