

## ANC Assessed Exercise

Robert Allison 1102085a

### **User Manual:**

The program is supplied as a .jar file, with all the java files built inside it. It should be run on the command line with the latest JVM installed. The JDK and JRE It can be obtained from <http://www.oracle.com/technetwork/java/javase/downloads/index.html>. Networks are supplied as text files in the format:

```
{N1,N2,N3,N4}
(N1,N2,3)
(N2,N3,1)
(N3,N4,7)
(N4,N1,5)
```

Where the first line is the list of nodes in the network, and the following lines are edges of the form (Node 1, Node 2, Cost). For ease of parsing, ensure there are NO SPACES after the commas in these lists.

If you place a text file network in the same directory as the jar (like the sample networks provided), you can run the program by typing “java -jar DVRouting.jar testnetwork” where 'testnetwork' is the name of the text file. However, you will likely see there is little output, besides the building of the graph. To get the most out of this program, you will need OPERATORS.

DVRouting.jar takes the following operators:

- help displays how to use these operators with context.
- **-s/-e #** : These two operators cannot be used together. -s runs the graph until all the nodes are 'stable' ie. they have not changed between exchanges. -e is followed by a number, and will run the program for the selected number of exchanges.  
Ex: java -jar DVRouting.jar testnetwork -e 10 – runs for 10 exchanges
- **-v N1**: This operator lets you choose a node to output. Supply a node label after this, and the program will print its routing table for every exchange.  
Ex: java -jar DVRouting.jar testnetwork -s -v N1 – runs until stable, displays table for 'N1'
- **-t N1,N2,#**: This operator lets you choose an exchange, and find the best route between two nodes for that exchange. Example: java -jar DVRouting.jar testnetwork -e 10 -t N2,N3,4  
- This runs the network for 10 exchanges, and prints the best route between Node N2 and Node N3 on exchange 4.
- **-f N1,N2,#**: This operator lets you choose a link to fail on a specific exchange. You must supply the two nodes you wish the link between to fail, and the exchange it will happen on.  
Ex: java -jar DVRouting.jar testnetwork -e 10 -f N2,N3,4 – Runs for 10 exchanges, fails the link between N2 and N3 (if it exists) on exchange 4
- **-c N1,N2,C,#**: Similar to -f, except that you must supply a cost after the two nodes. On the selected exchange, the program will change the cost of the selected link on the chosen exchange. Ex: java -jar DVRouting.jar testnetwork -e 10 -c N2,N3,15, 3 – Runs for 10 exchanges, changes link cost between Node N2 and Node N3 to 15 on exchange 3
- **-Split**: Turns on split horizon mode.

These operators can be combined for multiple effects, and can be in any order, for example:

```
java -jar DVRouting.jar testnetwork -e 10 -v N1 -f N2,N3,4 -t N1,N2,5 - split
```

**The first argument supplied after DVRouting.jar MUST be an input file!**