Ethan Hebert & Lucas Duran

2-24-23

CSC-450

Project - Link-State and Distance-Vector Routing

Group Member Responsibilities:

- Ethan – Code framework and Link-State Routing
- Lucas – Distance-Vector Routing

How to Run Program:

- python routing.py <FILENAME>.csv
- Choose a viable source node from the topology

Sample Program Runs:

```
$ python routing.py topology-1.csv
Please, provide the source node: u
Shortest path tree for node u:
uw, ux, uwv, uwvy, uwvyz
Costs of the least-cost paths for node u:
u:0, v:6, w:3, x:5, y:10, z:12

Distance vector for node u: 0 6 3 5 10 12
Distance vector for node v: 6 0 3 7 4 6
Distance vector for node w: 3 3 0 4 7 9
Distance vector for node x: 5 7 4 0 7 9
Distance vector for node y: 10 4 7 7 0 2
Distance vector for node z: 12 6 9 9 2 0
```

```
$ python routing.py topology-1.csv
Please, provide the source node: y
Shortest path tree for node y:
yz, yv, yvw, yx, yvwu
Costs of the least-cost paths for node y:
u:10, v:4, w:7, x:7, y:0, z:2

Distance vector for node u: 0 6 3 5 10 12
Distance vector for node v: 6 0 3 7 4 6
Distance vector for node w: 3 3 0 4 7 9
Distance vector for node x: 5 7 4 0 7 9
Distance vector for node y: 10 4 7 7 0 2
Distance vector for node z: 12 6 9 9 2 0
```

```
$ python routing.py topology-2.csv
Please, provide the source node: a
This node is not found in the topology.

Please, provide the source node: x
Shortest path tree for node x:
xy, xyz
Costs of the least-cost paths for node x:
x:0, y:2, z:3

Distance vector for node x: 0 2 3
Distance vector for node y: 2 0 1
Distance vector for node z: 3 1 0
```