

Instructions for JNW2 Project WAN2 – Dijkstra Routing Computation

See chapter 8 of *Understanding Internet Protocols*. Project WAN2 is basically the same as given at the end of Chapter 8. However, in JNW2 the Dijkstra computation is in `DijkstraRouting.java` and the cost function is in `ConnectivityMatrix.pathCost()`.

Your assignment is to complete `DijkstraRouting.java` based on the algorithms in the comments it contains, which are essentially the same as those in *Understanding Internet Protocols*. You are to code method `computeRoutingRow()`. To understand what is needed, first read through `DijkstraRouting.java` to learn how the JNW2 routing computation works. Remember to comment your code completely.

To debug your `DijkstraRouting.java`, right-click on the JNW2 project in NetBeans and select “Properties” and then “Run”. Use the Main Class option to select `JNW2.utility.DijkstraRouting`; also, type under Arguments “WAN.txt” to load the network configuration created in WAN1. Run your code and observe the Routing Matrix listed in the output. You should be able to obtain the correct routing matrix by human computation and compare it to the one printed out.

After debugging, you will run JNW2 as a simulator to complete WAN2. To run your `DijkstraRouting.java` in a simulation, right-click on the JNW2 project in NetBeans and select “Properties” and then “Run”. Use the Main Class option to select `JNW2.gui.GuiController`. When you run the GUI, select configuration file “WAN.txt” that you created in WAN1 and load it. Then add data files `email1-1.1-7.1.txt` and `email1-7.1-1.1.txt` and save it as `WAN2.txt`. Select “Print at” Application Layer, Network Layer, and DLC Layer. Then run the simulation.

NOTE: The purpose of the GUI’s Routing Matrix button is to allow the routing to be inserted interactively. To see the routing matrix produced by your `DijkstraRouting.java`, look in the output text just as you did while debugging.

When WAN2 is working correctly, two “email” messages will be exchanged between nodes 1.1 (router1) and 7.1 (router7), following the lowest-cost path from the routing calculation. The routing matrix produced by your code will be shown in the output after the network configuration.

Submit your `DijkstraRouting.java` and a copy of the output produced when you run the simulation with the `WAN.txt` configuration. Be sure to include your name in the code comment provided.