**MNC PA3 Report**
**Swapnesh Gandhi**
**50096836**

**Data structure used**: defined in server.h line 10

```
      //Host info
struct host_info{
      int id;     //Id of the host int topology file
      char server_ip[INET_ADDRSTRLEN];    //Ip address in xxx.xxx.xx.xx form
      uint16_t port;                      //server port
      bool neighbor;                      //true if host is neighbor of this server
      time_t start_time;                  // time at which update received last
                                          //i.e. timer start time
      int nextHop;                        //next Hop to reach this host
};
```

This is the host info data structure I used for maintaining the data about each host.
I also used a matrix to store the distance vector table at each host, updated this matrix on either timeout occurrence or specific commands on STDIN.

**Asynchronous I/O:** I used epoll interface for keeping track of the file descriptors like in PA1, I used the timeout value in the epoll_wait function to wait for update_interval time. Each time epoll_wait return or unblocks I check current time and detect timeout event.
For detecting individual timeouts I check start_time value with current time-stamp if It's greater than 3*update_interval then I assume the server has crashed.

**Message exchange:** I have used variable types like uint16_t etc. as they are guaranteed to be 2 bytes.
//file server_helper.cpp line 440

```
uint16_t NumUpdateFields;
uint16_t ServerPort;
uint32_t ServerIP;
uint16_t Filler=0;
uint16_t Id;
uint16_t Cost;

further, I copy these fields using memcpy as follows.
 memcpy(tmpptr,&NumUpdateFields,sizeof(NumUpdateFields));
```

**Important Functions:**

```
void UpdateCosts(int idx); //used to update Cost Matrix after each new message
                           //file server_helper.cpp line 810.
void sendDistanceVec();    //used to send distance vector to each neighbor
                           //file server_helper.cpp line 437
void WaitforEvent(int eventfd, int serverSock, int timeout);
                             //waits for an event to occur and takes the
                             //appropriate action file server_helper.cpp line 381
void detect_neighbor_timeout(); //dectects whether 3 intervals have passed since
                             //last message from a neighbor
                             //file server_helper.cpp line 341
void ExtractfromTopo(const char* topo_file); //Extracts data from topology file
                             //file server_helper.cpp line 117

I have tried to include details about other functions with the comments in the
each cpp file.
```