Program:

```
#include<iostream>
#include <fstream>
#include <iomanip>
#include <string>
using namespace std;

int main()
{
	int count,i,choice;

	string no,time,source,destination,protocol,length,info;
	string protocolChoice;

	do{
		ifstream file("data.csv");
		count = -1;
		i=0;
		cout<<"\nEnter which protocol packets you want to see"<<endl;
		cout<<"1.IP\n2.UDP\n3.TCP\n4.Ethernet\n0.Exit!!!"<<endl;
		cin>>choice;

		switch(choice){
			case 1: protocolChoice="\"ICMPv6\"";
			break;
			case 2: protocolChoice="\"UDP\"";
			break;
			case 3: protocolChoice="\"TCP\"";
			break;
			case 4: protocolChoice="\"ARP\"";
			break;
```

```
                }

                while(file.good()){
                        getline(file,no,',');
                        getline(file,time,',');
                        getline(file,source,',');
                        getline(file,destination,',');
                        getline(file,protocol,',');
                        getline(file,length,',');
                        getline(file,info,'\n');

                        if(protocol == protocolChoice || protocol == "Protocol"){
                                cout<<setw(4)<<left<<i++;
                                cout<<setw(12)<<left<<string(time,1,time.length()-2);
                                cout<<setw(30)<<left<<string(source,1,source.length()-2);
                                cout<<setw(30)<<left<<string(destination,1,destination.length()-2);
                                cout<<setw(8)<<left<<string(protocol,1,protocol.length()-2);
                                cout<<setw(8)<<left<<string(length,1,length.length()-2);
                                cout<<string(info,1,info.length()-2);
                                cout<<endl;
                                count++;
                        }
                }
                file.close();
        }while(choice!=0);
        return 0;
}
```

****

Output:

~$ g++ a.cpp

~$ ./a.out

Enter which protocol packets you want to see

1.IP

2.UDP

3.TCP

4.Ethernet

0.Exit!!!

3

0 2.151479000 47.74.170.156 10.10.13.46 TCP 78 [TCP segment of a reassembled PDU]

1 2.991293000 47.74.170.156 10.10.13.46 TCP 78 [TCP Retransmission] xmpp-client > 48590 [PSH, ACK] Seq=1 Ack=1 Win=105 Len=12 TSval=341833507 TSecr=10160046

2 4.670899000 47.74.170.156 10.10.13.46 TCP 78 [TCP Retransmission] xmpp-client > 48590 [PSH, ACK] Seq=1 Ack=1 Win=105 Len=12 TSval=341835187 TSecr=10160046

3 6.592677000 52.66.57.45 10.10.12.159 TCP 78 [TCP segment of a reassembled PDU]

4 6.938253000 52.66.57.45 10.10.12.159 TCP 78 [TCP Retransmission] xmpp-client > 51522 [PSH, ACK] Seq=1 Ack=1 Win=200 Len=12 TSval=1273344608 TSecr=20845088

5 7.095293000 172.217.166.74 10.10.15.5 TCP 60 https > 34447 [RST] Seq=1 Win=65535 Len=0

Enter which protocol packets you want to see

1.IP

2.UDP

3.TCP

4.Ethernet

0.Exit!!!

0

****

Dhruvil Shah 047

F18111051 TE Comp1

Assignment 7 (A)

Q1 Give a list of packet analyzer tool.

Ans
- Wireshark
- tcpdump
- NetworkMiner
- Fiddler
- Windump
- Capsa
- OmniPeek
- Etherape.

Q2 Explain wireshark in detail.

Ans Wireshark is a free and open source packet analyzer. It is used for network troubleshooting, analysis, software and communications protocol development and education. Originally named ~~Ether~~ Ethereal. Wireshark is a cross platform using the Qt widget toolkit in current release to implement its user interface ~~as~~ and using pcap to capture packets. Wireshark lets the user put network interface controllers into promiscous mode so they can see all the traffic visible on that interface including unicast traffic not sent to that interface.

Q3 Explain steps of installation of packet analyzer tool for ubuntu.

Ans Add the universe repository using the command :
$ sudo add-apt - repository universe.

Install wireshark using the command :
$ sudo apt install wireshark.

Enter the sudo password

Check the wireshark version using the command
$ wireshark — version.

Launch wireshark and start using it.