**Distance Vector Routing Protocol**

**Contributors:**

**Jeevitha Mahankali (800966168)**

**Raghava Adarsh Mandarapu (800937296)**

**Design and Implementation:** In this project, we used two classes DVRServer and DVRClient using Java. DVRServer will take care of reading the given directory path to calculate the number of files present in it and to accept the UDP port numbers for each file. DVRClient will take care of creating the routers for each file, calculating the path from one router to another, detect changes in the router(cost) and re-calculate the path.

The tasks implemented by the classes DVRServer and DVRClient are:

**DVRServer:**

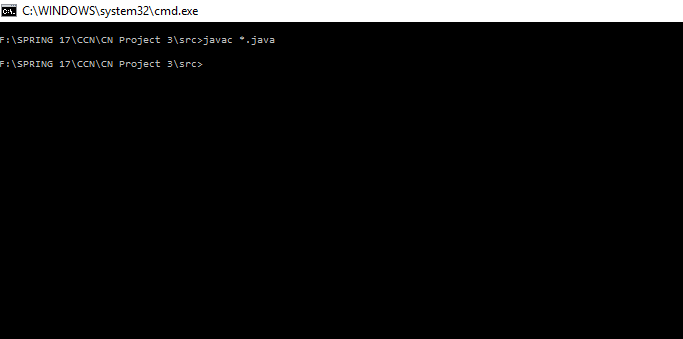
1. Validating the given directory.
2. Checking the number of files present in the directory.
3. Assigning UDP port numbers for each file.
4. Created processes with following arguments, corresponding id, parent path, number of routers and filename along with port number(E.g., A:1234 B:2345 C:4567) for each file and pass control to DVRClient.

**DVRClient**:

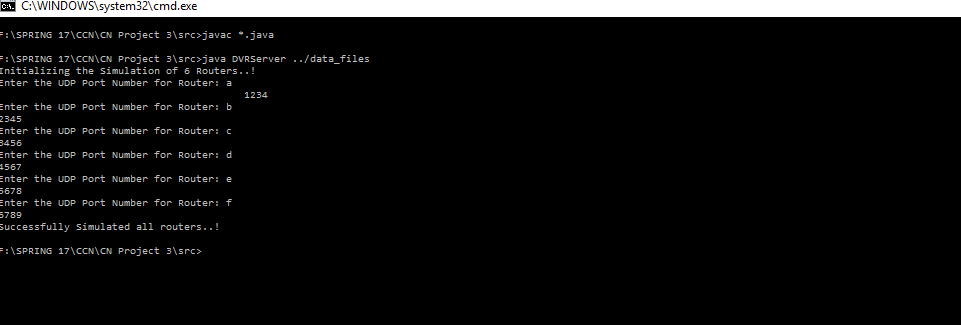
1. Calculates the number of routers based on the information given by DVRServer and creates the corresponding matrix.
2. Initialize all the local vectors i.e. the local cost function, hop information, neighbors list, corresponding file and the socket.
3. A new thread to handle the reading from network is created and detached from the main line.
4. When a vector is appeared in network this thread updates the corresponding vector in the global network vector matrix.
5. The thread performs the computations and writes the vector into the network.
6. First this thread will be reading data from the corresponding file and displays the local vector and then it writes the vector to the network.
7. After writing it keeps waiting for 5 sec in order to receive the vectors from all of its neighbors.
8. After 5sec it starts computing using the distance vector algorithm. After completion it waits for 10 more seconds and starts the iteration.

**Implementation and Output:**

1. Compile the DVRServerg.java and DVRClient.java programs using javac \*.java

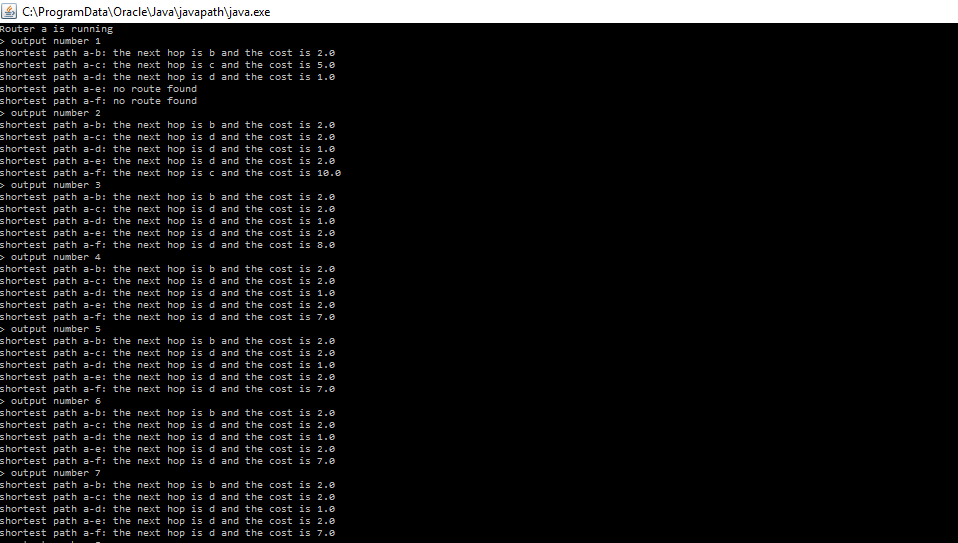


1. Execute DVRServer by providing the directory of files in the arguments (E.g**., java DVRServer ../data\_files**).

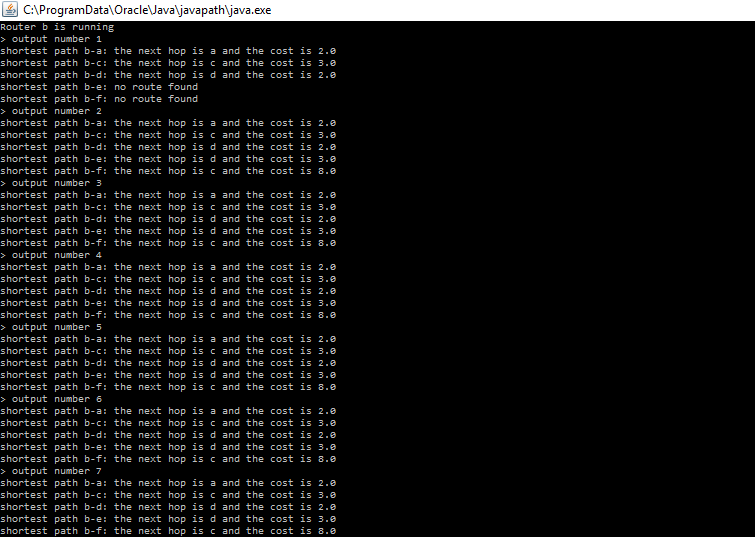


Output for each router is generated in the following manner:

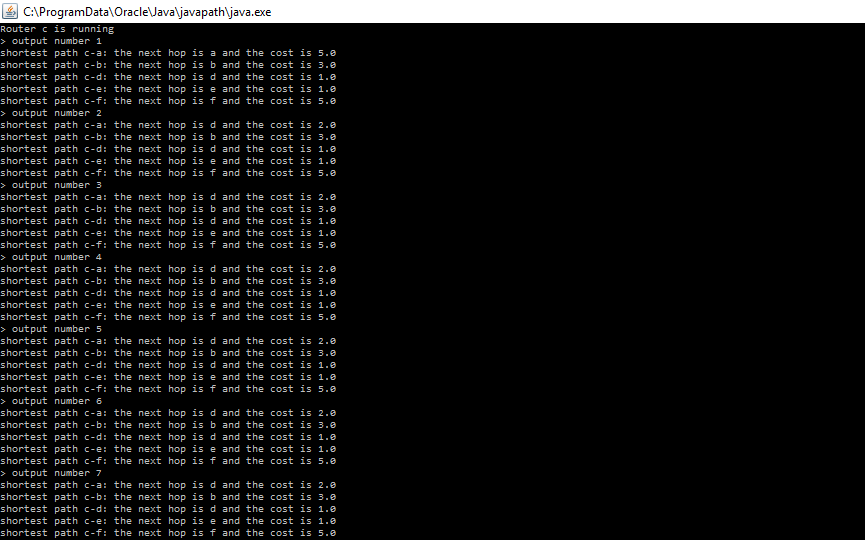
**Router a:**

****

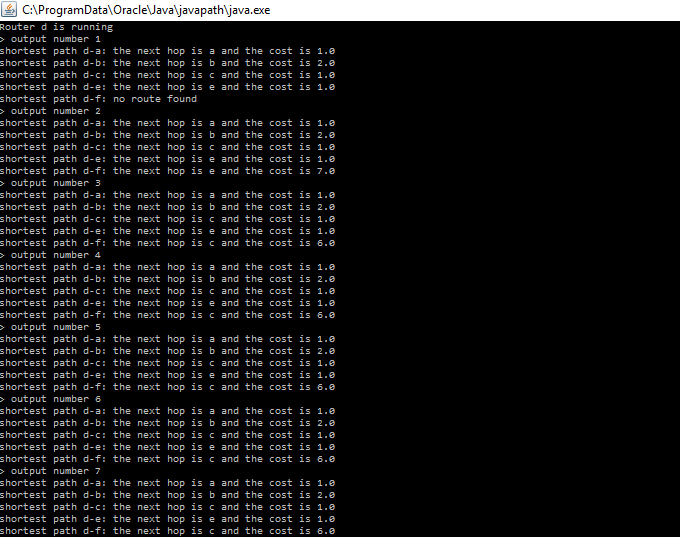
**Router b:**

****

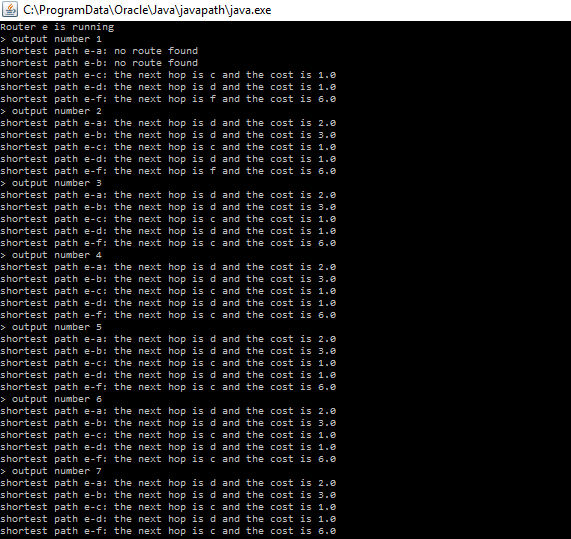
**Router c:**

****

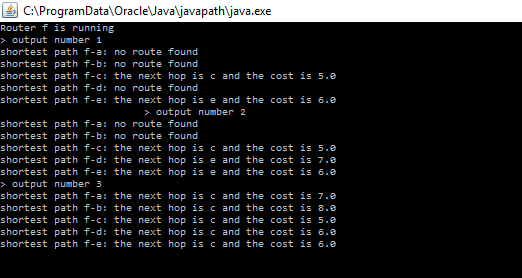
**Router d:**

****

**Router e:**

****

**Router f:**

****