Computer Networking
USI/Lugano — A. Carzaniga

Assignment 3
December 12, 2016

Assignment 3: Link-State Routing

***Due date:** Thursday, December 22, 2016 at 22:00 CEST*

This is an individual assignment. You may discuss it with others, but your code and documentation must be written on your own.

Problem

The goal of this assignment is to implement a *link-state* routing protocol within the *SimpleNet* network simulator, which is available from the course Web page along with documentation and examples. This simulator is written in Java, so your implementation must also be done in Java.

Your implementation must consist of a class called *LinkStateRouter* that extends the *simplenet.Router* class. You may define and use other classes. In particular, you will certainly need to define some classes that represent the messages exchanged by routers. Your implementation must cover all the fundamental aspects of link-state routing. In particular, (1) a router must discover its neighbors and therefore find out the state of its adjacent links; (2) a router must broadcast its link-state advertisement, and must receive and store those broadcast from other routers; (3) based on the state of the whole network, a router must compute the shortest paths from itself to all other known destinations, and use those to define the corresponding forwarding-table entries.

You may assume that the network is perfectly reliable and immutable. So, your router does not have to periodically resend its link-state advertisement once that is complete.

Implementation and Submission

You must submit a single *tar.gz* or *zip* archive containing your Java source files plus a file called *README.txt*. Do not include any other file or folder. You may use the IDE of your choice, but do not include their project files and folders. The *README.txt* file should contain a brief description of your implementation, possibly a list of limitations or errors you are aware of but that you were not able to fix, and clear references to any and all external material you might have used, including discussions with or help from other students.

Do not include any other file or folder. In particular, you may use the text editor or the IDE of your choice, but do not include project files and folders. Also, make sure your solution does not depend on any non-standard library, and that it compiles cleanly with the standard command-line compiler. Name your archive file following this format: *assign03-lastname-firstname.tar.gz*. Submit the *tar.gz* or *zip* archive through the iCorsi system.