CS 542 Project Description

Professor: Dr. Michael Y. Choi

TAs: Zhen Shao & Sughosh Divanji

Introduction

CS542 project is to simulate Link-state routing algorithm. Completing the project requires at least appropriate CS400 level C/C++ or JAVA programming skills under Windows 8 / 7 / XP OS GUI platform (like at Main/Rice campus PC lab) or command prompt mode (former MS-DOS or command line mode of Windows 7/XP platform). If you want to work it under a different platform such as MAC, Linux OS or UNIX OS, please get permission from your TA before you start. If you want to choose different programming languages such as MS Visual Basic or others, please also get permission from your TA in advance before you start.

Your project package MUST provide executable file in case of using a compiler so that it is not needed to compile again. If you select to use Java, compiled Java bytecodes must be included in the submission package and specify the detail instructions how to run your project or include JAR executable file.

Academic Honesty

Do not share your work with other peer. Your TAs have special tools to cross check potential cheating and have all history of similar works. If find a situation, both provider and requestor names will be placed on department chair and dean's desk.

You have to make sure your project MUST be executable at least on the PCs at the Rice campus or Main campus PC lab. Your TA will schedule everyone in Live and local online students to demonstrate their project. You have to be flexible to meet your TA's schedule. Bangalore and out of state people will be excluded for demonstration and your TA expects easy testing procedure. If not, those should schedule a remote demonstration to your TA at any time your TAs schedule.

This project can be done by an individual or a group of members up to 4 with different workload and requirements. A team project must be implemented by GUI. If any similar parts in the codes, customized program behavior, or any are founded, the work will be treated as cheating. It will cause failure of this course for both the helper (original work) and the requestor (duplicated or modified work). **No excuse, no exception.**

Objective

In this semester project, you are asked to develop a simulator to implement Link-State Routing Protocol. Your program should have two functions:

1) Simulate the process of generating connection table for each router in a given network,
2) Compute optimal path with least cost between any two specific routers.
    * A team project will require more functions from your TA when you reports.

Problem Description

Suppose we have a network with arbitrary number of routers. The network topology is given by a matrix, called the original topology (graph) matrix, which only indicates the costs of links between all directly connected routers. We assume each router only knows its own information and has no knowledge about others at the beginning.

In this project, to implement Link-State Routing Protocol, first your program is required to create the state of the links by each router after the input file containing the network information been loaded. By reading the topology matrix file a network graph can be determined. A ***Dijkstra's algorithm*** could be applied to find shortest path between two entities: source and destination nodes. Finally, your program should be able to output the connection table of any router, and output the optimal path between any two selected routers.

Sample Input

1) Sample Network Topology Diagram:
    - Each link between the two nodes may have different costs by direction
        (means your project should work any different topology/matrix)
    e.g)   Cost from R1 to R3 is 5 and from R3 to R1 is 3 respectively
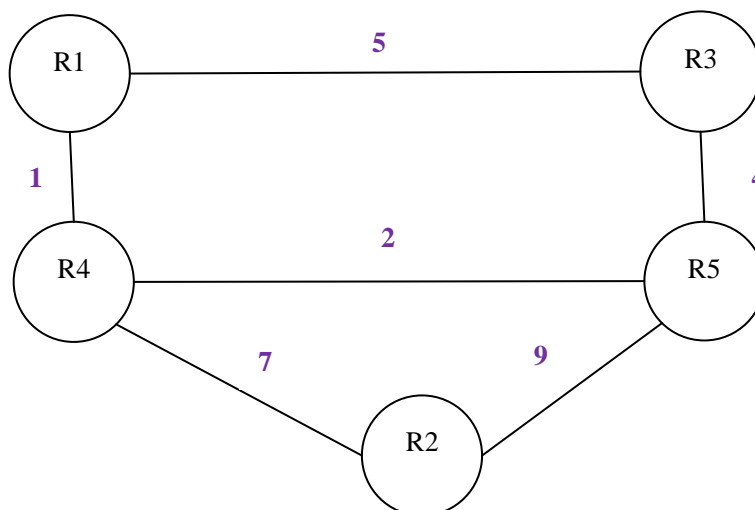   Cost between R1 and R4: both 1

Figure: A sample network topology with costs on links

2)   Sample Original Topology Matrix

Based on the above network topology diagram which is an undirected graph, we set the cost between a router and itself to 0, the cost between two indirectly connected routers to -1. Value -1 is equal meaning of infinite value (∞) as link state routing algorithm discussed in the class.   Then we have the following original topology matrix:

	R1	R2	R3	R4	R5
R1	0	-1	5	1	-1
R2	-1	0	-1	7	9
R3	5	-1	0	-1	4
R4	1	7	-1	0	2
R5	-1	9	4	2	0

Table: A sample original topology matrix

3)   Sample Input File (topology.txt)

The above table will be given as an input in a text file form named ***topology.txt***. Please notice that this is only a sample table and your program is required to be able to process a network with arbitrary number of routers and costs.

Expected Program Menu

Program prompts in blue. *User input in red.*

After start to run your program, the following menu is expected to prompt (but the menu and user interface are free design, no need to follow below way, even GUI is also welcome):
* A team project expects GUI design.

> **CS542 Link State Routing Simulator**
>
> **(1) Create a Network Topology**
> **(2) Build a Connection Table**
> **(3) Shortest Path to Destination Router**
> **(4) Modify a topology**
> **(5) Exit**
>
> **Command:**

Master Command: *1*

Prompt: Input original network topology matix data file:

Input: ***input.txt***

Prompt:

Review original topology matrix:

0   -1   5   1   -1

-1   0   -1   7   9

5   -1   0   -1   4

1   7   -1   0   2

-1   9   4   2   0

(Print matrix table and automatically return to main menu)

Master Command: *2*

Prompt: Select a source router:

Source router: *1*

Prompt:

   Router 1 Connection Table

Destination   Interface

=====================

    1   -

    2   4

    3   3

    4   4

    5   4

  (Print a connection table of router 1 and automatically return to main menu)

Master Command: *3*

Prompt: Select the destination router:

Destination Router? *5*

Prompt: The shortest path from router 1 to router 5 is 1-?-…-?-5, the total cost is (value)?

(Automatically exit to main menu)

Master Command: *4*

Prompt: Select a router to be removed:

Down router?   *3*   *// means router 3 becomes down*

(Result) Show an updated Connection table and a revised shortest path if source and destination path was chosen by former last selection.   If menu 3 has not done yet, find what intelligent way to proceed

Master Command: *5*

Prompt: Exit CS542 project.   Good Bye!

(Terminates the program)

Deliverables and Grading Policy
1. Well commented source code (5%)
2. Design and test report. The design of your program, what you did to convince yourself that the program works correctly, briefly explain link-state routing protocol. The test report for input files must be given in the project report. Also, you are required to specify at least five instances (5 routers at least, ×N times of routers for team project, N is number of members) you developed additional and report the corresponding results. TA will test your program with another file (in the same format as network.txt) which won't be released to you until the demo.   A PPT (Power point) presentation file with screen shots how your project works and link state routing protocol.   The maximum numbers of slides are 15 slides for an individual project.   A team project PPT should be more. (20%)
3. Detailed description of your algorithm applied to find the shortest path between two selected routers. (10%)
4. Detailed instructions how to compile and run your program.   (5%)
5. Results output by your program during demo. (30%)
6. TA will test your program with his own data file not released to the students. Your program should be able to process the input matrix regardless of its size when you show a demo. (30%)
- The above point allocation can be changed by TAs per their preference

Submission Guidelines

Due date: 11/21/2014 Saturday 23:59 US Central Time

- **NO** late submissions will be accepted in any reason

- An individual or team demo is required to your TA (will be scheduled later)

- A maximum of **four** students per team, you may work individually if you like

- The team work should be superior to any individual work

- Your TA will ask to test more different scenarios for team project

- If you choose a team project, you must provide team member list to your TA no later than 10/20/15 and should receive a confirmation from your TA

- **A team project will get additional feature implementation requirements from your TA**

- Once you choose a team project, that team cannot change team members and must stay in the team -- **No return to individual project again from a team.**

- Only one submission per team (or individual)

- The instructor and TA are not responsible for debugging your code

- Your submission should consist of four items only:
  - ➢ a project operations manual
  - ➢ the source code of your program
  - ➢ PPT file
  - ➢ executable program
- All items compress into one zip file format (not any other compression format at this time, no RAR format)

- filename format: **CS542Project_sec#_Lastname_Firstname.zip**
  - ➢ If a team project, use head of the team member as a filename but must list all team member names and other information in the separate file – team project requirement will be provided later
  - ➢ In a project operations manual, list who does what role
  - ➢ **E.g.   CS542TeamProject_sec#_Lastname_Firstname.zip**
    - o A team can be formed by cross section so section number is team leader section

- Submission to BB only and **only one submission**

- The BB, at this purpose, will not accept multiple submissions and your TA will NOT accept more than a single submission -- So you should be careful to select your submission

- If you make any mistake, no bridge to return. However, if you have a technical issue, contact your TAs but you must prove technical error.   If you submit at the last minute such as 10 min before the submission due and cannot submit it, it will be considered as late submission

- Must avoid last minute submission to BB.   With the excessive submissions from other courses, you may have long delay and it will cause late submission – strongly suggest to submit at least six hours early

- Please put down your name (Last, First, Middle) student ID and section ID you are enrolled in (section number or simply main campus, Internet, Bangalore) on the front page of the project report   -- Your name must be the same as registered name and if not, a penalty will be applied

- If you have any project related questions/doubts, please contact both TAs at the same time such as CC in email so both TAs understand what you ask and how they answer at the same page