Programming Assignment: COEN 337

This programming assignment focuses on socket communication between processes to achieve network routing. The assignment is described in two parts to simplify the process. All programming must be done in C/C++. No exceptions.

Part two:
In this part of the assignment you will simulate a network. You will need to use the code modules you wrote in part one of this project to send and receive messages. You will start independent and separate processes to represent each node in a network. Each node will read a file, *neighbor.config,* to determine its neighbors. The nodes then periodically exchange routing information with their neighbors and run Bellman Ford algorithm to compute a routing table. The routing table should be printed with a timestamp whenever there is a change in the table. Test your program with a variety of network configurations and double-check your computed routing table against what you manually calculate. Bring nodes down by killing the process for the nodes and check that the routing information propagates correctly. Likewise, make sure that when you bring a node back up that the routing information propagates correctly.

When your program is demonstrated, I will provide two configuration files to get things going. Be prepared for nodes from configuration file to die and come up again randomly.

Deliverables: This part of the assignment is due on: _____

1. A hard copy of all your program files and an emailed tar.gz/zip of all of your code. Include the output for at least one 5-node network with a dump of the routing tables. Demonstrate that the routing converges when a node is brought down and up again.
2. A hard copy of the output of your program generated by the config file I provide.
3. In one design document provide the following information:
    a. Documentation depicting your packet formats.
    b. Describe your design decisions and the consequent limitations. For example, did you use linked-lists or arrays, or what is the maximum number of neighbors that you can handle, or are the values of parameters (e.g., infinity) in the code configurable.
    c. Describe how you handle the "counting to infinity" problem.

4. Grading: Document 20, Compile 5, Execution 15

Simplified RIP Version 1 packet format: