

**Programming Assignment #4**  
**CSCE 5580 - Computer Networks**  
**Spring 2018**  
**100 Points**  
**Due: 04/30/2018, 11:59 PM**

**Instructions:**

- i). Compile the C programs and make sure it's working.
- ii). Comment your code.
- iii). Create a readme file text file that describes the working and usage of the code.
- iv). Please create a zip archive of your assignment folder (readme, code, and header files) and upload the zip file.
- v). Not following the above instructions could result up to 20% deduction from your assignment score.

**Objective:**

Demonstrate working of DHCP server using a client-server architecture.

**Requirements:**

1. Create a C-based client-server architecture using a UDP socket.
2. The DHCP server should be able to receive message from multiple clients.
3. The server should run on cse01.cse.unt.edu machine and the client can run on any CSE machines.
4. Demonstrate client requesting an IP address and DHCP server responding by offering three available IP addresses.
5. The client should randomly pick an IP address among the three offered IP addresses and send a DHCP request to the server.
6. The DHCP server should register the client selected IP address is taken and should acknowledge the client.

7. The client should randomly generate the transaction ID.
8. The lifetime should be set to 3600 seconds.

**Procedure:**

1. Create a C-based server that can accept data from multiple clients using UDP socket.
2. Make sure the server runs on cse01.cse.unt.edu and the format to start the server is as follows

```
dhcp <port_number>
```

```
gateway:
```

```
subnet_mask:
```

where `dhcp` is the UDP server executable and `port_number` is the port number on which the UDP server can accept data. The `gateway` is the IP address from which the DHCP server starts offering the IP addresses to the clients. See Procedure #6. The `subnet_mask` is the subnet mask associated with the IP addresses.

3. Create a C-based client that can communicate with the server using UDP sockets.
4. The client can run on any CSE machine, and sends and receives messages to the server as follows

```
client <port_number>
```

where `client` is the client executable, `port_number` is the port number on which the client sends messages to the DHCP server

5. The client tries to request an IP address from the DHCP server by contacting the server, and sending a packet with `yiaddr: 0.0.0.0` and a randomly generated transaction ID.
6. The DHCP server responds with three IP address offers, transaction ID, and a lifetime (3600 seconds). The offer IP addresses starts from the next available IP address. For example, if the gateway is 192.168.1.1 and the subnet mask is 255.255.255.0, then the available IP addresses are between the range 192.168.1.2 – 192.168.1.255
7. The client responds with a DHCP request by randomly choosing an IP address from the three addresses being offered and a new transaction ID (old transaction ID + 1).

8. The DHCP server responds with an ACK confirming the chosen IP address.
9. The DHCP server registers that the IP address is taken
10. Print all the communication between the DHCP server and the client on the terminal console.
11. A structure can be used to store the IP addresses, transaction ID, and the lifetime. The structure can be used for all the communications.
12. An example DHCP transaction is shown below in Figure 1.
13. Test the DHCP server and client to make sure the demonstrations are working.

**Deliverables:**

1. Commented DHCP server and client C code
2. A readme text-file that describes how to compile, execute, and test the C codes.

