LAB 1

Lab - 2

Aim:- Implementation of hub and switch

Procedure:-

i) Open Cisco packet tracer.

ii) Connect some end devices with hubs and connecter those hubs with swi-tch.

iii) Configure all those end devices and assign IP address to end devices.

iv) Pass simple pdu messages accross the devices and note down result using simulation mode.

Topology:- Star topology.

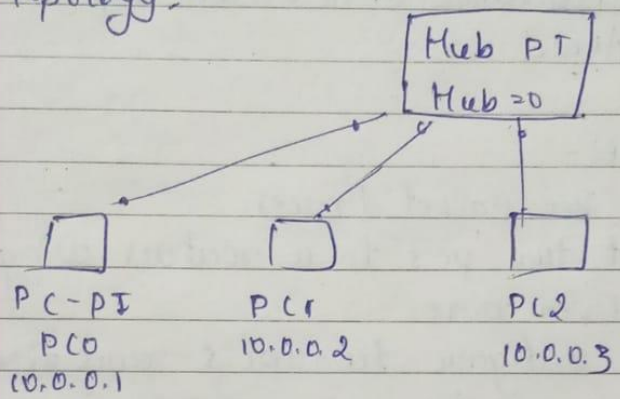
Result:- Successful transfer of simple pdu from source to destination end devices.

Observations:-

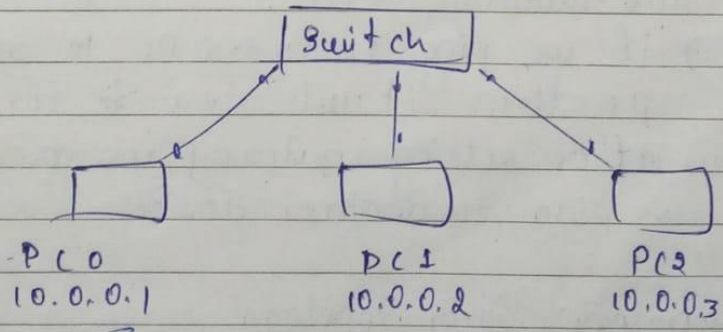
Hubs are the devices with doesn't have storage and processing feature hence they just boardcast / transmit to all the end devices of the same network. while switches perform as hub during initial stage but later transfer from

source to destination directly. due to storage and processing feature. using ARP table.

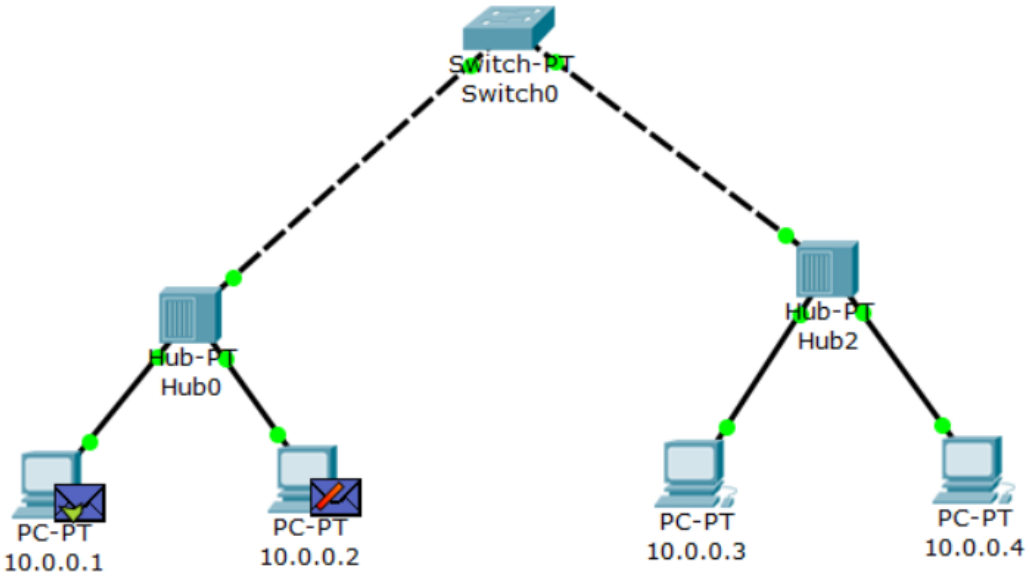
Topology :-

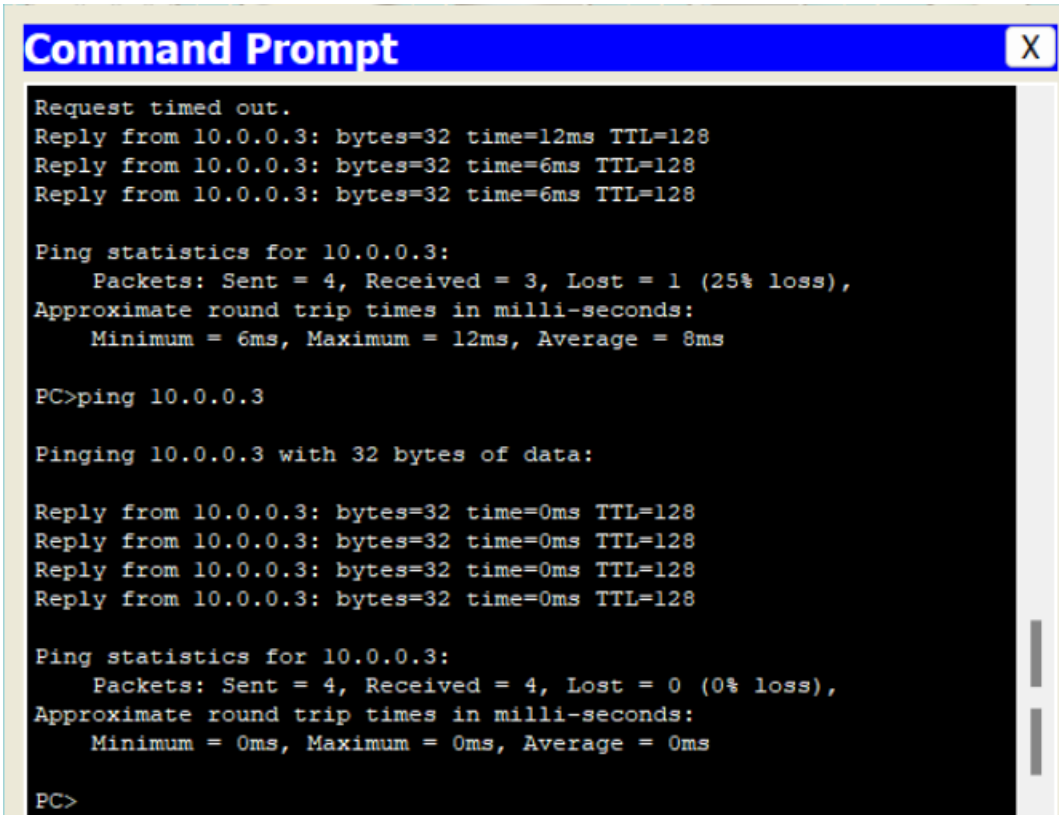


Using Switch

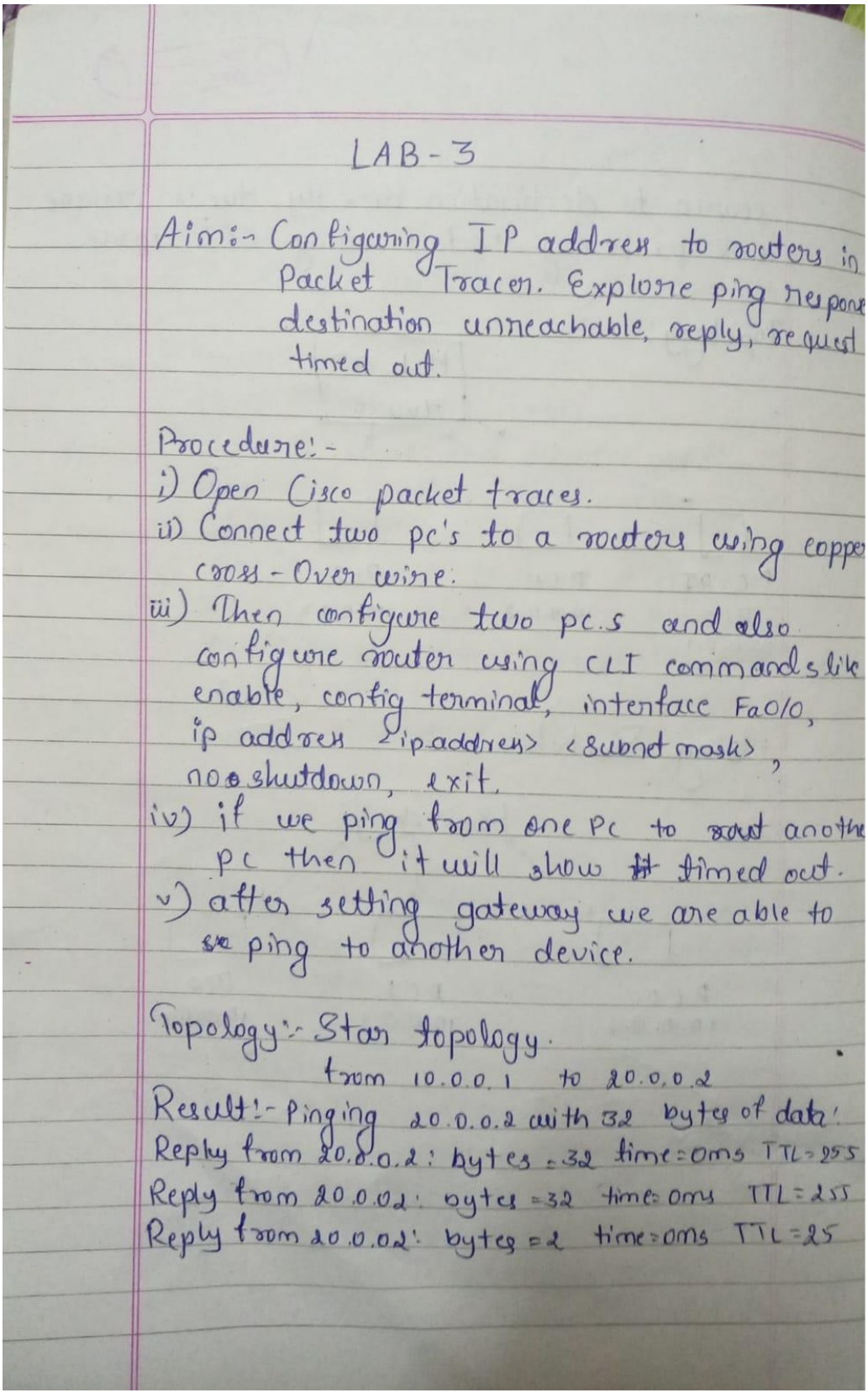


Meeting
13/11/2021


Switch-PT
Switch0
Hub-PT
Hub0
Hub-PT
Hub2
PC-PT
10.0.0.1
PC-PT
10.0.0.2
PC-PT
10.0.0.3
PC-PT
10.0.0.4


Command Prompt
X
Request timed out.
Reply from 10.0.0.3: bytes=32 time=12ms TTL=128
Reply from 10.0.0.3: bytes=32 time=6ms TTL=128
Reply from 10.0.0.3: bytes=32 time=6ms TTL=128
Ping statistics for 10.0.0.3:
Packets: Sent = 4, Received = 3, Lost = 1 (25% loss),
Approximate round trip times in milli-seconds:
Minimum = 6ms, Maximum = 12ms, Average = 8ms
PC>ping 10.0.0.3
Pinging 10.0.0.3 with 32 bytes of data:
Reply from 10.0.0.3: bytes=32 time=0ms TTL=128
Reply from 10.0.0.3: bytes=32 time=0ms TTL=128
Reply from 10.0.0.3: bytes=32 time=0ms TTL=128
Reply from 10.0.0.3: bytes=32 time=0ms TTL=128
Ping statistics for 10.0.0.3:
Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
Approximate round trip times in milli-seconds:
Minimum = 0ms, Maximum = 0ms, Average = 0ms
PC>

LAB 2



LAB-3

Aim:- Configuring IP address to routers in Packet Tracer. Explore ping response destination unreachable, reply, request timed out.

Procedure:-

i) Open Cisco packet traces.

ii) Connect two pc's to a routers using copper cross-Over wire.

iii) Then configure two pc.s and also configure router using CLI commands like enable, config terminal, interface Fa0/0, ip address <ipaddress> <subnet mask>, no shutdown, exit.

iv) if we ping from one Pc to another pc then it will show timed out.

v) after setting gateway we are able to ping to another device.

Topology:- Star topology.

from 10.0.0.1 to 20.0.0.2

Result:- Pinging 20.0.0.2 with 32 bytes of data:

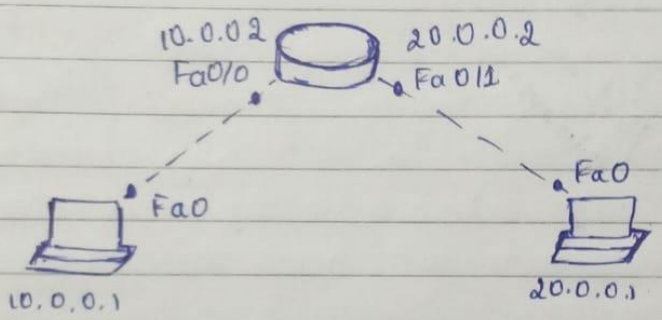
Reply from 20.0.0.2: bytes=32 time=0ms TTL=255

Reply from 20.0.0.2: bytes=32 time=0ms TTL=255

Reply from 20.0.0.2: bytes=2 time=0ms TTL=25

ping statistics for 20.0.0.2:
Packets: sent = 43, Received = 43, Lost = 0 (0% loss),
Approximate round trip times in milliseconds:
Minimum = 0ms, Maximum = 5ms, Avg = 1ms

Topology:-



Gateway: 10.0.0.2 Gateway: 20.0.0.2

Observation:-

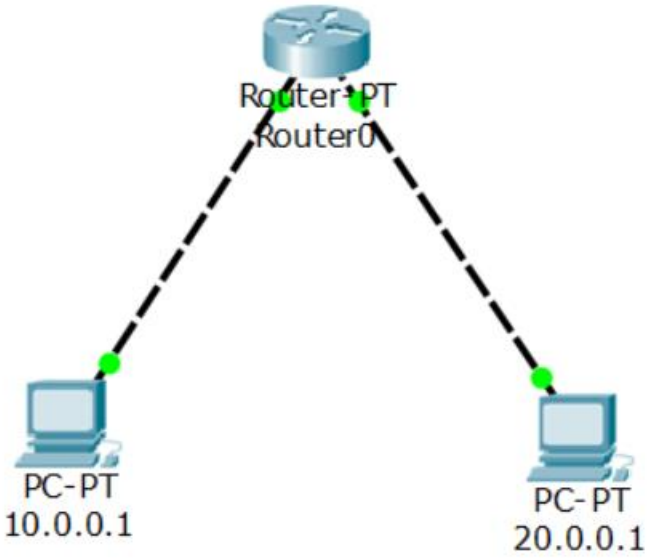
i) Before configuring PC with router ip-address if we ping from 10.0.0.1 to 20.0.0.2 on 20.0.0.1

pinging 20.0.0.2 with 32 bytes of data:
Request timed out:
Request timed out.
Request timed out.
ping statistics for 20.0.0.2:
Packets: Sent=3, Received=0, Lost = 3 (100% Loss),

and if we ping to 10.0.0.2, it will successfully gonna get reply.

ii) And if we configure gateway.
for PC1 : 10.0.0.2
PC2 : 20.0.0.2
And if we ping to all other devices, we will successfully getting reply from all devices.

Machine Problems



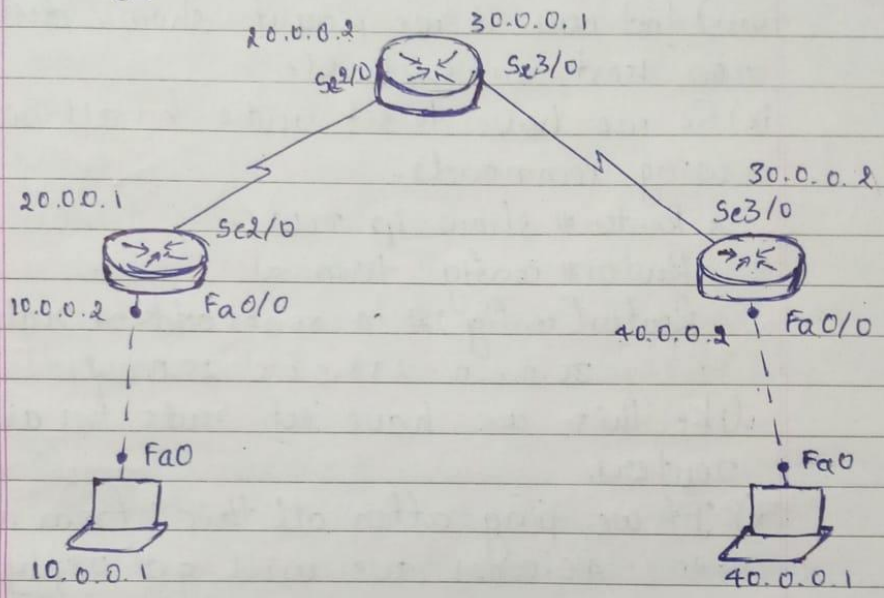

```
PC>ping 20.0.0.1

Pinging 20.0.0.1 with 32 bytes of data:

Reply from 20.0.0.1: bytes=32 time=6ms TTL=127
Reply from 20.0.0.1: bytes=32 time=0ms TTL=127
Reply from 20.0.0.1: bytes=32 time=1ms TTL=127
Reply from 20.0.0.1: bytes=32 time=0ms TTL=127

Ping statistics for 20.0.0.1:
    Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
Approximate round trip times in milli-seconds:
    Minimum = 0ms, Maximum = 6ms, Average = 1ms
```

LAB 3

Lab - 4 static

Aim:- Configuring default route to the Router

Topology:- static



Procedure.

i) Two routers are connected to end devices and another router is used connected those two routers

ii) And all used routers are generic routers.

iii) these routers are connected using serial DCE connection.

iv) Configure Ip address for end devices (PC-PT) and interfaces of routers.

v) PC1 -> IP- 10.0.0.1 PC0:- 40.0.0.1

router 1 = 10.0.0.2 & 20.0.0.1

router 2 = 20.0.0.2 & 30.0.0.1

router 3 = 30.0.0.2 & 40.0.0.2

vi) Now if we ping result will be request timed out.

vii) Therefore we will set gateway of PC1 as 20.0.0.1 & PC2 as 40.0.0.2

viii) And now if we ping it shows destination host unreachable.

ix) So we have to set route for all routers using commands.

- Router# show ip route.
  Router# config terminal
  Router(config)# ip ~~route address~~ route
  30.0.0.0 255.0.0.0 20.0.0.2

like this we have set route for all routers.

x) If we ping after all this from 10.0.0.1 to 40.0.0.1, we will get reply.

Observation:-

i) A ping cross the interface until a gateway has been set to the connected interface/router.

ii) Once gateway has been set, the ping will not cross over to another router as the routers are not connected to other networks and they won't know which route to take are where the next hop of the ping is to be done.

iii) The routers are configured with ip route centre the network is gives to ~~to~~ all the network not directly connected to it.

Result:- A Successful ping message is terminated from one end device to another end devices.

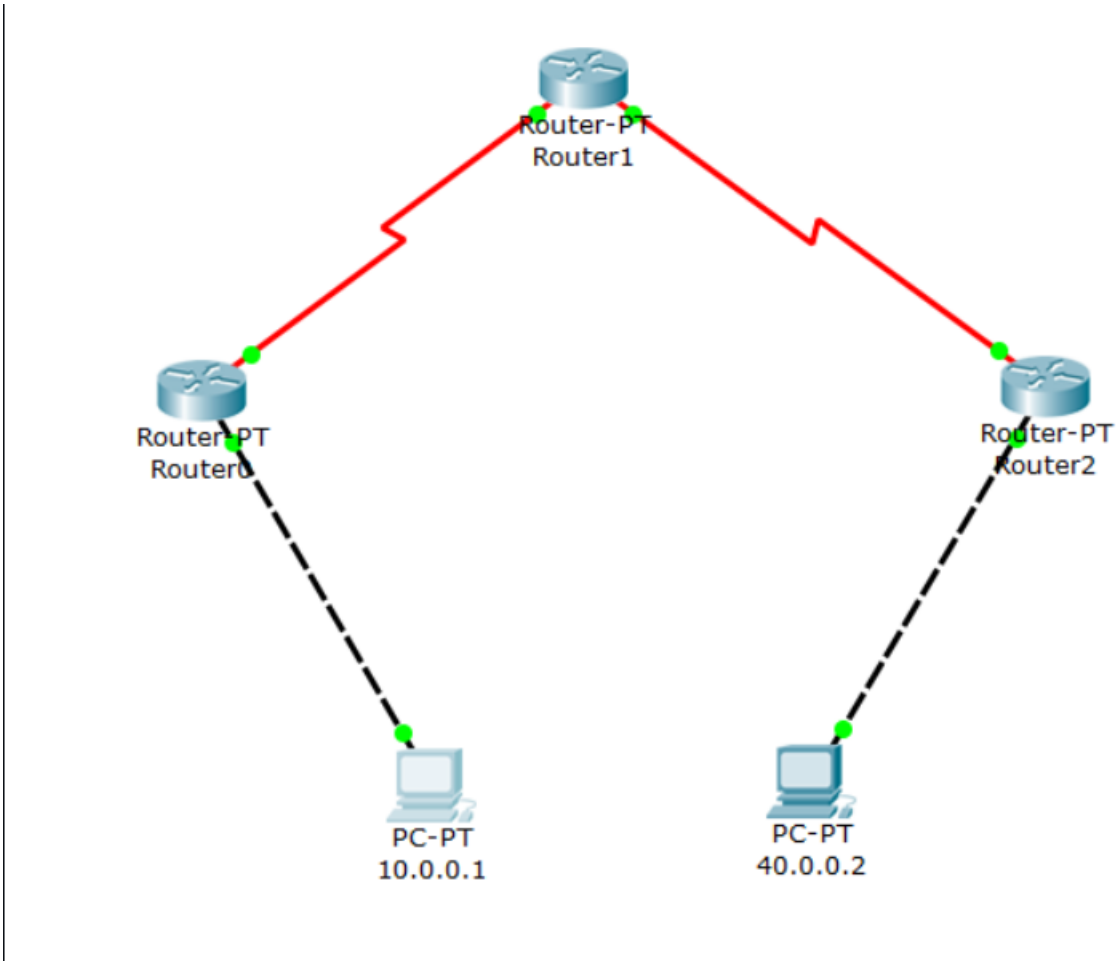
Output:-

1) PC> ping 20.0.0.1 (from 10.0.0.1)
ping ing 20.0.0.1 with 32 bytes of data.
request timed out
request timed out
request timd out.
request timed out.
ping statistics for 20.0.0.1:
Packets: sent=4 received=0, lost=4(100% loss)

2) PC> ping 30.0.0.1
Pinging 30.0.0.1 with 32 bytes of data.
Destination host not reachable: ~~over~~ error.

3) PC> ping 40.0.0.1
pinging 40.0.0.1 with 32 bytes of data.
Reply from 40.0.0.1: bytes=32 time=8ms TTL=125
Reply from 40.0.0.1: bytes=32 time=8ms TTL=125
Reply from 40.0.0.1: bytes=32 time=8ms TTL=125
Reply from 40.0.0.1: bytes=32 time=8ms TTL=125
ping statistics for 20.0.0.1
Packet: sent=4 received=4 lost=0 (0% loss)

Neelima
1/12/2022




```
PC>ping 40.0.0.2

Pinging 40.0.0.2 with 32 bytes of data:

Reply from 40.0.0.2: bytes=32 time=2ms TTL=125
Reply from 40.0.0.2: bytes=32 time=2ms TTL=125
Reply from 40.0.0.2: bytes=32 time=2ms TTL=125
Reply from 40.0.0.2: bytes=32 time=4ms TTL=125

Ping statistics for 40.0.0.2:
    Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
Approximate round trip times in milli-seconds:
    Minimum = 2ms, Maximum = 4ms, Average = 2ms
```

LAB 4

Result:- ping 30.0.0.1 (from 10.0.0.1)
Before configuring gateway.
request timed out
request timed out
request timed out
request timed out
ping Statistics
Packets: sent=4 recieved=0 lost=4 (100% loss)

Before configuring default route.
ping 30.0.0.1
Reply from 30.0.0.1 Destination host unreachable
Reply from 30.0.0.1 Destination host unreachable
Reply from 30.0.0.1 Destination host unreachable.
Ping Statistics
Packets: sent=4 recieved=0 lost=4 (100% loss)

After configuring
ping 30.0.0.1
Reply from 30.0.0.1 bytes=32 time=8ms TTL=125
Reply from 30.0.0.1 bytes=32 time=8ms TTL=125
Reply from 30.0.0.1 bytes=32 time=8ms TTL=125
Ping Statistics
Packets: sent=4 recieved=4 lost=0 (0% loss)

enable
config terminal
interface Se 2/0
ip route 10.0.0.2 255.0.0.0
no shutdown
exit

→ configure gateway address of the end devices with connected interface of the router.

→ Inorder to establish default route across the routers, ip route is configured using the same command

ip route 00.0.0 0.0.0.0 20.0.0.2

Observation

i) A ping doesn't cross the interface until a gateway has been sent to the connected interface/ router.

ii) Once gateway has been set, the ping will not cross over to another router as the routers are not connected to other networks and they won't know which route to take are where the next hop of the ping is to be done.

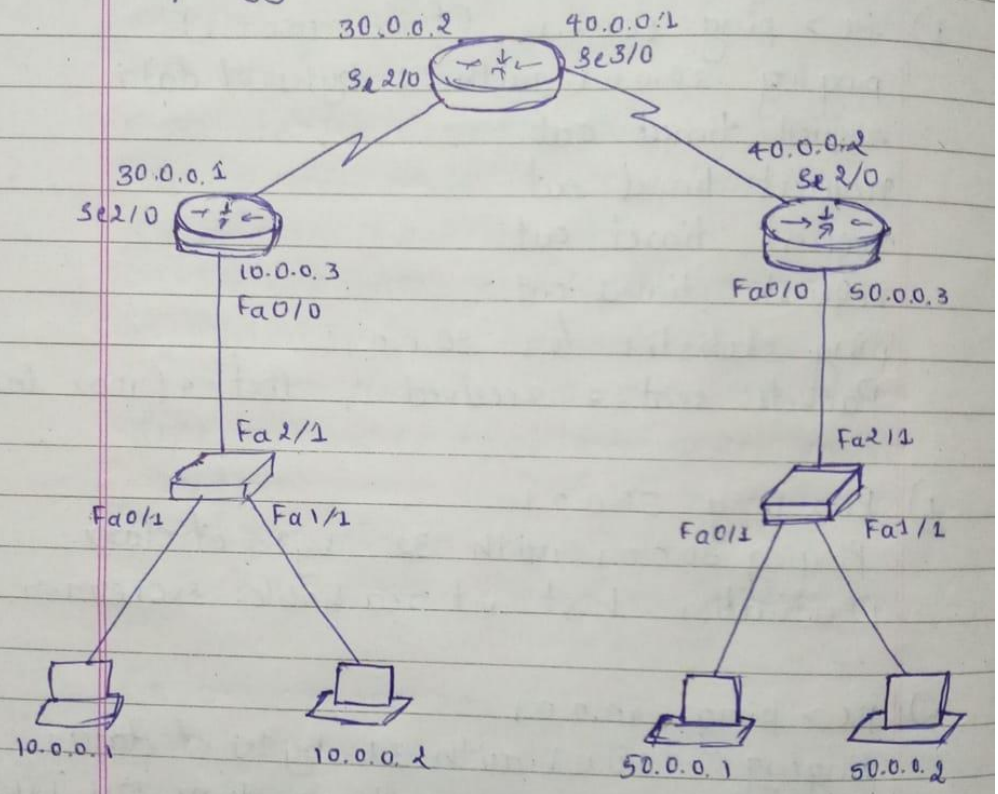
iii) Default route is configured between the router, where ip route and subnet mask is not specified. only via interface of connected router.

iv Later pinged connections between all the routers and end devices.

LAB - 5

Aim:- Configuring default route to router.

Topology:-



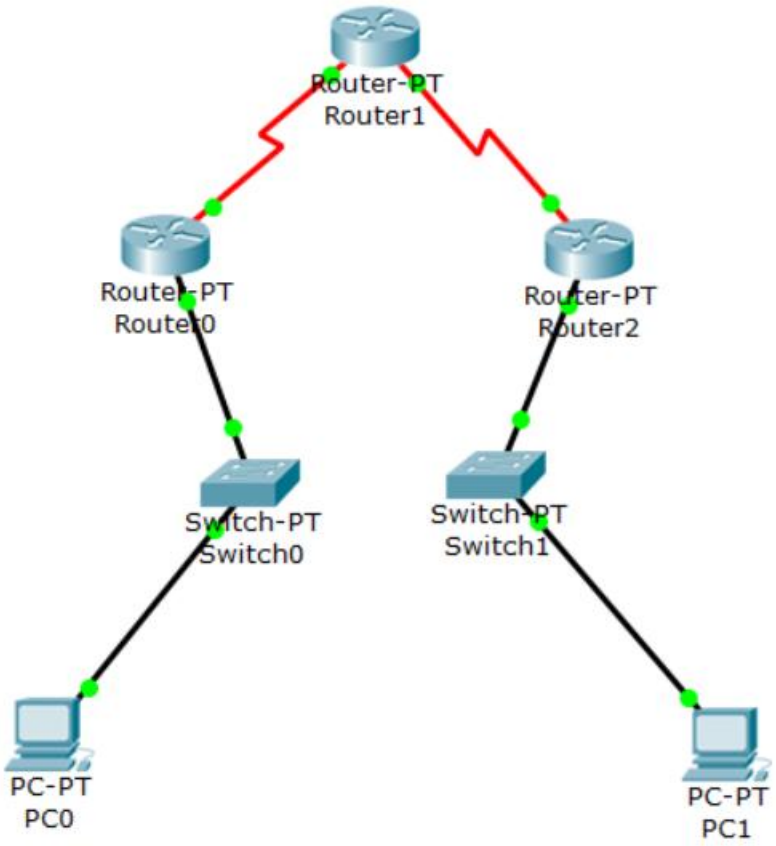
Procedure:-

i) Two routers are connected to end devices and through switch another router

ii) Configure end devices and interface router

iii) Configure IP address between routers as serial interface.

iv) To configure IP address following commands are executed.




```
PC>ping 40.0.0.2

Pinging 40.0.0.2 with 32 bytes of data:

Reply from 40.0.0.2: bytes=32 time=14ms TTL=125
Reply from 40.0.0.2: bytes=32 time=11ms TTL=125
Reply from 40.0.0.2: bytes=32 time=11ms TTL=125
Reply from 40.0.0.2: bytes=32 time=11ms TTL=125

Ping statistics for 40.0.0.2:
    Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
Approximate round trip times in milli-seconds:
    Minimum = 11ms, Maximum = 14ms, Average = 11ms
```

LAB 5

RIP: Routing Information protocol.
- is a dynamic routing protocol that uses hop count as a metric to find the best path between source and destination network.

Neelima
8/12/2022

ping statistics for 40.0.0.1:
Packets: sent=4, recieved=4; lost=0 (0% loss)
Approximate round ip in milliseconds
minimum= 2ms, maximum= 10ms,
Average= 10ms

Since RIP protocol has been established, IP route does not have to be set for each router.

Before RIP has set:
ping 10.0.0.1 → 40.0.0.1: Destination host unreachable

Before RIP
ping 10.x → 20.x
↳ Request Timed Out.

Only on correctly configuring gateways and protocol, is the reply recieved properly.

Result:
(Routing Information protocol) RIP is established in the network correctly.

Note: Even on proper connection and configuration the first packet of the first internetwork ping is timed out as the switches have not learnt the network yet.

enable
config terminal
interface Se 2/0
ip route 10.0.0.2 255.0.0.0
no shutdown
exit

* configure gateway address of the end devices with connected interface of the router.
* Inorder to establish default route across the routers ip route is configured using command

ip route 0.0.0.0 0.0.0.0 20.0.0.2

to configure rip protocol we use command

router rip
network 20.0.0.0

Observation:- For every serial dce connection to configure rip with defined clock rate use following command.

encapsulation ppp
clock rate 64000
no shutdown

Observation:-

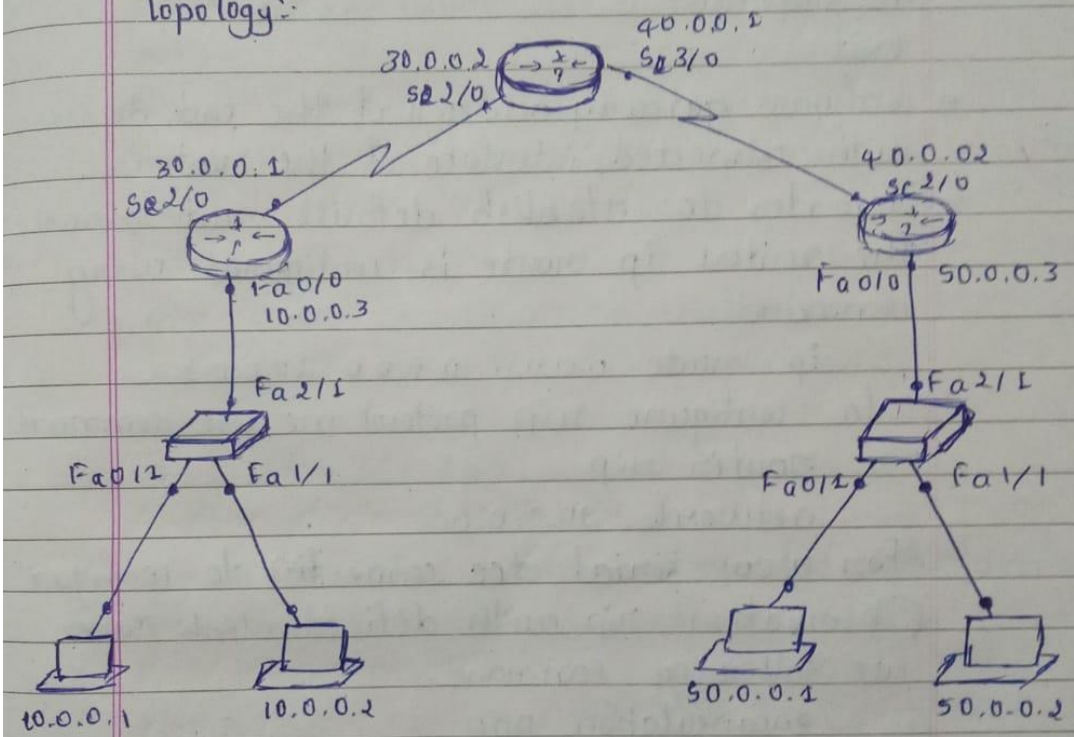
ping 40.0.0.1

Reply from 40.0.0.1 ~~bytes~~ with 32 bytes of data.
Reply from 40.0.0.1 bytes=32 time=2ms TTL=125
Reply from 40.0.0.1 bytes=32 time=2ms TTL=125
Reply from 40.0.0.1 bytes=32 time=2ms TTL=125
Reply from 40.0.0.1 bytes=32 time=2ms TTL=125

LAB-6

Aim:- Configuring RIP routing protocol in routers.

Topology:-



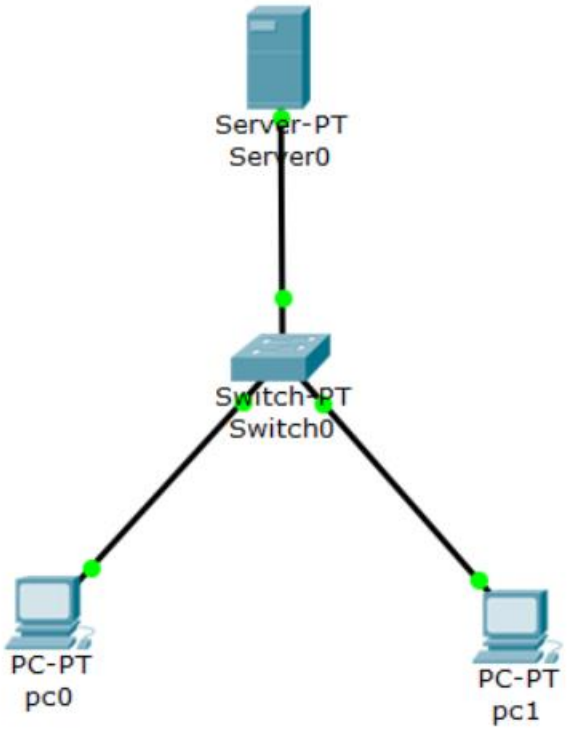
Procedure

i) Two routers are connected to end devices and another routers.

ii) Configure end devices and interface router.

iii) Configure IP address between router as serial dce interface.

iv) To configure IP address following commands are executed.




```
PC>ping 10.0.0.3

Pinging 10.0.0.3 with 32 bytes of data:

Reply from 10.0.0.3: bytes=32 time=1ms TTL=128
Reply from 10.0.0.3: bytes=32 time=0ms TTL=128
Reply from 10.0.0.3: bytes=32 time=3ms TTL=128
Reply from 10.0.0.3: bytes=32 time=0ms TTL=128

Ping statistics for 10.0.0.3:
    Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
Approximate round trip times in milli-seconds:
    Minimum = 0ms, Maximum = 3ms, Average = 1ms
```

LAB 6

vi) ping the IP address of all the end devices.

Observation:-

- The messages pinged from an end device to the server or from an end device to another end device is sucessfully sent
- A dynamic IP address is set to end devices when the DHCP is initiated in the server.
- RARP is used to assign the IP address of devices of mac address is known.

Output:-

pc> ping 10.0.0.2

pinging 10.0.0.2 with 32 bytes of data.

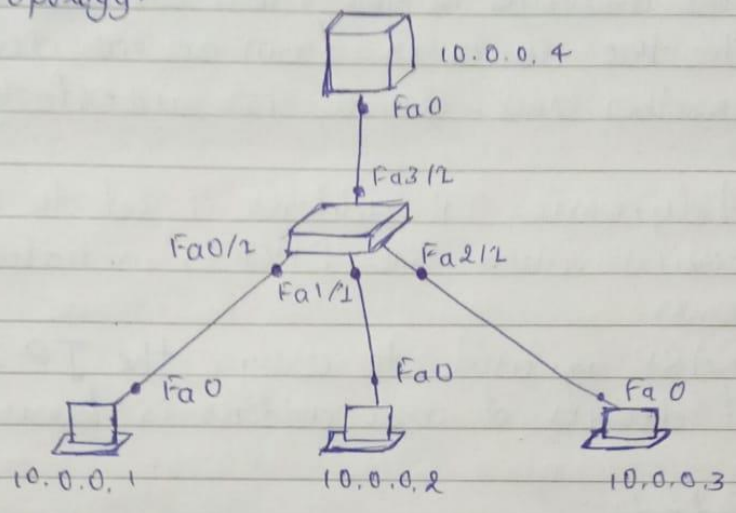
Reply from 10.0.0.2 byte=32 time=0ms TTL=128
reply from 10.0.0.2 bytes=32 time=3ms TTL=128
reply from 10.0.0.2 bytes=32 time=0ms TTL=128

Ping statistics for 10.0.0.1
packets sent=4, recieved=4 &
lost=0(0% loss)

LAB-7

Aim:- Configure DHCP server.

Topology:-



Procedure:-

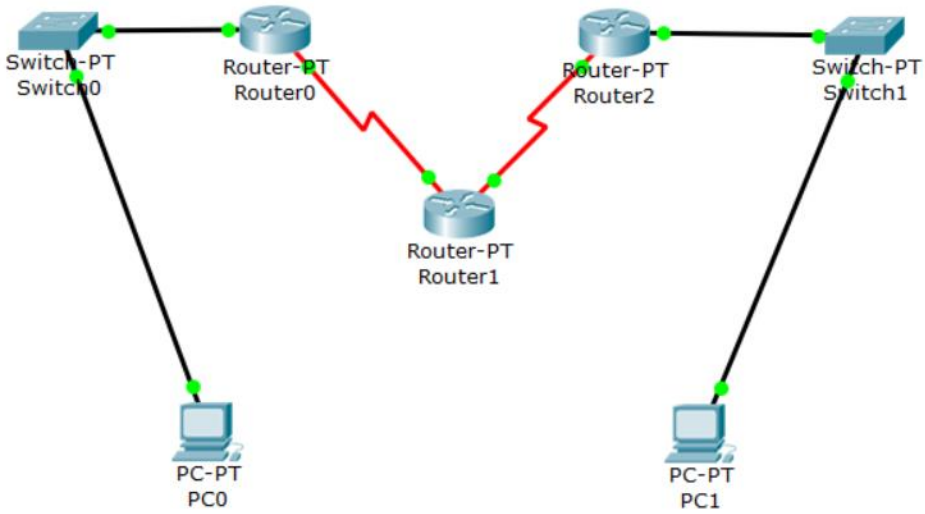
i) 3 end devices are connected to a switch when is then connected to a server.

ii) Configure IP address of server to 10.0.0.4.

iii) Switch on the DHCP server.

iv) add and save the IP address of all the end device in server to then appropriate IP and save them to connect the end devices automatically.

v) After saving the address in server change the gateway/DNS in the end devices to DHCP.




```
PC>ping 40.0.0.2

Pinging 40.0.0.2 with 32 bytes of data:

Reply from 40.0.0.2: bytes=32 time=2ms TTL=125
Reply from 40.0.0.2: bytes=32 time=12ms TTL=125
Reply from 40.0.0.2: bytes=32 time=2ms TTL=125
Reply from 40.0.0.2: bytes=32 time=12ms TTL=125

Ping statistics for 40.0.0.2:
    Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
```

LAB 7

Aim:- How to configure web server & DNS server.

→ From one of the end devices, check if the webserver is reachable from the end devices by entering URL.

Observation:-

→ Web browser module is opened in the end device and the set domain name (www.bmsce.com) is entered.

→ If the system/server hasn't been configured properly i.e., set DNS server and default gateway the Host 'Unresolved' is shown.

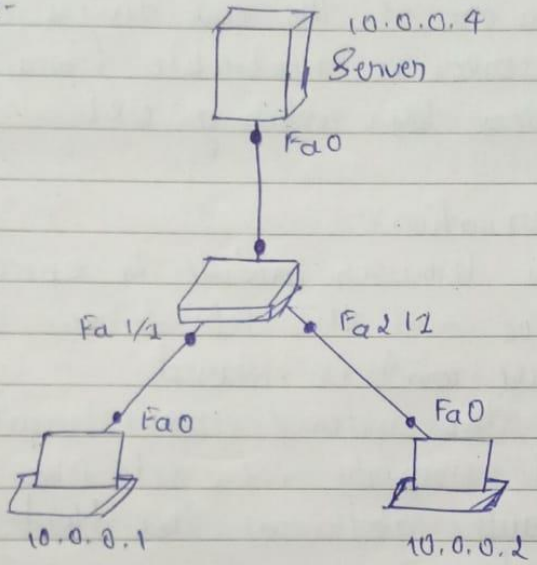
→ If configured properly the page of Cisco packet tracer is opened.

15/12

Lab - 8

Aim :- Configure web server and DNS server.

Topology :-



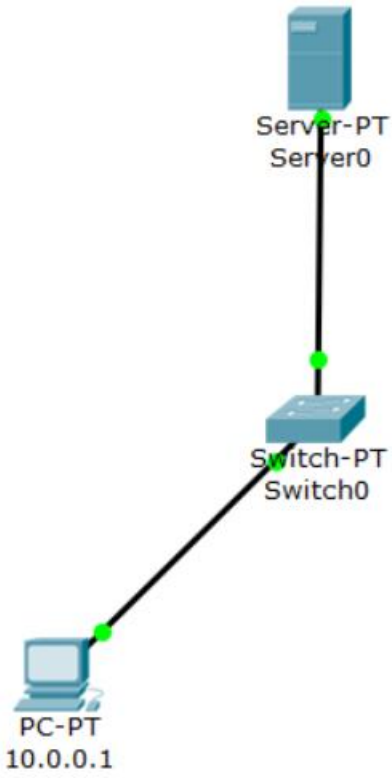
Procedure :-

→ Constant the following topology where Server is connected to end device through switches

→ Configure IP address of server and end devices.

→ Set the HTTP and DNS server to on state.

→ Set the server domain name - and address of it as same as server. add and save the following.


Server-PT
Server0
Switch-PT
Switch0
PC-PT
10.0.0.1


Web Browser
X
<
>
URL
http://csebmsce
Go
Stop
BMSCE
Welcome to BMSCE. Opening doors to new opportunities. Mind Wide Open.
Quick Links:
A small page
Copyrights
Image page
Image

CYCLE 2

LAB 1

29/12/2022

WAP for error detection using CRC-16 bit.

```
import java.util.Scanner;

class Main {
  public static void main (String args[]) {
   Scanner sc = new Scanner (System.in);
   Syso ("Enter the number of bits");
   int n = sc.nextInt();
   int [] data = new int [n+16];
   int[] dup = new int [n];
   int[] g = {1, 0, 0, 0, 1, 0, 0, 0, 0, 0, 0, 1, 0, 0, 0, 0, 1};
   for (int i=0 ; i<n+16; i++) {
    dt data [i] = (i<n)? sc.nextInt() : 0;
    if (i<n)
     dup[i] = data [i];
  }
  for (int i=0; i<n; i++) {
   if (data [i] != 1)
      continue;
   for( int j=0; j<17; j++) {
    data[i+j] = data [i+j] ^ g[j];
    }
   for ( int i=0; i<n; i++) {
    data [i] = dup[i];
  }
  for (int i=0; i<n+16; i++) {
   Syso ( data[i]+" ");
```

```
    }
  }
}
```

Output:

Enter the data to be inserted.

2

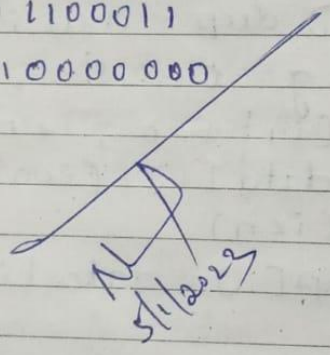
1

1

1

1 1 1 0 1 1 0 0 0 0 0 1 1 0 0 0 1 1

0 0 1 0 0 0 0 0 0 0 1 0 0 0 0 0 0 0



```
PS D:\ENGINEERING\5th sem\cn\lab\CRC>  & 'C:\Program Files\Java\jdk-19\bin\java.exe' '--enable-preview' '-XX:+ShowCodeDetailsInExceptionMessages' '-cp
' 'C:\Users\Lenovo\AppData\Roaming\Code\User\workspaceStorage\d6e0223732a806da462738d5dd655315\redhat.java\jdt_ws\jdt.ls-java-project\bin' 'Lab.CRC.CR
C'
Enter the number of bits :
3
Enter the bits:
1 1 1
Data to be transmitted:
1 1 1 0 1 1 1 0 0 0 0 1 1 1 0 0 1 1 1
Enter the received bits:
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
no error in the received bits
```

LAB 2

Date
Page 12/1/23

Bellman Ford

```
# include <iostream>
#define MAX 10
using namespace std;
typedef struct edge
{
int src;
int dest;
int wt;
}edge;
void bellman_ford(int nv, edge e[], int src_gr
                  -aph, int ne)
{
int u, v, weight, i, j=0;
int dis [MAX];
for ( i=0; i<nv; i++)
{ dis [i]=999;
}
dis [src_graph]=0;
for (i=0; i<nv-1; i++)
{
 for (j=0; j<ne; j++)
{
 u = e[j].src;
 v= e[j].dest;
weight = e[j].wt;
if (dis[u] != 999 && dis[u] + weight < dis[v])
{ dis [v] = dis [u] + weight;
 }}}
```

```
for (j=0; j<ne;j++)
{
 u=e[j].src;
 v=e[j].dest;
weight = e[j].wt;
if (dis[u]+weight <dis[v])
{
cout<<"\n\n Negative Cycle present..!!\n";
return;
}}
cout<<"\n Vertex" << " Distance from source";
for(i=1; i<=nv; i++)
{
 cout<<"\n"<<i<<"\t"<<dis[i];
 }}
int main()
{
int nv, ne, src_graph;
edge e[max];
cout<<"Enter the number of vertices:";
cin>>nv;
cout<<"Enter the source vertex of the graph:";
cin>>src_graph;
cout<<"\n Enter no. of edges:";
cin>>ne;
for(int i=0; i<ne; i++)
{
cout<<"\n for edge "<<i+1<<"=>";
cout<<"\n Enter source vertex:";
```

```
cin >> e[i].src;
cout << "Enter destination vertex:";
cin >> e[i].dest;
cout << "Enter weight:";
cin >> e[i].wt;
}
bellman_ford(nv, e, src_graph, ne);
return 0;
}
```

Output:

```
Enter no. of edges: 4
Edge 1 =>
Source vertex: 1
destination vertex: 2
weight: 3

Edge 2 =>
Source vertex: 2
destination vertex: 3
weight: 5

Edge 3 =>
Source vertex: 1
destination vertex: 4
weight = 4

Edge 4 =>
Source vertex: 4
```

destination vertex: 3
weight: 5

Vertex	Distance from Source
1	0
2	3
3	8
4	4

12/4/2023

```
Enter the bucket capacity :
500
Enter output rate
50
Enter the input rate :
100
Bucket Capacity is 50
Do you want to continue, 2 to exit ,1 to continue
1
Enter the input rate :
250
Bucket Capacity is 250
Do you want to continue, 2 to exit ,1 to continue
1
Enter the input rate :
250
Bucket Capacity is 450
Do you want to continue, 2 to exit ,1 to continue
1
Enter the input rate :
100
Bucket Overflow
Do you want to continue, 2 to exit ,1 to continue
```

LAB 3

Date Page 12/10/23

Dijkstra's Algorithm.

```
# include <iostream>
using namespace std;
# include <limits.h>
# define v 9

int minDistance(int dist[], bool sptSet[])
{
 int min = INT_MAX, min_index;
 for(int v=0; v<V; v++)
  if(sptSet[v] == false && dist[v] <= min)
    min = dist[v], min_index = v;
 return min_index;
}
void printSolution(int dist[])
{
 cout << "Vertex\t Distance from Source" << endl;
 for (int i=0; i<V; i++)
 cout << i << " \t\t\t " << dist[i] << endl;
}
void dijkstra(int graph[V][V], int src)
{
 int dist[V];
 bool sptSet[V];
 for (int i=0; i<V; i++)
  dist[i] = INT_MAX, sptSet[i] = false;
  dist[src] = 0;
 for (int count = 0; count < V-1; count++){
 int u = minDistance(dist, sptSet);
```

```
sptSet[u] = true;
for (int v=0; v<V; v++)
if (!sptSet[v] && graph[u][v]
&& dist[u] != INT_MAX
&& dist[u] + graph[u][v] < dist[v])
dist[v] = dist[u] + graph[u][v];
}
printSolution(dist);
}
int main(){
int graph[V][V] = {{0,4,0,0,0,0,0,8,0},
{4,0,8,0,0,0,0,11,0},
{0,8,0,7,0,4,0,0,2},
{0,0,7,0,9,14,0,0,0},
{0,0,0,9,0,10,0,0,0},
{0,0,4,14,10,0,2,0,0},
{0,0,0,0,0,2,0,1,6},
{8,11,0,0,0,0,1,0,7},
{0,0,2,0,0,0,6,7,0}};
dijkstra(graph,0);
return 0;
}
```

12/1/2023

Output:

Vertex	Distance from Source
0	0
1	4
2	12
3	19
4	21
5	11
6	9
7	8
8	14

```
Vertex          Distance from Source
0               0
1               4
2               12
3               19
4               21
5               11
6               9
7               8
8               14
PS D:\ENGINEERING\5th sem\cn\lab\CRC>
```

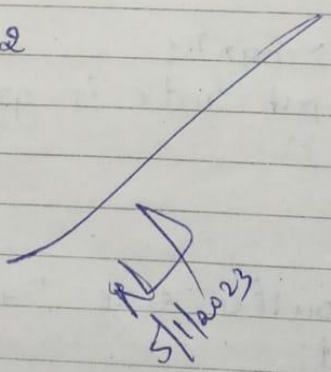
LAB 4

Leaky Bucket Algorithm:

```
import java.util.Scanner;
class main {
  public static void main(String args[]) {
   int size =0;
   Scanner sc = new Scanner(System.in);
   Syso ("Enter the capacity of buffer");
   int max = sc.nextInt();
  System.out.println(" Enter the output rate");
  int out = sc.nextInt();
  while (true){
  Syso.("Enter the input rate");
   int in = sc.nextInt();
   size+=in;
   if (size > max) {
   Syso ("Input rate is greater than buffer
        size");
   size-=in;
  } else {
   Syso (" buffer size: "+size);
   size-=out;
   size = Math.max(size, 0);
   Syso ("buffer size after outflow: "+size);
  }
   Syso ("do you want to continue(1=yes)
        2=no");
  int t = sc.nextInt();
  if(t==2)
```

```
        break;
    }
}
}
```

Output:- Enter the Capacity of buffer
500
Output rate
40
input rate
100
buffer size: 100
after outflow: 60
do you want to continue (1=yes, 2=no)
2



5/12/2023

```
Enter the bucket capacity :
500
Enter output rate
50
Enter the input rate :
100
Bucket Capacity is 50
Do you want to continue, 2 to exit ,1 to continue
1
Enter the input rate :
250
Bucket Capacity is 250
Do you want to continue, 2 to exit ,1 to continue
1
Enter the input rate :
250
Bucket Capacity is 450
Do you want to continue, 2 to exit ,1 to continue
1
Enter the input rate :
100
Bucket Overflow
Do you want to continue, 2 to exit ,1 to continue
```

LAB 5

TCP

1) Client

```
from socket import *
serverName = '127.0.0.1'
serverPort = 12000
```

2

```
clientSocket = socket(AF_INET, SOCK_STREAM)
clientSocket.connect((serverName, serverPort))
sentence = input("\n Enter file name:")
clientSocket.send(sentence.encode())
filecontents = clientSocket.recv(1024).decode()
print("\n From Server:\n")
print(filecontents)
clientSocket.close()
```

Output:

Enter file name : servertcp.py

Reply from Server:

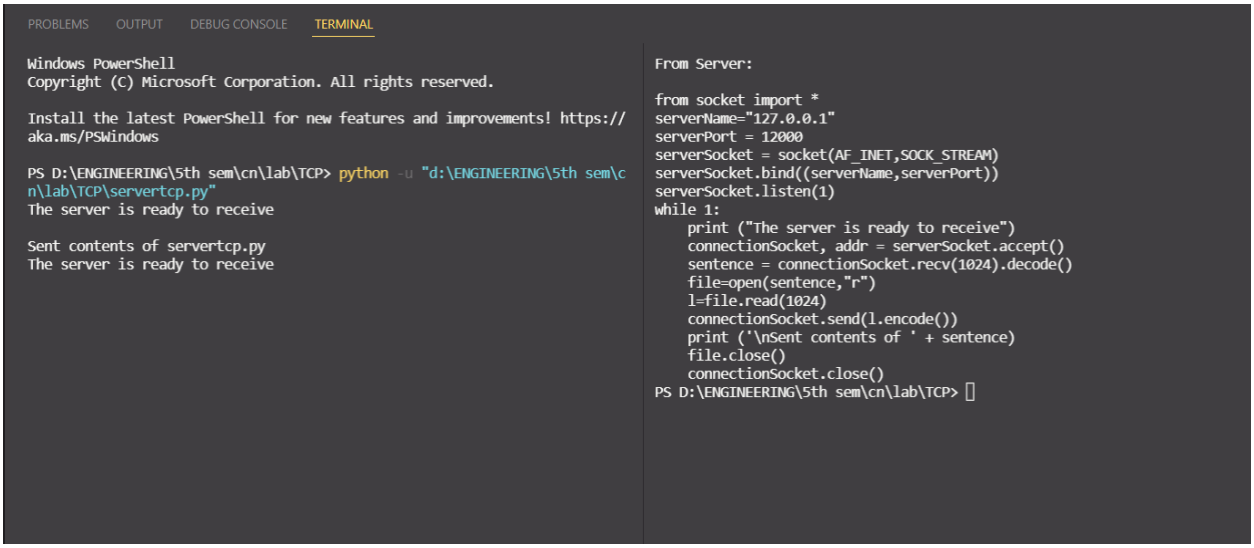
"Contents of servertcp.py file will be displayed"

Server

```
from socket import *
serverName = "127.0.0.1"
serverPort = 12000
serverSocket = socket(AF_INET, SOCK_STREAM)
serverSocket.bind((serverName, serverPort))
serverSocket.listen(1)
while 1:
    print("The server is ready to receive")
    connectionSocket, addr = serverSocket.accept()
    sentence = connectionSocket.recv(1024).decode()
    file = open(sentence, "r")
    l = file.read(1024)
    connectionSocket.send(l.encode())
    print('\n Sent contents of ' + sentence)
    file.close()
    connectionSocket.close()
```

Output:

```
The Server is ready to receive

Sent contents of servertcp.py
The server is ready to receive
```


PROBLEMS
OUTPUT
DEBUG CONSOLE
TERMINAL
Windows PowerShell
Copyright (C) Microsoft Corporation. All rights reserved.
Install the latest PowerShell for new features and improvements! https://aka.ms/PSWindows
PS D:\ENGINEERING\5th sem\cn\lab\TCP> python -u "d:\ENGINEERING\5th sem\cn\lab\TCP\servertcp.py"
The server is ready to receive
Sent contents of servertcp.py
The server is ready to receive
From Server:
from socket import *
serverName="127.0.0.1"
serverPort = 12000
serverSocket = socket(AF_INET,SOCK_STREAM)
serverSocket.bind((serverName,serverPort))
serverSocket.listen(1)
while 1:
print ("The server is ready to receive")
connectionSocket, addr = serverSocket.accept()
sentence = connectionSocket.recv(1024).decode()
file=open(sentence,"r")
l=file.read(1024)
connectionSocket.send(l.encode())
print ('\nSent contents of ' + sentence)
file.close()
connectionSocket.close()
PS D:\ENGINEERING\5th sem\cn\lab\TCP>

LAB 6

UDP

```
from Socket import *
serverName = '127.0.0.1'
serverPort = 12000
clientSocket = socket(AF_INET, SOCK_DGRAM)
sentence = input("\n Enter file name: ")
clientSocket.sendto(bytes(sentence, "utf-8"), (serverName,
                    serverPort))
fileContents, serverAddress = clientSocket.recvfrom(2048)
print ('\n Reply from Server: \n")
print (fileContents.decode("utf-8"))
clientSocket.close()
clientSocket.close()
```

Output:

Enter file name: serverudp.py

Reply from Server:

"Contents of the file serverudp.py
file will be displayed"

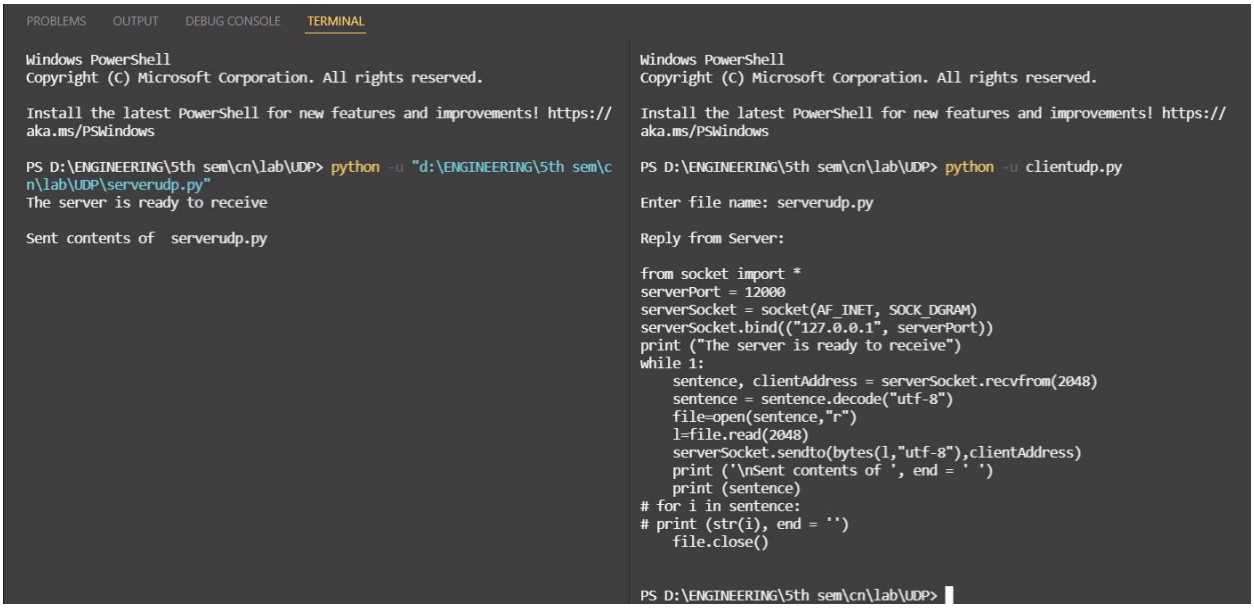
Server

```
fromom saket import *
serverPort = 12000
serverSocket = socket (AF_INET, SOCK_DGRAM)
serverSocket.bind (("127.0.0.1", serverPort))
print ("The server is ready to receive")
while 1 :
  sentence, clientAddress = serverSocket.recvfrom(20
  sentence = sentence.decode ("utf-8")
  file = open (sentence, "r")
  l = file.read (2048)
  serverSocket.sendto (bytes (l, "utf-8"), client
    - Address)
  print ('\n Sent contents of ', end = ' ')
  print (sentence)
  file.close()
```

Output:

The server is ready to receive

Sent contents of servercudp.py


PROBLEMS
OUTPUT
DEBUG CONSOLE
TERMINAL
Windows PowerShell
Copyright (C) Microsoft Corporation. All rights reserved.
Install the latest PowerShell for new features and improvements! https://aka.ms/PSWindows
PS D:\ENGINEERING\5th sem\cn\lab\UDP> python -u "d:\ENGINEERING\5th sem\cn\lab\UDP\serverudp.py"
The server is ready to receive
Sent contents of serverudp.py
Windows PowerShell
Copyright (C) Microsoft Corporation. All rights reserved.
Install the latest PowerShell for new features and improvements! https://aka.ms/PSWindows
PS D:\ENGINEERING\5th sem\cn\lab\UDP> python -u clientudp.py
Enter file name: serverudp.py
Reply from Server:
from socket import *
serverPort = 12000
serverSocket = socket(AF_INET, SOCK_DGRAM)
serverSocket.bind(("127.0.0.1", serverPort))
print ("The server is ready to receive")
while 1:
sentence, clientAddress = serverSocket.recvfrom(2048)
sentence = sentence.decode("utf-8")
file=open(sentence,"r")
l=file.read(2048)
serverSocket.sendto(bytes(l,"utf-8"),clientAddress)
print ('\nSent contents of ', end = ' ')
print (sentence)
# for i in sentence:
# print (str(i), end = '')
file.close()
PS D:\ENGINEERING\5th sem\cn\lab\UDP>