Homework 9, Due Date: 10:00 am 04/17/2014,
Submission: Upload all source code (.java) files to Blackboard, and java programs (.class files) to EC2 instances

Write a program to build up the forwarding table at **router V_1** using the Dijkstra's algorithm.

- The routers in the network are labeled as V1, V2, V3, …, etc
- Your routing program needs to
    1. Display a prompt message to ask the user to input the total number of routers, **n**, in the network. Validate **n** to make sure that it is greater than or equal to 2.
    2. Display a prompt message to ask the user to input the name of a txt file that contains costs of all links. Each line in this file provides the cost between a pair of routers as below, where tab ('\t') is used to separate the numbers in each line.

       index of one router   index of the other router   link cost between these two routers
       …   …   …

       where the first and the second numbers in each row need to be validated to be between 1 and **n**, the third number needs to be validated to be positive. This txt file can locate in the same directory where this program runs such that a path is not needed. Display a message saying that in which row the first invalid number is detected, close the txt file, and KEEP asking for the name of the cost input file until all numbers are checked to be valid. Record the cost information in the Cost matrix.
    3. Implement the Dijsktra's algorithm to build up the shortest-path tree rooted at source router V_1. As the intermediate results, at the end of **Initialization** and each iteration of the **Loop**, display

       The set N'
       The set Y'
       D(i) for each i between 2 and **n**
       p(i) for each i between 2 and **n**
    4. Use the shortest-path tree resulted from the Dijsktra's algorithm to build up the forwarding table for router V_1. Display the forwarding table in the following format:

Destination	Link
V2	(V1, …)
V3	(V1, …)
…	
Vn	(V1, …)