Department of Computer Science and Engineering

Islamic University of Technology (IUT)

A subsidiary organ of OIC

Laboratory Report

CSE 4512: Computer Networks Lab

Name: Md. Abdullah Al Jubaer Gem
Student ID: 210041226
Section: 2B
Semester: 5th
Academic Year: 2023-2024

Date of Submission: 23-02-2025

Title: Network Topology Implementation using Point-to-Point Channel in ns3

Objective:

1. Discuss the concept of networking in ns3
2. Identify the key components of a network in ns3
3. Creating topologies in ns3

Devices/ Software Used:

1. ns3

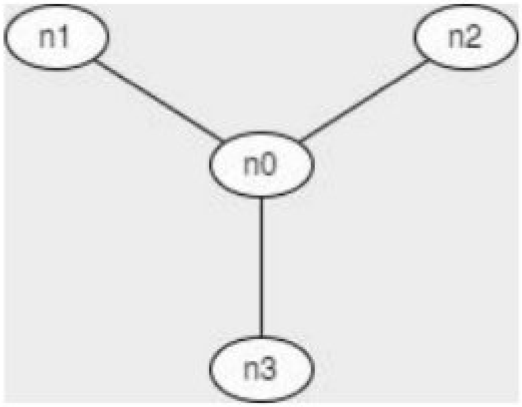
Theory:

Some of the terms used in networking in ns3 are nodes, channels, net device, node container, point to point helper, net device container, Internet Stack Helper, Ipv4 address helper, UdpEchoServerHelper, UdpEchoClientHelper etc. All of these are required to create a network in ns3. After adding all of these, we simulate in ns3 how the network should perform.

Diagram of the experiment:

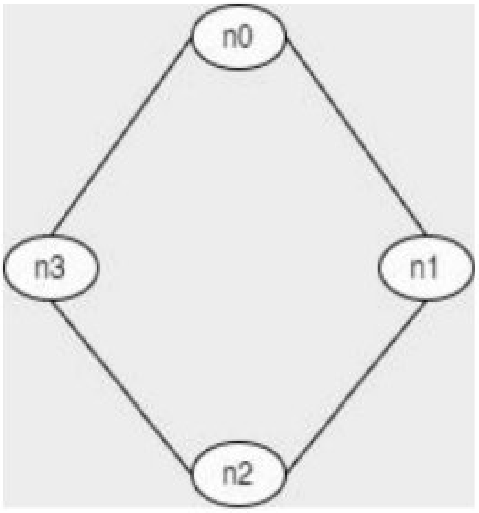
Task #01:

Star Topology with 4 nodes



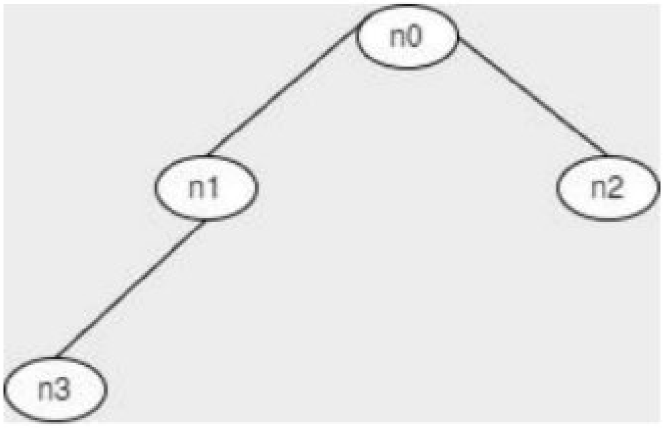
Task #02:

Ring Topology with 4 nodes



Task #03:

Tree Topology with 4 nodes



Working Procedure:

TASK #01:

1. Create a star-shaped network with a central node connecting to several leaf nodes using point-to-point links.
2. Create nodes, node containers, PointToPointHelper, NetDeviceContainer, InternetStackHelper, Ipv4AddressHelper, Ipv4InterfaceContainer, UdpEchoServerHelper and UdpEchoClientHelper for the network.
3. Run the simulator
4. After running, destroy the simulator.

The code under main function is given below:

```
int
main(int argc, char* argv[])
{
    CommandLine cmd(__FILE__);
    cmd.Parse(argc, argv);


    Time::SetResolution(Time::NS);
    LogComponentEnable("UdpEchoClientApplication", LOG_LEVEL_INFO);
    LogComponentEnable("UdpEchoServerApplication", LOG_LEVEL_INFO);


    NodeContainer nodes;
    nodes.Create(4);
    PointToPointHelper pointToPoint;
    pointToPoint.SetDeviceAttribute("DataRate", StringValue("5Mbps"));
    pointToPoint.SetChannelAttribute("Delay", StringValue("2ms"));


    NetDeviceContainer devices[3];
    devices[0] = pointToPoint.Install(nodes.Get(0), nodes.Get(1));
    devices[1] = pointToPoint.Install(nodes.Get(0), nodes.Get(2));
    devices[2] = pointToPoint.Install(nodes.Get(0), nodes.Get(3));


    InternetStackHelper stack;
    stack.Install(nodes);
    Ipv4AddressHelper address;


    Ipv4InterfaceContainer interfaces[3];
    address.SetBase("192.6.26.0", "255.255.255.0");
    interfaces[0] = address.Assign(devices[0]);
    address.SetBase("192.6.27.0", "255.255.255.0");
    interfaces[1] = address.Assign(devices[1]);
    address.SetBase("192.6.28.0", "255.255.255.0");
    interfaces[2] = address.Assign(devices[2]);


    UdpEchoServerHelper echoServer(9);


    for (int serverNode = 0; serverNode <= 3; serverNode++)
    {
        ApplicationContainer serverApps =
echoServer.Install(nodes.Get(serverNode));
        serverApps.Start(Seconds(1.0));
        serverApps.Stop(Seconds(40.0));


        if (serverNode == 0)
        {
            for (int clientNode = 1; clientNode <= 3; clientNode++)
            {
                UdpEchoClientHelper echoClient(interfaces[clientNode -
1].GetAddress(0), 9);
                echoClient.SetAttribute("MaxPackets", UintegerValue(1));
                echoClient.SetAttribute("Interval",
TimeValue(Seconds(1.0)));
                echoClient.SetAttribute("PacketSize",
UintegerValue(1024));


                ApplicationContainer clientApps =
echoClient.Install(nodes.Get(clientNode));
                clientApps.Start(Seconds(2.0 + (clientNode - 1) * 2));
                clientApps.Stop(Seconds(20.0));
            }
        }
        else if (serverNode == 1)
        {
            UdpEchoClientHelper echoClient1(interfaces[0].GetAddress(1),
9);
            echoClient1.SetAttribute("MaxPackets", UintegerValue(3));
            echoClient1.SetAttribute("Interval", TimeValue(Seconds(1.0)));
            echoClient1.SetAttribute("PacketSize", UintegerValue(1024));


            ApplicationContainer clientApps1 =
echoClient1.Install(nodes.Get(0));
            clientApps1.Start(Seconds(8.0));
            clientApps1.Stop(Seconds(20.0));


            UdpEchoClientHelper echoClient2(interfaces[1].GetAddress(0),
9);
            echoClient2.SetAttribute("MaxPackets", UintegerValue(3));
            echoClient2.SetAttribute("Interval", TimeValue(Seconds(1.0)));
            echoClient2.SetAttribute("PacketSize", UintegerValue(1024));


            ApplicationContainer clientApps2 =
echoClient2.Install(nodes.Get(2));
            clientApps2.Start(Seconds(10.0));
            clientApps2.Stop(Seconds(20.0));


            UdpEchoClientHelper echoClient3(interfaces[2].GetAddress(0),
9);
            echoClient3.SetAttribute("MaxPackets", UintegerValue(3));
            echoClient3.SetAttribute("Interval", TimeValue(Seconds(1.0)));
            echoClient3.SetAttribute("PacketSize", UintegerValue(1024));


            ApplicationContainer clientApps3 =
echoClient3.Install(nodes.Get(3));
            clientApps3.Start(Seconds(12.0));
            clientApps3.Stop(Seconds(20.0));
        }
        else if (serverNode == 2)
        {
            UdpEchoClientHelper echoClient1(interfaces[1].GetAddress(1),
9);
            echoClient1.SetAttribute("MaxPackets", UintegerValue(3));
            echoClient1.SetAttribute("Interval", TimeValue(Seconds(1.0)));
            echoClient1.SetAttribute("PacketSize", UintegerValue(1024));
            ApplicationContainer clientApps1 =
echoClient1.Install(nodes.Get(0));
            clientApps1.Start(Seconds(10.0));
            clientApps1.Stop(Seconds(40.0));


            UdpEchoClientHelper echoClient2(interfaces[0].GetAddress(0),
9);
            echoClient2.SetAttribute("MaxPackets", UintegerValue(3));
            echoClient2.SetAttribute("Interval", TimeValue(Seconds(1.0)));
            echoClient2.SetAttribute("PacketSize", UintegerValue(1024));
            ApplicationContainer clientApps2 =
echoClient2.Install(nodes.Get(1));
            clientApps2.Start(Seconds(12.0));
            clientApps2.Stop(Seconds(40.0));


            UdpEchoClientHelper echoClient3(interfaces[2].GetAddress(0),
9);
            echoClient3.SetAttribute("MaxPackets", UintegerValue(3));
            echoClient3.SetAttribute("Interval", TimeValue(Seconds(1.0)));
            echoClient3.SetAttribute("PacketSize", UintegerValue(1024));


            ApplicationContainer clientApps3 =
echoClient3.Install(nodes.Get(3));
            clientApps3.Start(Seconds(14.0));
            clientApps3.Stop(Seconds(40.0));
        }
        else if (serverNode == 3)
        {
            UdpEchoClientHelper echoClient1(interfaces[2].GetAddress(1),
9);
            echoClient1.SetAttribute("MaxPackets", UintegerValue(1));
            echoClient1.SetAttribute("Interval", TimeValue(Seconds(1.0)));
            echoClient1.SetAttribute("PacketSize", UintegerValue(1024));
            ApplicationContainer clientApps1 =
echoClient1.Install(nodes.Get(0));
            clientApps1.Start(Seconds(16.0));
            clientApps1.Stop(Seconds(40.0));


            UdpEchoClientHelper echoClient2(interfaces[0].GetAddress(0),
9);
```

```
            echoClient2.SetAttribute("MaxPackets", UintegerValue(3));
            echoClient2.SetAttribute("Interval", TimeValue(Seconds(1.0)));
            echoClient2.SetAttribute("PacketSize", UintegerValue(1024));
            ApplicationContainer clientApps2 =
echoClient2.Install(nodes.Get(1));
            clientApps2.Start(Seconds(18.0));
            clientApps2.Stop(Seconds(40.0));


            UdpEchoClientHelper echoClient3(interfaces[1].GetAddress(0),
9);
            echoClient3.SetAttribute("MaxPackets", UintegerValue(3));
            echoClient3.SetAttribute("Interval", TimeValue(Seconds(1.0)));
            echoClient3.SetAttribute("PacketSize", UintegerValue(1024));


            ApplicationContainer clientApps3 =
echoClient3.Install(nodes.Get(2));
            clientApps3.Start(Seconds(20.0));
            clientApps3.Stop(Seconds(40.0));
        }
    }


    AnimationInterface anim("Topology1.xml");
    Simulator::Run();
    Simulator::Destroy();
    return 0;
}
```

Output:

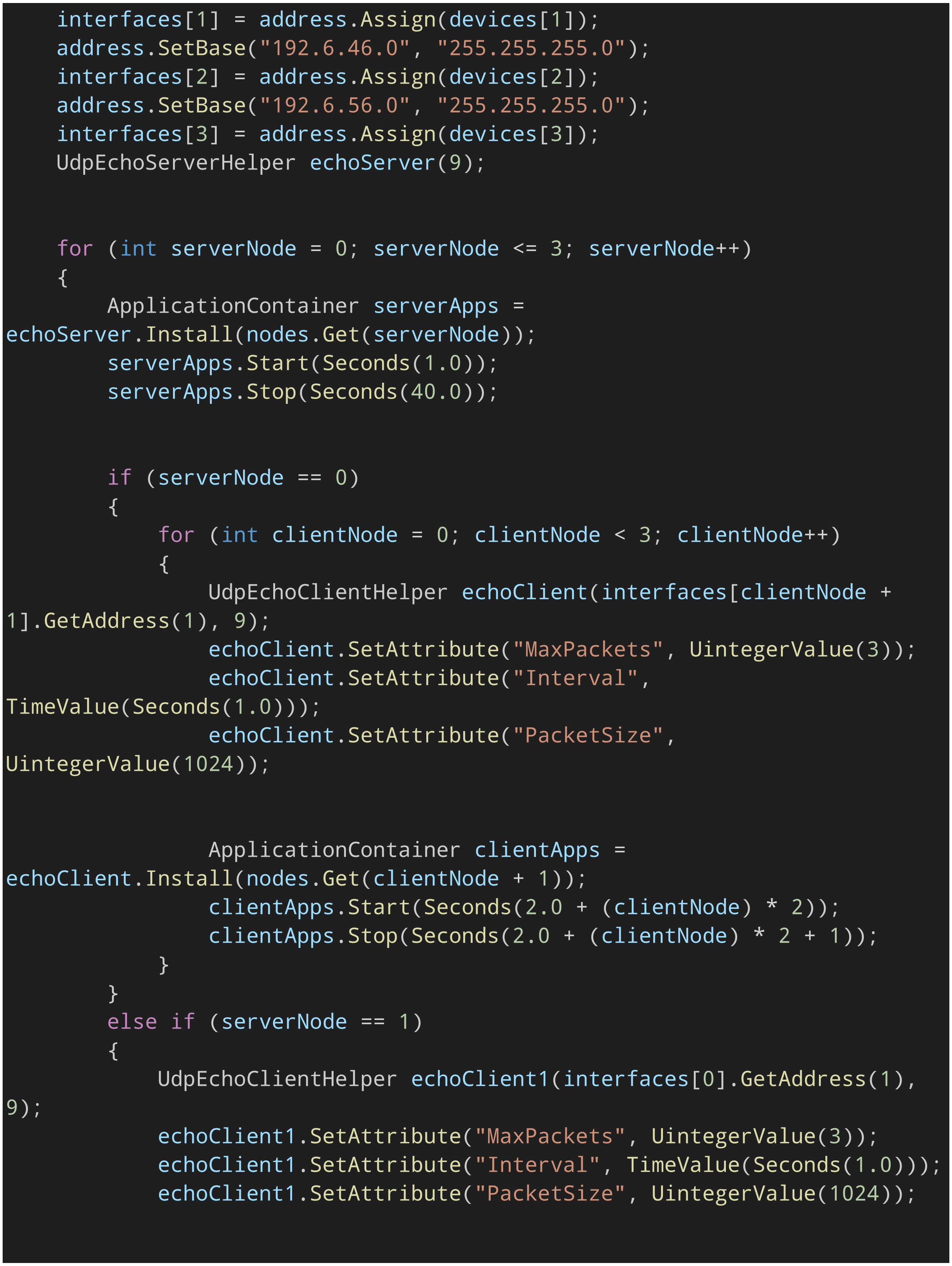
```
At time +2s client sent 1024 bytes to 192.6.26.1 port 9
At time +2.00369s server received 1024 bytes from 192.6.26.2 port 49153
At time +2.00369s server sent 1024 bytes to 192.6.26.2 port 49153
At time +2.00737s client received 1024 bytes from 192.6.26.1 port 9
At time +4s client sent 1024 bytes to 192.6.27.1 port 9
At time +4.00369s server received 1024 bytes from 192.6.27.2 port 49153
At time +4.00369s server sent 1024 bytes to 192.6.27.2 port 49153
At time +4.00737s client received 1024 bytes from 192.6.27.1 port 9
At time +6s client sent 1024 bytes to 192.6.28.1 port 9
At time +6.00369s server received 1024 bytes from 192.6.28.2 port 49153
At time +6.00369s server sent 1024 bytes to 192.6.28.2 port 49153
At time +6.00737s client received 1024 bytes from 192.6.28.1 port 9
At time +8s client sent 1024 bytes to 192.6.26.2 port 9
At time +8.00369s server received 1024 bytes from 192.6.26.1 port 49153
At time +8.00369s server sent 1024 bytes to 192.6.26.1 port 49153
At time +8.00737s client received 1024 bytes from 192.6.26.2 port 9
At time +9s client sent 1024 bytes to 192.6.26.2 port 9
At time +9.00369s server received 1024 bytes from 192.6.26.1 port 49153
At time +9.00369s server sent 1024 bytes to 192.6.26.1 port 49153
At time +9.00737s client received 1024 bytes from 192.6.26.2 port 9
At time +10s client sent 1024 bytes to 192.6.26.2 port 9
At time +10s client sent 1024 bytes to 192.6.27.2 port 9
At time +10s client sent 1024 bytes to 192.6.27.1 port 9
At time +10.0037s server received 1024 bytes from 192.6.26.1 port 49153
At time +10.0037s server sent 1024 bytes to 192.6.26.1 port 49153
At time +10.0037s server received 1024 bytes from 192.6.27.1 port 49154
At time +10.0037s server sent 1024 bytes to 192.6.27.1 port 49154
At time +10.0037s server received 1024 bytes from 192.6.27.2 port 49154
At time +10.0037s server sent 1024 bytes to 192.6.27.2 port 49154
At time +10.0074s client received 1024 bytes from 192.6.26.2 port 9
At time +10.0074s client received 1024 bytes from 192.6.27.2 port 9
At time +10.0074s client received 1024 bytes from 192.6.27.1 port 9
At time +11s client sent 1024 bytes to 192.6.27.2 port 9
At time +11s client sent 1024 bytes to 192.6.27.1 port 9
At time +11.0037s server received 1024 bytes from 192.6.27.1 port 49154
At time +11.0037s server sent 1024 bytes to 192.6.27.1 port 49154
At time +11.0037s server received 1024 bytes from 192.6.27.2 port 49154
At time +11.0037s server sent 1024 bytes to 192.6.27.2 port 49154
At time +11.0074s client received 1024 bytes from 192.6.27.2 port 9
At time +11.0074s client received 1024 bytes from 192.6.27.1 port 9
At time +12s client sent 1024 bytes to 192.6.27.2 port 9
At time +12s client sent 1024 bytes to 192.6.27.1 port 9
At time +12s client sent 1024 bytes to 192.6.26.1 port 9
At time +12s client sent 1024 bytes to 192.6.28.1 port 9
At time +12.0037s server received 1024 bytes from 192.6.27.1 port 49154
At time +12.0037s server sent 1024 bytes to 192.6.27.1 port 49154
At time +12.0037s server received 1024 bytes from 192.6.27.2 port 49154
At time +12.0037s server sent 1024 bytes to 192.6.27.2 port 49154
At time +12.0037s server received 1024 bytes from 192.6.26.2 port 49154
At time +12.0037s server sent 1024 bytes to 192.6.26.2 port 49154
At time +12.0037s server received 1024 bytes from 192.6.28.2 port 49154
At time +12.0037s server sent 1024 bytes to 192.6.28.2 port 49154
At time +12.0074s client received 1024 bytes from 192.6.27.2 port 9
At time +12.0074s client received 1024 bytes from 192.6.27.1 port 9
At time +12.0074s client received 1024 bytes from 192.6.26.1 port 9
At time +12.0074s client received 1024 bytes from 192.6.28.1 port 9
At time +13s client sent 1024 bytes to 192.6.26.1 port 9
At time +13s client sent 1024 bytes to 192.6.28.1 port 9
At time +13.0037s server received 1024 bytes from 192.6.26.2 port 49154
At time +13.0037s server sent 1024 bytes to 192.6.26.2 port 49154
At time +13.0037s server received 1024 bytes from 192.6.28.2 port 49154
At time +13.0037s server sent 1024 bytes to 192.6.28.2 port 49154
At time +13.0074s client received 1024 bytes from 192.6.26.1 port 9
At time +13.0074s client received 1024 bytes from 192.6.28.1 port 9
At time +14s client sent 1024 bytes to 192.6.26.1 port 9
At time +14s client sent 1024 bytes to 192.6.28.1 port 9
At time +14s client sent 1024 bytes to 192.6.28.1 port 9
At time +14.0037s server received 1024 bytes from 192.6.26.2 port 49154
At time +14.0037s server sent 1024 bytes to 192.6.26.2 port 49154
At time +14.0037s server received 1024 bytes from 192.6.28.2 port 49154
At time +14.0037s server sent 1024 bytes to 192.6.28.2 port 49154
At time +14.0054s server received 1024 bytes from 192.6.28.2 port 49155
At time +14.0054s server sent 1024 bytes to 192.6.28.2 port 49155
At time +14.0074s client received 1024 bytes from 192.6.26.1 port 9
At time +14.0074s client received 1024 bytes from 192.6.28.1 port 9
At time +14.0091s client received 1024 bytes from 192.6.28.1 port 9
At time +15s client sent 1024 bytes to 192.6.28.1 port 9
At time +15.0037s server received 1024 bytes from 192.6.28.2 port 49155
At time +15.0037s server sent 1024 bytes to 192.6.28.2 port 49155
At time +15.0074s client received 1024 bytes from 192.6.28.1 port 9
At time +16s client sent 1024 bytes to 192.6.28.1 port 9
At time +16s client sent 1024 bytes to 192.6.28.2 port 9
At time +16.0037s server received 1024 bytes from 192.6.28.2 port 49155
At time +16.0037s server sent 1024 bytes to 192.6.28.2 port 49155
At time +16.0037s server received 1024 bytes from 192.6.28.1 port 49155
At time +16.0037s server sent 1024 bytes to 192.6.28.1 port 49155
At time +16.0074s client received 1024 bytes from 192.6.28.1 port 9
At time +16.0074s client received 1024 bytes from 192.6.28.2 port 9
At time +18s client sent 1024 bytes to 192.6.26.1 port 9
At time +18.0037s server received 1024 bytes from 192.6.26.2 port 49155
At time +18.0037s server sent 1024 bytes to 192.6.26.2 port 49155
At time +18.0074s client received 1024 bytes from 192.6.26.1 port 9
At time +19s client sent 1024 bytes to 192.6.26.1 port 9
At time +19.0037s server received 1024 bytes from 192.6.26.2 port 49155
At time +19.0037s server sent 1024 bytes to 192.6.26.2 port 49155
At time +19.0074s client received 1024 bytes from 192.6.26.1 port 9
At time +20s client sent 1024 bytes to 192.6.26.1 port 9
At time +20s client sent 1024 bytes to 192.6.27.1 port 9
At time +20.0037s server received 1024 bytes from 192.6.26.2 port 49155
At time +20.0037s server sent 1024 bytes to 192.6.26.2 port 49155
At time +20.0037s server received 1024 bytes from 192.6.27.2 port 49155
At time +20.0037s server sent 1024 bytes to 192.6.27.2 port 49155
At time +20.0074s client received 1024 bytes from 192.6.26.1 port 9
At time +20.0074s client received 1024 bytes from 192.6.27.1 port 9
At time +21s client sent 1024 bytes to 192.6.27.1 port 9
At time +21.0037s server received 1024 bytes from 192.6.27.2 port 49155
At time +21.0037s server sent 1024 bytes to 192.6.27.2 port 49155
At time +21.0074s client received 1024 bytes from 192.6.27.1 port 9
At time +22s client sent 1024 bytes to 192.6.27.1 port 9
At time +22.0037s server received 1024 bytes from 192.6.27.2 port 49155
At time +22.0037s server sent 1024 bytes to 192.6.27.2 port 49155
At time +22.0074s client received 1024 bytes from 192.6.27.1 port 9
```

TASK #02:

1. Design a circular network where nodes are connected in a ring, and set up point-to-point links

between adjacent nodes.
2.Rest of the instructions are similar to task 1.
.
The code under main function is given below:

```
int
main(int argc, char* argv[])
{
    CommandLine cmd(__FILE__);
    cmd.Parse(argc, argv);


    Time::SetResolution(Time::NS);
    LogComponentEnable("UdpEchoClientApplication", LOG_LEVEL_INFO);
    LogComponentEnable("UdpEchoServerApplication", LOG_LEVEL_INFO);


    NodeContainer nodes;
    nodes.Create(4);
    PointToPointHelper pointToPoint;
    pointToPoint.SetDeviceAttribute("DataRate", StringValue("5Mbps"));
    pointToPoint.SetChannelAttribute("Delay", StringValue("2ms"));


    NetDeviceContainer devices[4];
    devices[0] = pointToPoint.Install(nodes.Get(0), nodes.Get(1));
    devices[1] = pointToPoint.Install(nodes.Get(1), nodes.Get(2));
    devices[2] = pointToPoint.Install(nodes.Get(2), nodes.Get(3));
    devices[3] = pointToPoint.Install(nodes.Get(3), nodes.Get(0));


    InternetStackHelper stack;
    stack.Install(nodes);


    Ipv4AddressHelper address;


    Ipv4InterfaceContainer interfaces[4];
    address.SetBase("192.6.26.0", "255.255.255.0");
    interfaces[0] = address.Assign(devices[0]);
    address.SetBase("192.6.36.0", "255.255.255.0");
```



```
    interfaces[1] = address.Assign(devices[1]);
    address.SetBase("192.6.46.0", "255.255.255.0");
    interfaces[2] = address.Assign(devices[2]);
    address.SetBase("192.6.56.0", "255.255.255.0");
    interfaces[3] = address.Assign(devices[3]);
    UdpEchoServerHelper echoServer(9);


    for (int serverNode = 0; serverNode <= 3; serverNode++)
    {
        ApplicationContainer serverApps =
echoServer.Install(nodes.Get(serverNode));
        serverApps.Start(Seconds(1.0));
        serverApps.Stop(Seconds(40.0));


        if (serverNode == 0)
        {
            for (int clientNode = 0; clientNode < 3; clientNode++)
            {
                UdpEchoClientHelper echoClient(interfaces[clientNode +
1].GetAddress(1), 9);
                echoClient.SetAttribute("MaxPackets", UintegerValue(3));
                echoClient.SetAttribute("Interval",
TimeValue(Seconds(1.0)));
                echoClient.SetAttribute("PacketSize",
UintegerValue(1024));


                ApplicationContainer clientApps =
echoClient.Install(nodes.Get(clientNode + 1));
                clientApps.Start(Seconds(2.0 + (clientNode) * 2));
                clientApps.Stop(Seconds(2.0 + (clientNode) * 2 + 1));
            }
        }
        else if (serverNode == 1)
        {
            UdpEchoClientHelper echoClient1(interfaces[0].GetAddress(1),
9);
            echoClient1.SetAttribute("MaxPackets", UintegerValue(3));
            echoClient1.SetAttribute("Interval", TimeValue(Seconds(1.0)));
            echoClient1.SetAttribute("PacketSize", UintegerValue(1024));
            ApplicationContainer clientApps1 =
echoClient1.Install(nodes.Get(0));
            clientApps1.Start(Seconds(8.0));
            clientApps1.Stop(Seconds(8.5));


            UdpEchoClientHelper echoClient2(interfaces[2].GetAddress(1),
9);
            echoClient2.SetAttribute("MaxPackets", UintegerValue(3));
            echoClient2.SetAttribute("Interval", TimeValue(Seconds(1.0)));
            echoClient2.SetAttribute("PacketSize", UintegerValue(1024));


            ApplicationContainer clientApps2 =
echoClient2.Install(nodes.Get(2));
            clientApps2.Start(Seconds(9.0));
            clientApps2.Stop(Seconds(9.5));


            UdpEchoClientHelper echoClient3(interfaces[3].GetAddress(1),
9);
            echoClient3.SetAttribute("MaxPackets", UintegerValue(3));
            echoClient3.SetAttribute("Interval", TimeValue(Seconds(1.0)));
            echoClient3.SetAttribute("PacketSize", UintegerValue(1024));


            ApplicationContainer clientApps3 =
echoClient3.Install(nodes.Get(3));
            clientApps3.Start(Seconds(10.0));
            clientApps3.Stop(Seconds(10.5));
        }
        else if (serverNode == 2)
        {
            UdpEchoClientHelper echoClient1(interfaces[1].GetAddress(1),
9);
            echoClient1.SetAttribute("MaxPackets", UintegerValue(3));
            echoClient1.SetAttribute("Interval", TimeValue(Seconds(1.0)));
            echoClient1.SetAttribute("PacketSize", UintegerValue(1024));
            ApplicationContainer clientApps1 =
echoClient1.Install(nodes.Get(1));
            clientApps1.Start(Seconds(11.0));
            clientApps1.Stop(Seconds(11.5));
            UdpEchoClientHelper echoClient2(interfaces[0].GetAddress(1),
9);
            echoClient2.SetAttribute("MaxPackets", UintegerValue(3));
            echoClient2.SetAttribute("Interval", TimeValue(Seconds(1.0)));
            echoClient2.SetAttribute("PacketSize", UintegerValue(1024));
            ApplicationContainer clientApps2 =
echoClient2.Install(nodes.Get(0));
            clientApps2.Start(Seconds(12.0));
            clientApps2.Stop(Seconds(12.5));


            UdpEchoClientHelper echoClient3(interfaces[3].GetAddress(1),
9);
            echoClient3.SetAttribute("MaxPackets", UintegerValue(3));
            echoClient3.SetAttribute("Interval", TimeValue(Seconds(1.0)));
            echoClient3.SetAttribute("PacketSize", UintegerValue(1024));


            ApplicationContainer clientApps3 =
echoClient3.Install(nodes.Get(3));
            clientApps3.Start(Seconds(13.0));
            clientApps3.Stop(Seconds(13.5));
        }
        else if (serverNode == 3)
        {
            UdpEchoClientHelper echoClient1(interfaces[2].GetAddress(1),
9);
            echoClient1.SetAttribute("MaxPackets", UintegerValue(3));
            echoClient1.SetAttribute("Interval", TimeValue(Seconds(1.0)));
            echoClient1.SetAttribute("PacketSize", UintegerValue(1024));
            ApplicationContainer clientApps1 =
echoClient1.Install(nodes.Get(2));
            clientApps1.Start(Seconds(14.0));
            clientApps1.Stop(Seconds(14.5));


            UdpEchoClientHelper echoClient2(interfaces[0].GetAddress(1),
9);
            echoClient2.SetAttribute("MaxPackets", UintegerValue(3));
            echoClient2.SetAttribute("Interval", TimeValue(Seconds(1.0)));
            echoClient2.SetAttribute("PacketSize", UintegerValue(1024));
            ApplicationContainer clientApps2 =
echoClient2.Install(nodes.Get(0));
            clientApps2.Start(Seconds(15.0));
            clientApps2.Stop(Seconds(15.5));
            UdpEchoClientHelper echoClient3(interfaces[1].GetAddress(1),
9);
            echoClient3.SetAttribute("MaxPackets", UintegerValue(3));
            echoClient3.SetAttribute("Interval", TimeValue(Seconds(1.0)));
            echoClient3.SetAttribute("PacketSize", UintegerValue(1024));


            ApplicationContainer clientApps3 =
echoClient3.Install(nodes.Get(1));
            clientApps3.Start(Seconds(16.0));
            clientApps3.Stop(Seconds(16.5));
        }
    }


    AnimationInterface anim("Topology2Ring.xml");
    Simulator::Run();
    Simulator::Destroy();
    return 0;
}
```

Output:

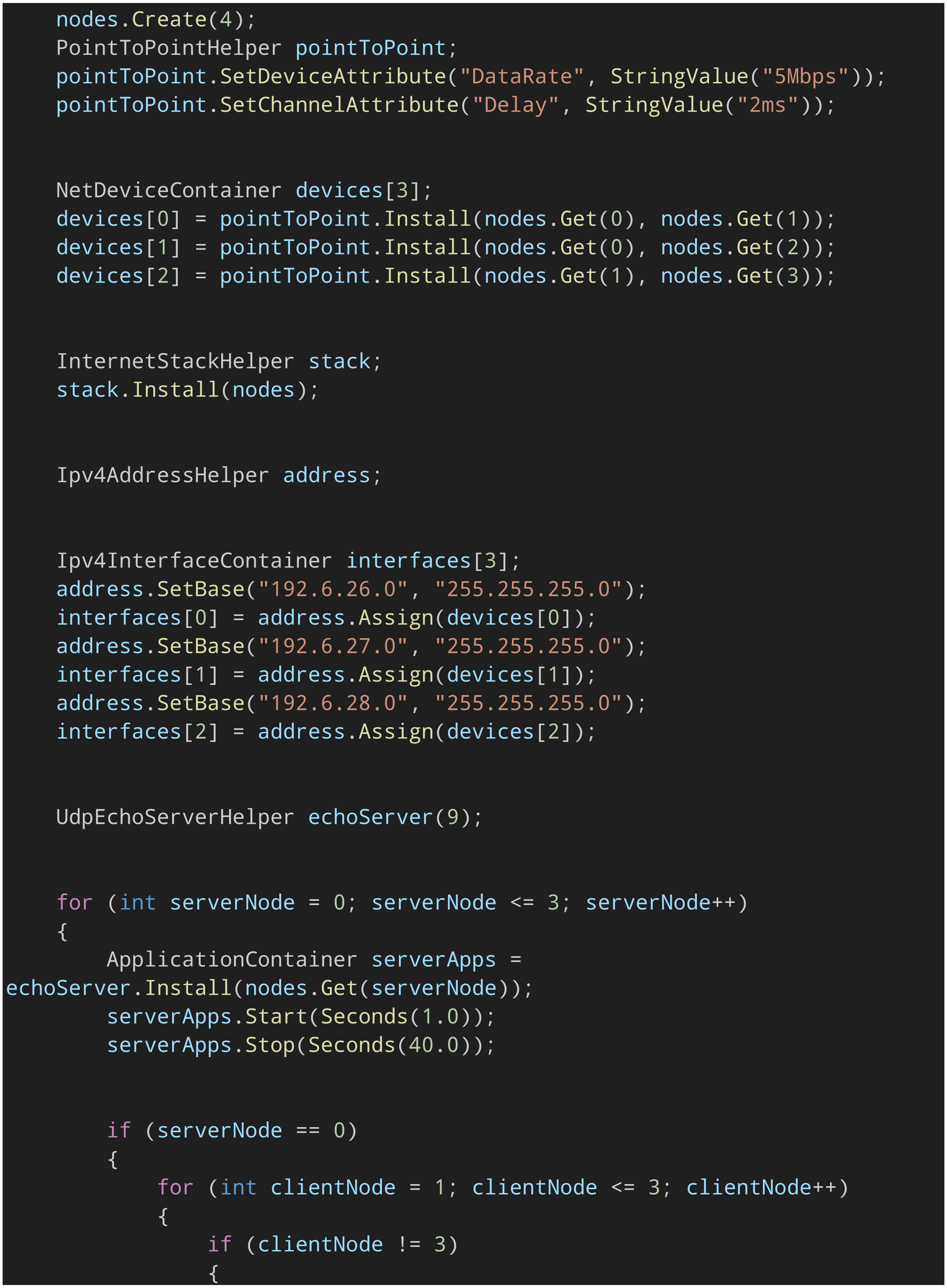
```
At time +2s client sent 1024 bytes to 192.6.36.2 port 9
At time +2.00369s server received 1024 bytes from 192.6.36.1 port 49153
At time +2.00369s server sent 1024 bytes to 192.6.36.1 port 49153
At time +2.00737s client received 1024 bytes from 192.6.36.2 port 9
At time +4s client sent 1024 bytes to 192.6.46.2 port 9
At time +4.00369s server received 1024 bytes from 192.6.46.1 port 49153
At time +4.00369s server sent 1024 bytes to 192.6.46.1 port 49153
At time +4.00737s client received 1024 bytes from 192.6.46.2 port 9
At time +6s client sent 1024 bytes to 192.6.56.2 port 9
At time +6.00369s server received 1024 bytes from 192.6.56.1 port 49153
At time +6.00369s server sent 1024 bytes to 192.6.56.1 port 49153
At time +6.00737s client received 1024 bytes from 192.6.56.2 port 9
At time +8s client sent 1024 bytes to 192.6.26.2 port 9
At time +8.00369s server received 1024 bytes from 192.6.26.1 port 49153
At time +8.00369s server sent 1024 bytes to 192.6.26.1 port 49153
At time +8.00737s client received 1024 bytes from 192.6.26.2 port 9
At time +9s client sent 1024 bytes to 192.6.46.2 port 9
At time +9.00369s server received 1024 bytes from 192.6.46.1 port 49154
At time +9.00369s server sent 1024 bytes to 192.6.46.1 port 49154
At time +9.00737s client received 1024 bytes from 192.6.46.2 port 9
At time +10s client sent 1024 bytes to 192.6.56.2 port 9
At time +10.0037s server received 1024 bytes from 192.6.56.1 port 49154
At time +10.0037s server sent 1024 bytes to 192.6.56.1 port 49154
At time +10.0074s client received 1024 bytes from 192.6.56.2 port 9
At time +11s client sent 1024 bytes to 192.6.36.2 port 9
At time +11.0037s server received 1024 bytes from 192.6.36.1 port 49154
At time +11.0037s server sent 1024 bytes to 192.6.36.1 port 49154
At time +11.0074s client received 1024 bytes from 192.6.36.2 port 9
At time +12s client sent 1024 bytes to 192.6.26.2 port 9
At time +12.0037s server received 1024 bytes from 192.6.26.1 port 49154
At time +12.0037s server sent 1024 bytes to 192.6.26.1 port 49154
At time +12.0074s client received 1024 bytes from 192.6.26.2 port 9
At time +13s client sent 1024 bytes to 192.6.56.2 port 9
At time +13.0037s server received 1024 bytes from 192.6.56.1 port 49155
At time +13.0037s server sent 1024 bytes to 192.6.56.1 port 49155
At time +13.0074s client received 1024 bytes from 192.6.56.2 port 9
At time +14s client sent 1024 bytes to 192.6.46.2 port 9
At time +14.0037s server received 1024 bytes from 192.6.46.1 port 49155
At time +14.0037s server sent 1024 bytes to 192.6.46.1 port 49155
At time +14.0074s client received 1024 bytes from 192.6.46.2 port 9
At time +15s client sent 1024 bytes to 192.6.26.2 port 9
At time +15.0037s server received 1024 bytes from 192.6.26.1 port 49155
At time +15.0037s server sent 1024 bytes to 192.6.26.1 port 49155
At time +15.0074s client received 1024 bytes from 192.6.26.2 port 9
At time +16s client sent 1024 bytes to 192.6.36.2 port 9
At time +16.0037s server received 1024 bytes from 192.6.36.1 port 49155
At time +16.0037s server sent 1024 bytes to 192.6.36.1 port 49155
At time +16.0074s client received 1024 bytes from 192.6.36.2 port 9
```

TASK #03:

1. Imagine a hierarchical tree network, like a binary tree, with a root node and multiple levels of child nodes. Connect them using point-to-point links.
2. Rest of the instructions are similar to task 1.

The code under main function is given below:

```
int
main(int argc, char* argv[])
{
    CommandLine cmd(__FILE__);
    cmd.Parse(argc, argv);


    Time::SetResolution(Time::NS);
    LogComponentEnable("UdpEchoClientApplication", LOG_LEVEL_INFO);
    LogComponentEnable("UdpEchoServerApplication", LOG_LEVEL_INFO);


    NodeContainer nodes;
```



```
    nodes.Create(4);
    PointToPointHelper pointToPoint;
    pointToPoint.SetDeviceAttribute("DataRate", StringValue("5Mbps"));
    pointToPoint.SetChannelAttribute("Delay", StringValue("2ms"));


    NetDeviceContainer devices[3];
    devices[0] = pointToPoint.Install(nodes.Get(0), nodes.Get(1));
    devices[1] = pointToPoint.Install(nodes.Get(0), nodes.Get(2));
    devices[2] = pointToPoint.Install(nodes.Get(1), nodes.Get(3));


    InternetStackHelper stack;
    stack.Install(nodes);


    Ipv4AddressHelper address;


    Ipv4InterfaceContainer interfaces[3];
    address.SetBase("192.6.26.0", "255.255.255.0");
    interfaces[0] = address.Assign(devices[0]);
    address.SetBase("192.6.27.0", "255.255.255.0");
    interfaces[1] = address.Assign(devices[1]);
    address.SetBase("192.6.28.0", "255.255.255.0");
    interfaces[2] = address.Assign(devices[2]);


    UdpEchoServerHelper echoServer(9);


    for (int serverNode = 0; serverNode <= 3; serverNode++)
    {
        ApplicationContainer serverApps =
echoServer.Install(nodes.Get(serverNode));
        serverApps.Start(Seconds(1.0));
        serverApps.Stop(Seconds(40.0));


        if (serverNode == 0)
        {
            for (int clientNode = 1; clientNode <= 3; clientNode++)
            {
                if (clientNode != 3)
                {
                    UdpEchoClientHelper echoClient(interfaces[clientNode - 
1].GetAddress(0), 9);
                    echoClient.SetAttribute("MaxPackets", 
UintegerValue(1));
                    echoClient.SetAttribute("Interval", 
TimeValue(Seconds(1.0)));
                    echoClient.SetAttribute("PacketSize", 
UintegerValue(1024));


                    ApplicationContainer clientApps = 
echoClient.Install(nodes.Get(clientNode));
                    clientApps.Start(Seconds(2.0 + (clientNode - 1) * 2));
                    clientApps.Stop(Seconds(6.0));
                }


                UdpEchoClientHelper 
echoClient1(interfaces[2].GetAddress(0), 9);
                echoClient1.SetAttribute("MaxPackets", UintegerValue(3));
                echoClient1.SetAttribute("Interval", 
TimeValue(Seconds(1.0)));
                echoClient1.SetAttribute("PacketSize", 
UintegerValue(1024));


                ApplicationContainer clientApps1 = 
echoClient1.Install(nodes.Get(3));
                clientApps1.Start(Seconds(7.0));
                clientApps1.Stop(Seconds(8.0));
            }
        }
        else if (serverNode == 1)
        {
            UdpEchoClientHelper echoClient1(interfaces[0].GetAddress(1), 
9);
            echoClient1.SetAttribute("MaxPackets", UintegerValue(3));
            echoClient1.SetAttribute("Interval", TimeValue(Seconds(1.0)));
            echoClient1.SetAttribute("PacketSize", UintegerValue(1024));


            ApplicationContainer clientApps1 = 
echoClient1.Install(nodes.Get(0));
            clientApps1.Start(Seconds(8.0));
            clientApps1.Stop(Seconds(8.5));
            UdpEchoClientHelper echoClient2(interfaces[1].GetAddress(0),
9);
            echoClient2.SetAttribute("MaxPackets", UintegerValue(3));
            echoClient2.SetAttribute("Interval", TimeValue(Seconds(1.0)));
            echoClient2.SetAttribute("PacketSize", UintegerValue(1024));


            ApplicationContainer clientApps2 =
echoClient2.Install(nodes.Get(2));
            clientApps2.Start(Seconds(9.0));
            clientApps2.Stop(Seconds(9.5));


            UdpEchoClientHelper echoClient3(interfaces[2].GetAddress(0),
9);
            echoClient3.SetAttribute("MaxPackets", UintegerValue(3));
            echoClient3.SetAttribute("Interval", TimeValue(Seconds(1.0)));
            echoClient3.SetAttribute("PacketSize", UintegerValue(1024));


            ApplicationContainer clientApps3 =
echoClient3.Install(nodes.Get(3));
            clientApps3.Start(Seconds(10.0));
            clientApps3.Stop(Seconds(10.5));
        }
        else if (serverNode == 2)
        {
            UdpEchoClientHelper echoClient1(interfaces[1].GetAddress(1),
9);
            echoClient1.SetAttribute("MaxPackets", UintegerValue(3));
            echoClient1.SetAttribute("Interval", TimeValue(Seconds(1.0)));
            echoClient1.SetAttribute("PacketSize", UintegerValue(1024));
            ApplicationContainer clientApps1 =
echoClient1.Install(nodes.Get(2));
            clientApps1.Start(Seconds(11.0));
            clientApps1.Stop(Seconds(11.5));


            UdpEchoClientHelper echoClient2(interfaces[0].GetAddress(0),
9);
            echoClient2.SetAttribute("MaxPackets", UintegerValue(3));
            echoClient2.SetAttribute("Interval", TimeValue(Seconds(1.0)));
            echoClient2.SetAttribute("PacketSize", UintegerValue(1024));
            ApplicationContainer clientApps2 =
echoClient2.Install(nodes.Get(0));
            clientApps2.Start(Seconds(12.0));
            clientApps2.Stop(Seconds(12.5));


            UdpEchoClientHelper echoClient3(interfaces[2].GetAddress(0),
9);
            echoClient3.SetAttribute("MaxPackets", UintegerValue(3));
            echoClient3.SetAttribute("Interval", TimeValue(Seconds(1.0)));
            echoClient3.SetAttribute("PacketSize", UintegerValue(1024));


            ApplicationContainer clientApps3 =
echoClient3.Install(nodes.Get(3));
            clientApps3.Start(Seconds(13.0));
            clientApps3.Stop(Seconds(13.5));
        }
        else if (serverNode == 3)
        {
            UdpEchoClientHelper echoClient1(interfaces[2].GetAddress(1),
9);
            echoClient1.SetAttribute("MaxPackets", UintegerValue(1));
            echoClient1.SetAttribute("Interval", TimeValue(Seconds(1.0)));
            echoClient1.SetAttribute("PacketSize", UintegerValue(1024));
            ApplicationContainer clientApps1 =
echoClient1.Install(nodes.Get(1));
            clientApps1.Start(Seconds(14.0));
            clientApps1.Stop(Seconds(14.5));


            UdpEchoClientHelper echoClient2(interfaces[0].GetAddress(1),
9);
            echoClient2.SetAttribute("MaxPackets", UintegerValue(3));
            echoClient2.SetAttribute("Interval", TimeValue(Seconds(1.0)));
            echoClient2.SetAttribute("PacketSize", UintegerValue(1024));
            ApplicationContainer clientApps2 =
echoClient2.Install(nodes.Get(0));
            clientApps2.Start(Seconds(15.0));
            clientApps2.Stop(Seconds(15.5));


            UdpEchoClientHelper echoClient3(interfaces[1].GetAddress(0),
9);
            echoClient3.SetAttribute("MaxPackets", UintegerValue(3));
            echoClient3.SetAttribute("Interval", TimeValue(Seconds(1.0)));
            echoClient3.SetAttribute("PacketSize", UintegerValue(1024));


            ApplicationContainer clientApps3 =
echoClient3.Install(nodes.Get(2));
            clientApps3.Start(Seconds(16.0));
            clientApps3.Stop(Seconds(16.5));
        }
    }


    AnimationInterface anim("Topology3Tree.xml");
    Simulator::Run();
    Simulator::Destroy();
    return 0;
}
```

Output:

```
At time +2s client sent 1024 bytes to 192.6.26.1 port 9
At time +2.00369s server received 1024 bytes from 192.6.26.2 port 49153
At time +2.00369s server sent 1024 bytes to 192.6.26.2 port 49153
At time +2.00737s client received 1024 bytes from 192.6.26.1 port 9
At time +4s client sent 1024 bytes to 192.6.27.1 port 9
At time +4.00369s server received 1024 bytes from 192.6.27.2 port 49153
At time +4.00369s server sent 1024 bytes to 192.6.27.2 port 49153
At time +4.00737s client received 1024 bytes from 192.6.27.1 port 9
At time +7s client sent 1024 bytes to 192.6.28.1 port 9
At time +7s client sent 1024 bytes to 192.6.28.1 port 9
At time +7s client sent 1024 bytes to 192.6.28.1 port 9
At time +7.00369s server received 1024 bytes from 192.6.28.2 port 49153
At time +7.00369s server sent 1024 bytes to 192.6.28.2 port 49153
At time +7.00537s server received 1024 bytes from 192.6.28.2 port 49154
At time +7.00537s server sent 1024 bytes to 192.6.28.2 port 49154
At time +7.00706s server received 1024 bytes from 192.6.28.2 port 49155
At time +7.00706s server sent 1024 bytes to 192.6.28.2 port 49155
At time +7.00737s client received 1024 bytes from 192.6.28.1 port 9
At time +7.00906s client received 1024 bytes from 192.6.28.1 port 9
At time +7.01075s client received 1024 bytes from 192.6.28.1 port 9
At time +8s client sent 1024 bytes to 192.6.26.2 port 9
At time +8.00369s server received 1024 bytes from 192.6.26.1 port 49153
At time +8.00369s server sent 1024 bytes to 192.6.26.1 port 49153
At time +8.00737s client received 1024 bytes from 192.6.26.2 port 9
At time +9s client sent 1024 bytes to 192.6.27.1 port 9
At time +9.00369s server received 1024 bytes from 192.6.27.2 port 49154
At time +9.00369s server sent 1024 bytes to 192.6.27.2 port 49154
At time +9.00737s client received 1024 bytes from 192.6.27.1 port 9
At time +10s client sent 1024 bytes to 192.6.28.1 port 9
At time +10.0037s server received 1024 bytes from 192.6.28.2 port 49156
At time +10.0037s server sent 1024 bytes to 192.6.28.2 port 49156
At time +10.0074s client received 1024 bytes from 192.6.28.1 port 9
At time +11s client sent 1024 bytes to 192.6.27.2 port 9
At time +11s server received 1024 bytes from 192.6.27.2 port 49155
At time +11s server sent 1024 bytes to 192.6.27.2 port 49155
At time +11s client received 1024 bytes from 192.6.27.2 port 9
At time +12s client sent 1024 bytes to 192.6.26.1 port 9
At time +12s server received 1024 bytes from 192.6.26.1 port 49154
At time +12s server sent 1024 bytes to 192.6.26.1 port 49154
At time +12s client received 1024 bytes from 192.6.26.1 port 9
At time +13s client sent 1024 bytes to 192.6.28.1 port 9
At time +13.0037s server received 1024 bytes from 192.6.28.2 port 49157
At time +13.0037s server sent 1024 bytes to 192.6.28.2 port 49157
At time +13.0074s client received 1024 bytes from 192.6.28.1 port 9
At time +14s client sent 1024 bytes to 192.6.28.2 port 9
At time +14.0037s server received 1024 bytes from 192.6.28.1 port 49154
At time +14.0037s server sent 1024 bytes to 192.6.28.1 port 49154
At time +14.0074s client received 1024 bytes from 192.6.28.2 port 9
At time +15s client sent 1024 bytes to 192.6.26.2 port 9
At time +15.0037s server received 1024 bytes from 192.6.26.1 port 49155
At time +15.0037s server sent 1024 bytes to 192.6.26.1 port 49155
At time +15.0074s client received 1024 bytes from 192.6.26.2 port 9
At time +16s client sent 1024 bytes to 192.6.27.1 port 9
At time +16.0037s server received 1024 bytes from 192.6.27.2 port 49156
At time +16.0037s server sent 1024 bytes to 192.6.27.2 port 49156
At time +16.0074s client received 1024 bytes from 192.6.27.1 port 9
```

Observation:

1. Each node has to be used as client and server at least once**.**
2. We have to be careful while creating connections for each topology.**.**
3. The sending and receiving times of packets has to be noted down.
4. IP addresses have to be assigned carefully