COEN 146 – Spring 2018
Lab Project 5

Routing

Demo in the lab and upload your code by midnight on Friday (June 1)

This project consists of building the code to run the link-state algorithm in a router. Assume the topology has N routers.

The following information will be available:

- Router ID, which is its index into the tables below, is given at the command line.
- Number of nodes, N, in the topology will be given by the command line.
- Table with costs, NxN, will be obtained from file1 (name given at the command line).
- Table with machines, machine names, IP addresses, and port numbers, Nx3, will be obtained from file2 (name given at the command line).

Your main data will be:

- Cost table – contains the cost from every node to every node, initially obtained from file1.
- Least cost array – obtained with the link state algorithm.

Your code will have 3 threads:

- Thread 1 reads a new change from the keyboard every 10 seconds, updates the neighbor cost table, and sends messages to the other nodes using UDP. It finishes 30 seconds after executing 3 changes. You may execute this part in the main thread.
- Thread2 loops forever. It receives messages from other nodes and updates the neighbor cost table. When receiving a new cost c from x to neighbor y, it should update the cost in both costs: x to y and y to x.
- Thread 3 loops forever. It sleeps for a random number of seconds (10-20), run the algorithm to update the least costs. After the algorithm executes it outputs the current least costs.

You will need a mutex lock to synchronize the access to the neighbor cost table.

The messages between the routers will have 3 integers:
<Routers' ID><neighbor ID><new cost>

The input for Thread 2, will have two integers:
<neighbor><space><new cost><new line>

The table with costs will look like this if N = 3:

<0,0>	<0,1>	<0,2>
<1,0>	<1,1>	<1,2>
<2,0>	<2,1>	<2,2>

where <i,j> represents the cost between node i and node j. If <i, j> is equal to infinite (defined as 10,000), nodes i and j are not neighbors.

The table with machines will look like this, if N = 3.

<machine0> <IP0> <port>
<machine1> <IP1> <port>
<machine2> <IP2> <port>