README FILE

Prepared by:

Anuj Rastogi

UBPerson# 50134324

anujrast@buffalo.edu

- The program contains four files
  - anuj_makefile.GNUmakefile
  - anujrast_proj2.c
  - helperFunctions.c
  - network.h

- A topology file should always be present in the correct format. While preparing the topology file be sure that there are only required spaces and no extra spaces. The topology file should be present in the same directory as the program.

- To start the program type the command:
  - make –f anuj_makefile.GNUmakefile

- This will create an object file by the name “server”

- Post this give the following command to execute the program
  - ./server –t <TopologyFileName.txt> -i <timeIntervalsOfPackets>

  NOTE** Please ensure to provide the time intervals of more than 1 sec. The program cannot handle something less than 1 sec.

- **Please DO NOT JUST PRESS ENTER WITHOUT TYPING ANYTING. It will cause the program to crash.** Once the program starts running we can then run the following commands.
  - Display
    See the routing table

  - Packets
    See the number of packets received by you

  - Step
    Send forceful updates to neighbors

  - Update <serverID1><ServerID2><cost>
    To update the link between two neighbors to a new cost.

- The following commands are NOT supported
  - Update<serverID1><ServerID2> inf (the functionality is implemented but commented)

- Crash
- Disable<serverID> (the functionality is implemented but commented)

- The functionality for the following commands is implemented but commented.

    - Update<serverID1><ServerID2>inf (See lines 570 to 702 in file anujrast_proj2.c)
    - Disable<serverID> (See lines 796 to 870in file anujrast_proj2.c)

- For a detailed description, see the report attached with the project.
- Also, the below section also describes some key points about the report:

Implementation of routing table and Information table

There are two tables that are maintained in the program. These tables are maintained in the form of "**struct**" namely "**NetworkInfo**" and **"RoutingTable".**

- For NetworkInfo struct we have an array of NetworkInfo struct. This struct stores the details of the servers present in the network.

```
struct NetworkInfo
{
int ServerIDs;
char IP[100];
char port[20];
};
```

- For RoutingTable we have an array of struct RoutingTable which maintains the details of routing information for that server. This looks like:

```
struct RoutingTable
{
int SourceId; //Id of the destination server
int NId; //Id of the destination
char NIP[100]; //IP of the neighbour
char NPort[20]; //Port of the neighbour
int IntermediateNode; //Next hop ID
int IsNeighbour; //Flag to identify if the NId is the neighbour
int cost; // stores the cost from SourceId to NId
int IsDisabled; //Flag for link status between SourceId and NId
int Existed; //Flag to if the Destination ever existed as NId
};
```

The implementation for this is present in the “anujrast_proj2.c” file. Lines 3 to 21.

NOTE**: The tables are filled initially by the File that is read.