Assignment 2

Group number: 12

Students' Names:

Ahmed Ahmed Alwasefy

Ibrahim Mostafa Elshenhapy

Mohamed Salah Gabr

Date:

2023/05/27

Attention:

1. Only one of the group members should submit the solution.
2. You have to submit the solution using this template.
3. Your code and this report file must be attached in Brightspace.
4. In this file, first talk about modifications you did in code, then results (texts, figures, ...). Finally list the name of the files you attached in Brightspace

Task 2

1-1- Modifications in code

1. Please provide images of modifications in code. Highlight the parts you have changed.

```
#include "ns3/core-module.h"
#include "ns3/network-module.h"
#include "ns3/csma-module.h"
#include "ns3/internet-module.h"
#include "ns3/point-to-point-module.h"
#include "ns3/applications-module.h"
#include "ns3/mobility-module.h"
#include "ns3/ipv4-global-routing-helper.h"
#include "ns3/netanim-module.h" // Include the NetAnim module


// Default Network Topology
//
//                          10.1.1.0
// n8   n7   n6   n5   n0 --------------n1   n2   n3   n4
//  |    |    |    |    |  point-to-point  |    |    |    |
// ======================                ================
//    LAN2 10.1.3.0                        LAN1 10.1.2.0
```

```
bool verbose = true;
uint32_t nCsma1 = 3; //Lan1 4 nodes
uint32_t nCsma2 = 4; //define second lan2 network as it has 5 nodes

CommandLine cmd (__FILE__);
cmd.AddValue ("nCsma1", "Number of devices in the first CSMA network", nCsma1);
cmd.AddValue ("nCsma2", "Number of devices in the second CSMA network", nCsma2);
cmd.AddValue ("verbose", "Tell echo applications to log if true", verbose);

cmd.Parse (argc,argv);

if (verbose)
{
  LogComponentEnable ("UdpEchoClientApplication", LOG_LEVEL_INFO);
  LogComponentEnable ("UdpEchoServerApplication", LOG_LEVEL_INFO);
}

nCsma1 = nCsma1 == 0 ? 1 : nCsma1;
nCsma2 = nCsma2 == 0 ? 1 : nCsma2;
```

```
NodeContainer csmaNodes1;
csmaNodes1.Add (p2pNodes.Get (1)); // Last node of first CSMA network
csmaNodes1.Create (nCsma1);

NodeContainer csmaNodes2;
csmaNodes2.Add (p2pNodes.Get (0)); // First node of second CSMA network
csmaNodes2.Create (nCsma2);

//P2P topology with rate of 10Mbps and delay of 2ms.
PointToPointHelper pointToPoint;
pointToPoint.SetDeviceAttribute ("DataRate", StringValue ("10Mbps"));
pointToPoint.SetChannelAttribute ("Delay", StringValue ("2ms"));

NetDeviceContainer p2pDevices;
p2pDevices = pointToPoint.Install (p2pNodes);

//CSMA topology with data rate of 100Mbps and delay of 50ms. Node n1 from
CsmaHelper csma1;
csma1.SetChannelAttribute ("DataRate", StringValue ("100Mbps"));
csma1.SetChannelAttribute ("Delay", StringValue ("50ms"));

NetDeviceContainer csmaDevices1;
csmaDevices1 = csma1.Install (csmaNodes1);

//CSMA topology with data rate of 200Mbps and delay of 20ms. Node n0 from
CsmaHelper csma2;
csma2.SetChannelAttribute ("DataRate", StringValue ("200Mbps"));
csma2.SetChannelAttribute ("Delay", StringValue ("20ms"));

NetDeviceContainer csmaDevices2;
csmaDevices2 = csma2.Install (csmaNodes2);

InternetStackHelper stack;
stack.Install (csmaNodes1);
stack.Install (csmaNodes2);
```

```
address.SetBase ("10.1.2.0", "255.255.255.0");
Ipv4InterfaceContainer csmaInterfaces1;
csmaInterfaces1 = address.Assign (csmaDevices1);

address.SetBase ("10.1.3.0", "255.255.255.0");
Ipv4InterfaceContainer csmaInterfaces2;
csmaInterfaces2 = address.Assign (csmaDevices2);

//Install mobility on the nodes
MobilityHelper mobility;
mobility.SetPositionAllocator ("ns3::GridPositionAllocator",
                               "MinX", DoubleValue (0.0),
                               "MinY", DoubleValue (0.0),
                               "DeltaX", DoubleValue (5.0),
                               "DeltaY", DoubleValue (10.0),
                               "GridWidth", UintegerValue (3),
                               "LayoutType", StringValue ("RowFirst"));

//Constant position mobility model for this assignment.
mobility.SetMobilityModel ("ns3::ConstantPositionMobilityModel");

mobility.Install (csmaNodes1);
mobility.Install (csmaNodes2);
```

```
//The server is installed on the last node of LAN1 (n4).
ApplicationContainer serverApps = echoServer.Install (csmaNodes1.Get (nCsma1)); // Last node of first CSMA
serverApps.Start (Seconds (1.0));
serverApps.Stop (Seconds (11.0));

UdpEchoClientHelper echoClient (csmaInterfaces1.GetAddress (nCsma1), 9); // Last address of second CSMA net
echoClient.SetAttribute ("MaxPackets", UintegerValue (20));
echoClient.SetAttribute ("Interval", TimeValue (Seconds (1.0)));
echoClient.SetAttribute ("PacketSize", UintegerValue (1024));

//The client is installed on the last node of LAN2 (n8).
ApplicationContainer clientApps = echoClient.Install (csmaNodes2.Get (nCsma2)); // Last node of second CSMA
clientApps.Start (Seconds (2.0));
clientApps.Stop (Seconds (11.0));
```

```
pointToPoint.EnablePcapAll ("/home/shenhapy/Downloads/ns-allinone-3.35/ns-3.35/pcap/second");
csma1.EnablePcap ("/home/shenhapy/Downloads/ns-allinone-3.35/ns-3.35/pcap/second", csmaDevices1.Get (1), true);
csma2.EnablePcap ("/home/shenhapy/Downloads/ns-allinone-3.35/ns-3.35/pcap/second", csmaDevices2.Get (0), true);

//Add animation module in order to visualize the simulation.
// Create a trace file for NetAnim.
AnimationInterface anim("/home/shenhapy/Downloads/ns-allinone-3.35/animation.xml");

// Set positions for csmaNodes1 (n1, n2, n3, n4).
anim.SetConstantPosition(csmaNodes1.Get(0), 10, 10);  // n1
anim.SetConstantPosition(csmaNodes1.Get(1), 0, 5);  // n2
anim.SetConstantPosition(csmaNodes1.Get(2), 20, 5);  // n3
anim.SetConstantPosition(csmaNodes1.Get(3), 10, 0);  // n4

// Set positions for csmaNodes2 (n0, n5, n6, n7, n8).
anim.SetConstantPosition(csmaNodes2.Get(0), 10, 17.5);  // n0
anim.SetConstantPosition(csmaNodes2.Get(1), 0, 20);  // n5
anim.SetConstantPosition(csmaNodes2.Get(2), 20, 15);  // n6
anim.SetConstantPosition(csmaNodes2.Get(3), 20, 20); // n7
anim.SetConstantPosition(csmaNodes2.Get(4), 10, 25); // n8
```

2. A technical reflection on the assignment: Write a brief account (around 250 words) about your approach and the challenges you faced during this assignment. Reflect on the following:

   • Difficulties encountered and solutions implemented while setting up the UDP Echo Client and Server application.

   While setting up the UDP Echo Client and Server application in the network simulation, one of the difficulties arose from the complex network topology, involving three networks: csma1, csma2, and p2p. The task of defining the points for the p2p nodes, considering their connection with different csma networks. However, by carefully analyzing the network structure and adjusting, we were able to overcome this challenge and successfully define the points.

   • Your insights about the interplay between P2P and CSMA network topologies, gained through this assignment.

   Regarding the experience and learning gained from adding the animation module to the script, it was an insightful process. We started by reading the NetAnim documentation to understand its capabilities and then integrated it into the script. The simulation assisted in identifying and rectifying configuration errors in the network setup, such as inaccurately specifying the number of nodes for one of the csma networks and the server for sending packets. NetAnim proved to be a powerful tool for visualizing the network and aided in rearranging the nodes to achieve better visualization.

   • Experience and learning about network simulation from the process of adding the animation module to your script.

   Through this assignment, I gained insights into the interplay between P2P and CSMA network topologies. The P2P topology exhibited minimal delay and a lower data rate since it established a dedicated, high-speed connection between two nodes. On the other hand, the CSMA networks represented shared Ethernet-like environments, resulting in higher delay but a higher data rate as the communication was extended to multiple points. This experience highlighted the trade-offs between dedicated point-to-point links and shared networks, providing a deeper understanding of their characteristics and implications in network simulations.

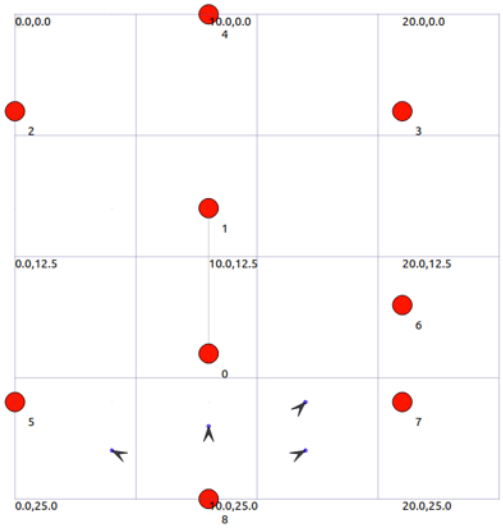
1-2- Results

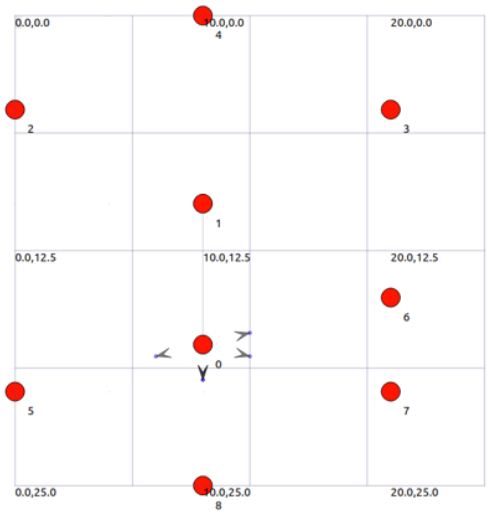
Please provide your output (figures, texts, …):

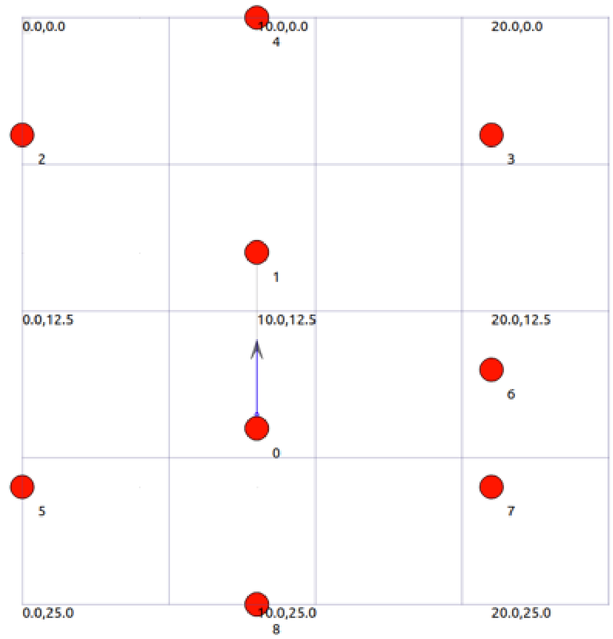
1. A snapshot of the command line output showing the log of sent and received packets

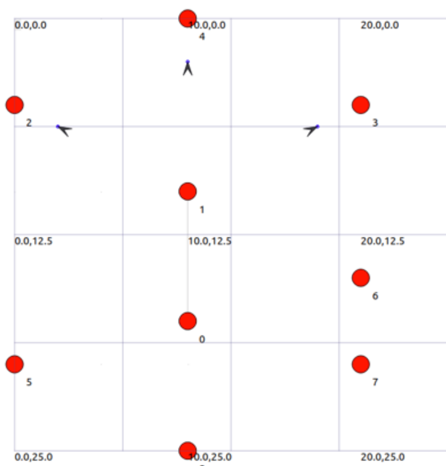
```
shenhapy@shenhapy: ~/Downloads/ns-allinone-3.35/ns-3.35
[2100/2171] Compiling scratch/mysecond.cc
[2131/2171] Linking build/scratch/mysecond
Waf: Leaving directory `/home/shenhapy/Downloads/ns-allinone-3.35/ns-3.35/build'
Build commands will be stored in build/compile_commands.json
'build' finished successfully (4.723s)
At time +2s client sent 1024 bytes to 10.1.2.4 port 9
At time +2.22099s server received 1024 bytes from 10.1.3.5 port 49153
At time +2.22099s server sent 1024 bytes to 10.1.3.5 port 49153
At time +2.44098s client received 1024 bytes from 10.1.2.4 port 9
At time +3s client sent 1024 bytes to 10.1.2.4 port 9
At time +3.07297s server received 1024 bytes from 10.1.3.5 port 49153
At time +3.07297s server sent 1024 bytes to 10.1.3.5 port 49153
At time +3.14594s client received 1024 bytes from 10.1.2.4 port 9
At time +4s client sent 1024 bytes to 10.1.2.4 port 9
At time +4.07297s server received 1024 bytes from 10.1.3.5 port 49153
At time +4.07297s server sent 1024 bytes to 10.1.3.5 port 49153
At time +4.14594s client received 1024 bytes from 10.1.2.4 port 9
At time +5s client sent 1024 bytes to 10.1.2.4 port 9
At time +5.07297s server received 1024 bytes from 10.1.3.5 port 49153
At time +5.07297s server sent 1024 bytes to 10.1.3.5 port 49153
At time +5.14594s client received 1024 bytes from 10.1.2.4 port 9
At time +6s client sent 1024 bytes to 10.1.2.4 port 9
At time +6.07297s server received 1024 bytes from 10.1.3.5 port 49153
At time +6.07297s server sent 1024 bytes to 10.1.3.5 port 49153
At time +6.14594s client received 1024 bytes from 10.1.2.4 port 9
At time +7s client sent 1024 bytes to 10.1.2.4 port 9
At time +7.07297s server received 1024 bytes from 10.1.3.5 port 49153
At time +7.07297s server sent 1024 bytes to 10.1.3.5 port 49153
At time +7.14594s client received 1024 bytes from 10.1.2.4 port 9
At time +8s client sent 1024 bytes to 10.1.2.4 port 9
At time +8.07297s server received 1024 bytes from 10.1.3.5 port 49153
At time +8.07297s server sent 1024 bytes to 10.1.3.5 port 49153
At time +8.14594s client received 1024 bytes from 10.1.2.4 port 9
At time +9s client sent 1024 bytes to 10.1.2.4 port 9
At time +9.07297s server received 1024 bytes from 10.1.3.5 port 49153
At time +9.07297s server sent 1024 bytes to 10.1.3.5 port 49153
At time +9.14594s client received 1024 bytes from 10.1.2.4 port 9
At time +10s client sent 1024 bytes to 10.1.2.4 port 9
At time +10.073s server received 1024 bytes from 10.1.3.5 port 49153
At time +10.073s server sent 1024 bytes to 10.1.3.5 port 49153
At time +10.1459s client received 1024 bytes from 10.1.2.4 port 9
shenhapy@shenhapy:~/Downloads/ns-allinone-3.35/ns-3.35$
```

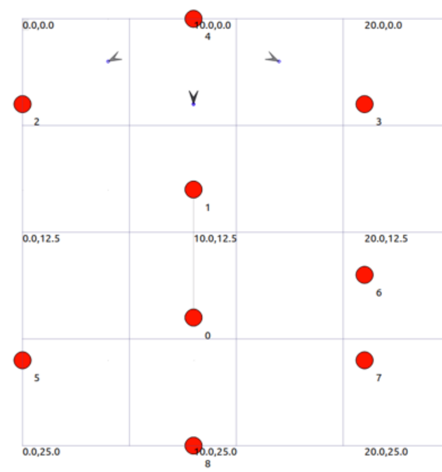
2. A snapshot of the visualization using NetAnim

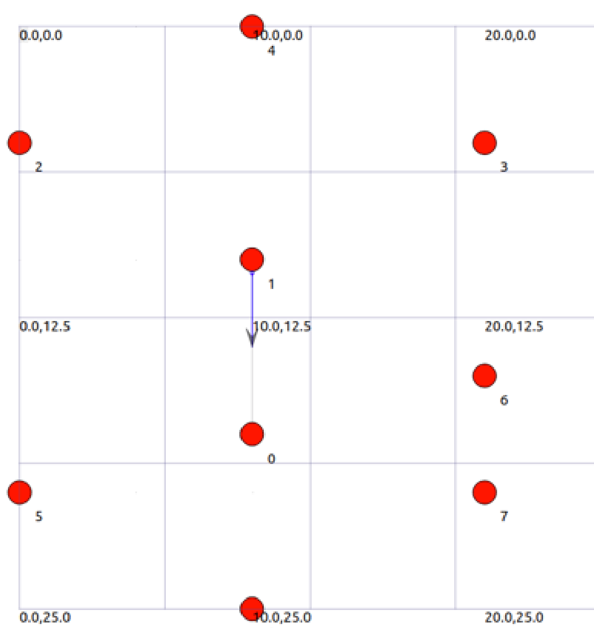


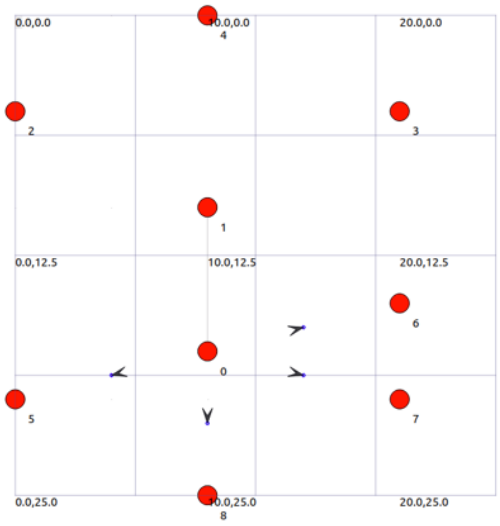


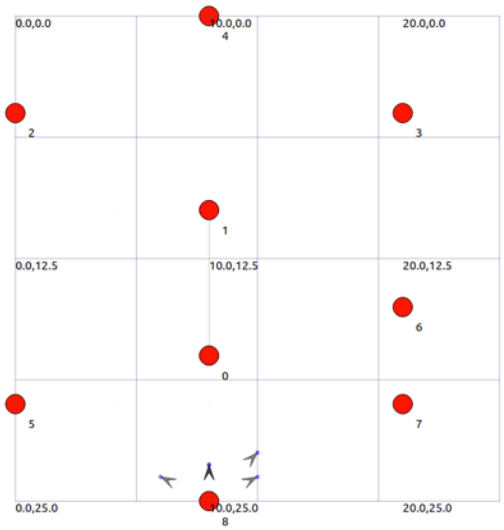

0.0,0.0
10.0,0.0
20.0,0.0
4
2
3
1
0.0,12.5
10.0,12.5
20.0,12.5
6
0
5
7
0.0,25.0
10.0,25.0
20.0,25.0
8


0.0,0.0
10.0,0.0
20.0,0.0
4
2
3
1
0.0,12.5
10.0,12.5
20.0,12.5
6
0
5
7
0.0,25.0
10.0,25.0
20.0,25.0
8


0.0,0.0
10.0,0.0
20.0,0.0
4
2
3
1
0.0,12.5
10.0,12.5
20.0,12.5
6
0
5
7
0.0,25.0
10.0,25.0
20.0,25.0
8


0.0,0.0
10.0,0.0
20.0,0.0
4
2
3
1
0.0,12.5
10.0,12.5
20.0,12.5
6
0
5
7
0.0,25.0
10.0,25.0
20.0,25.0
8





1-3- List of files

Please mention the name of the files you have attached in Brightspace. Also, attach this file and modified code file in Brightspace

Group12_Assignment2.rar

- 1-Code_GP12.cpp
- 2-Report_GP12.pdf
- 5-animation.xml
- 5- second-0-0.pcap
- 5- second-0-1.pcap
- 5- second-1-0.pcap
- 5- second-2-0.pcap
- 5- second-3-0.pcap
- 5- second-3-1.pcap
- 5- second-4-0.pcap
- 5- second-4-1.pcap
- 5- second-5-0.pcap
- 5- second-6-0.pcap
- 5- second-7-0.pcap
- 5- second-8-0.pcap