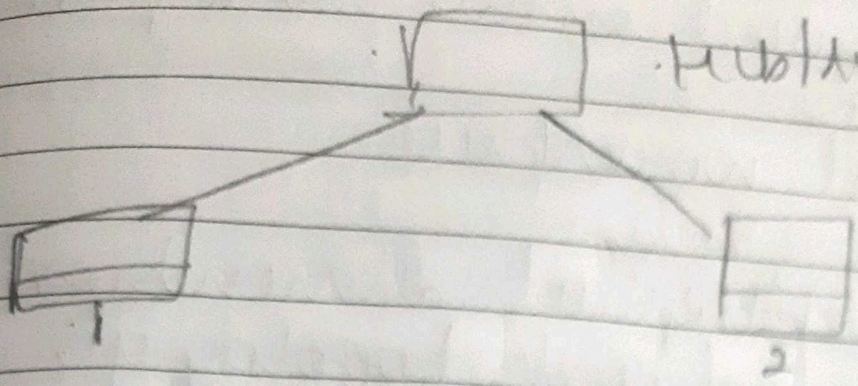
Cisco Packet Tracer Student



Computer Networks.

CISCO Packet Tracer.

Item:-

① Protocols.

LAN, Switching, TCP/IP, Routing, other,
WAN, Security, QoS.

② Logical workspace.

Network topology creation.

Devices: generic, real, & modular with customizable images

Routers, switches, hubs, hosts, bridges, wireless access points, wireless routers, DSL / cable modems

Multiuser remote networks.

- allows student to get a global view o
network through real time or simulation m

The devices are added, linked, configured on this workspace.

③ Physical workspace

- Gives physical dimension & logical network topology.
- Important for wireless LAN as distance parameter is one of the factors to determine if 2 devices able to connect or not & work other device.

④ Realtime mode
⑤ Simulation mode
⑥ ~~Logical~~ Local authoring & Shared

★ file → calls group of 8 bit.
IP address
=
click on device → config : type
IP - addr
=
1st device IP addr = 10.0.0.1
10.0.0.2
Fast ethernet

In simulation
if we add packet / we get, In Play ground
Go to simulation mode

ping → used whether it is connected or not.
Go → to PC-PT PC0 click on that cmd prompt
PC> ping ip address
hub → physical layer.

IP address should be unique for each device.

Step1:-

① Go to Endevice. Selet and choose generic.
PC-PT PC0.
② same again repeat Step1. PC-PT PC1
③ connect it by selecting connections → automatically chose connection type connect it.

click on PC-PT-PC0. go to config → FastEthernet0 write IP address 10.0.0.1 then close it

papergrid
Date:

Step (5) Repeat 4 for PC-PT PC1

step (6) Go to add Simple pdu
click on it.
→ shows successful. in Realtime

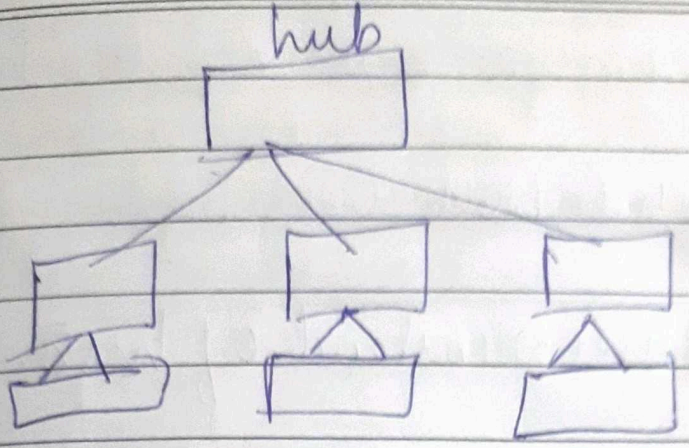
Step (7) Go to simulation mode →
shows In progress

step (8) click on the device Desktop. Select
command prompt.

Type PC > ping 10.0.0.1
PC > ping 10.0.0.2

Query msg → Reply → 4 Query packets
dummy packets to check
whether we can connect it or not.

Date: / /



Step 1:- 4 device → PC-PT
- IPaddress 10.0.0.1 - - -
- - - Same Net.
→ broadcasting the network.

Step 2:- hub → Generic.
Hub-PT.

Step 3:- Normal command prompt

Step 4:- Cisco. prompt Ipaddress.

edit filter :- Select ARP and ICMP

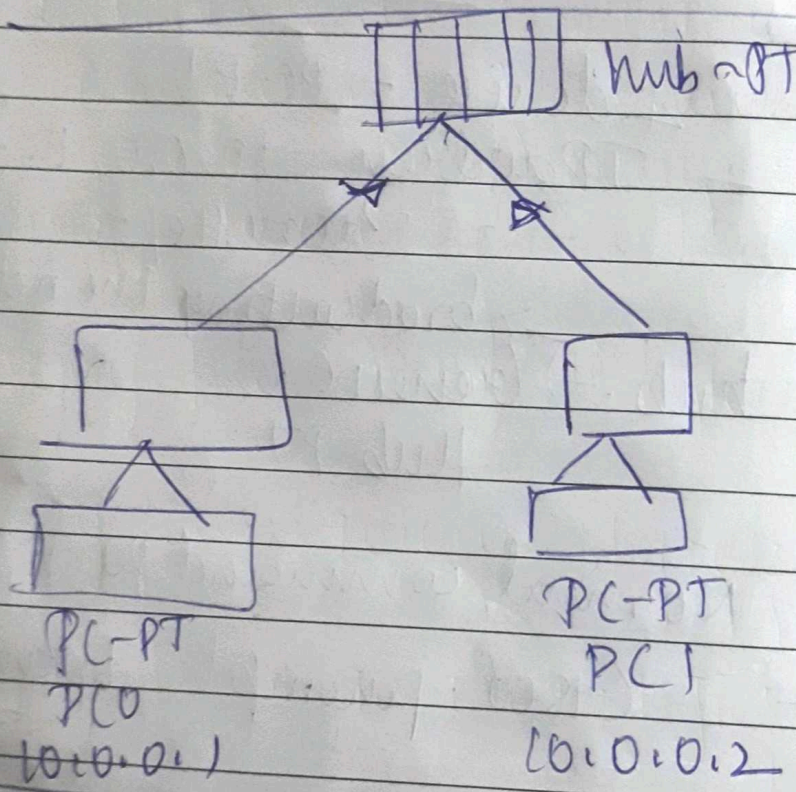
then capture | Forward.

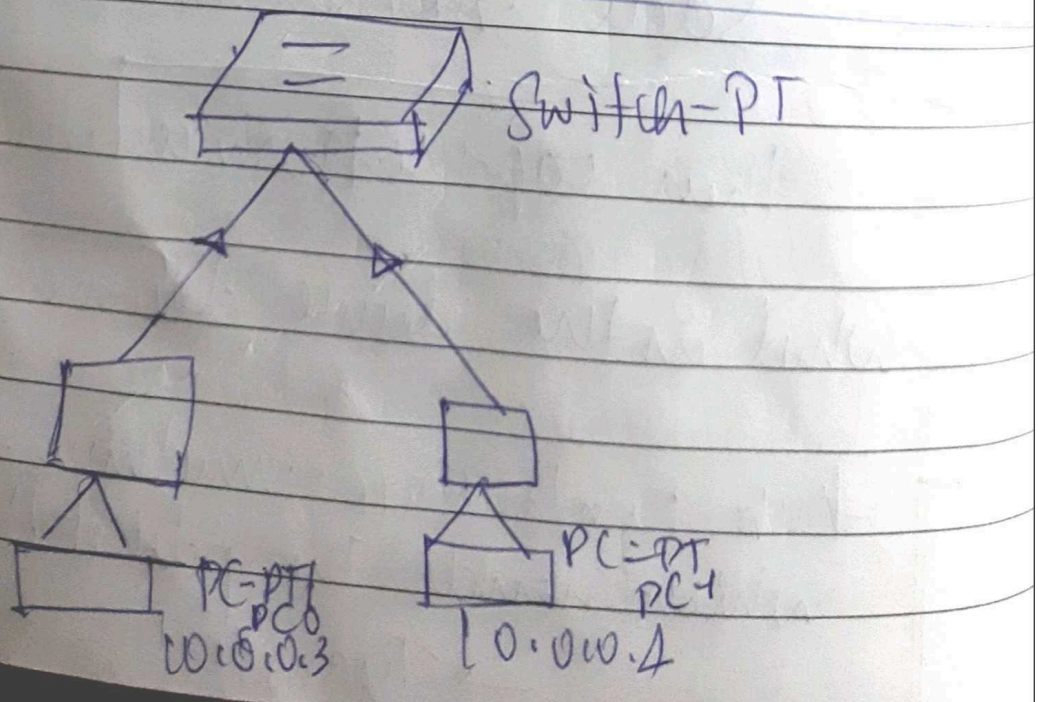
① what is the device decision in this layer.

- Hub can't store anything.
which sends the packet that Ethar. received.

Demonstration

2) Swith → instead of hub.





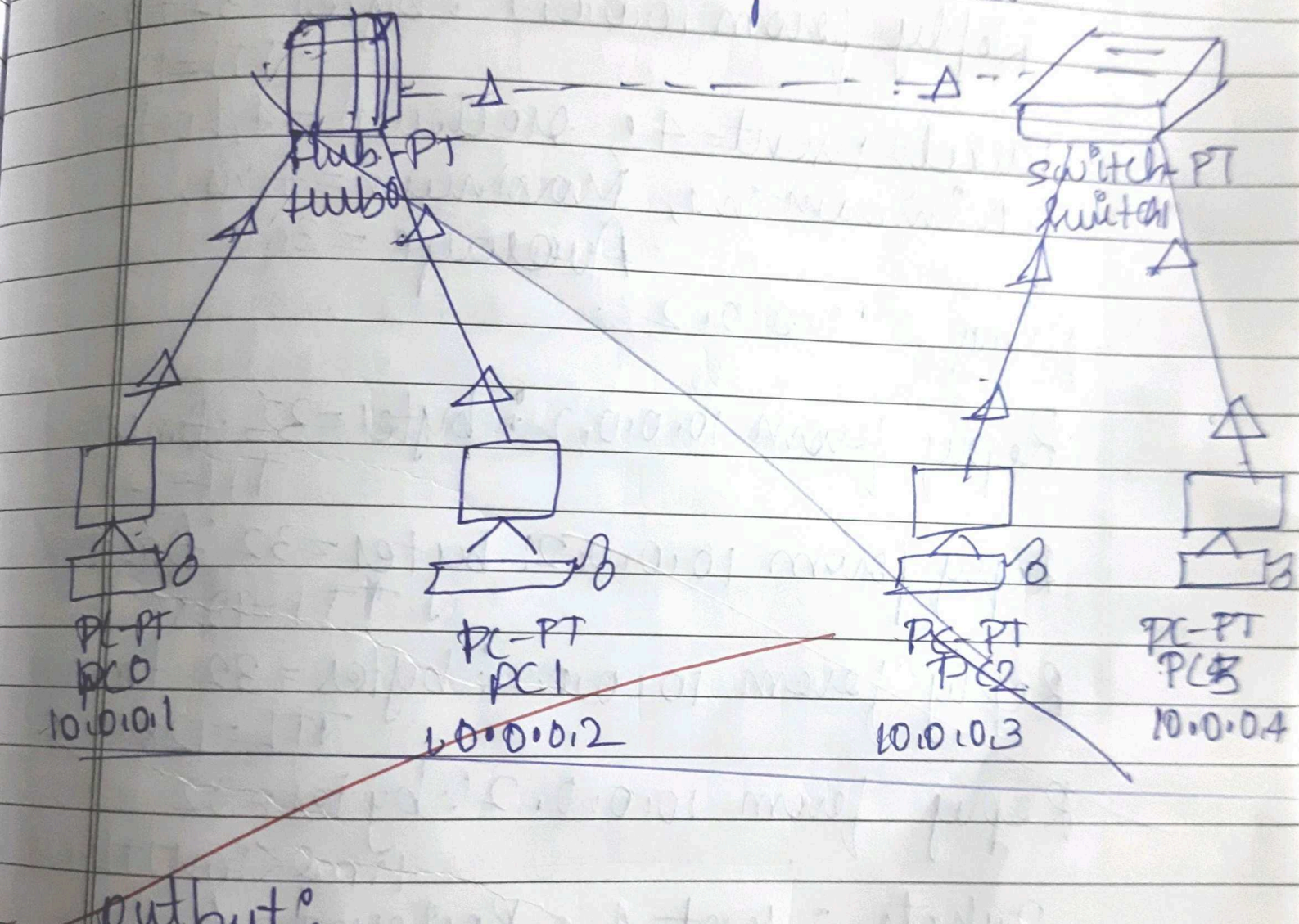
10/11/22

Lab-1

papergrid

Date: / /

1. Creating a topology and simulate sending a simple PDU from source to destination using hub and switch as connecting devices.



Output:

> ping 10.0.0.1

Reply from 10.0.0.1: bytes=32 time=7ms TTL=128

Reply from 10.0.0.1: bytes=32 time
TTL=1
Reply from 10.0.0.1: bytes=32 time
TTL
Reply from 10.0.0.1: bytes=32
TTL=1
Packets: sent=4, received=4, Lo
Minimum=0ms, Maximum=7ms,
Average=3ms

> ping 10.0.0.2

Reply from 10.0.0.2: bytes=32 time
TTL=128
Reply from 10.0.0.2: bytes=32 time=
TTL=128
Reply from 10.0.0.2: bytes=32 time=
TTL=128
Reply from 10.0.0.2: bytes=32
time<1ms TTL=1
Packets: sent=4, Received=4,
Lost=0 (0% loss)

Minimum=0ms, Maximum=1ms,
Average=0ms

steps:-

step1: Start → c drive → program files(x86) → cisco packet tracer 6.2sv → help → default → index

step2: go to work space → select the device type → insert the device → select the (switch, Routers, hub) → select connection → establish the connection → Green (connection build), orange (getting connected), Red (not connected).

Step3: Right click on device → Go to the fast ethernet() → set IP address → Subnet mask address generated. IP address like 192.168.0.104 and 192.168.0.103

Step4: Go to common tool bar → select packet which send to source to destination.

Step 5: place it on source → and destination.
→ Go to real time → Simulation
→ click on auto capture play
see the status → successful (then transaction is successful else failed)

Step 6: Go to Simulation → Capture forward. Observe the movement of packets from one device. (closer) to destination. another

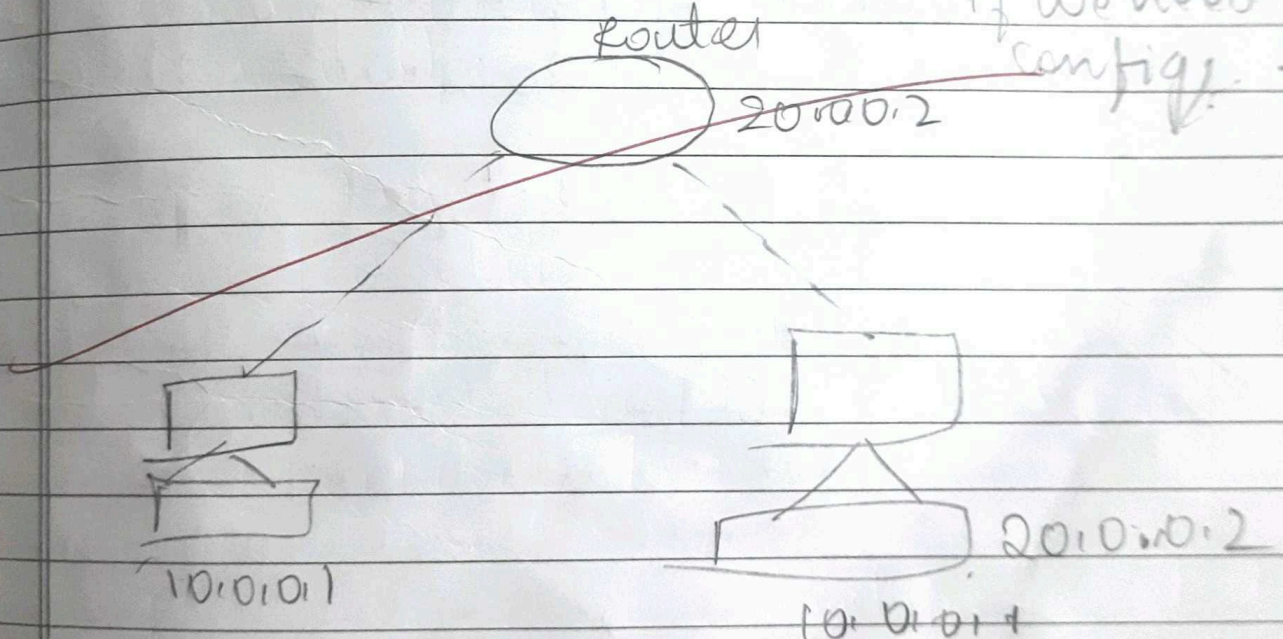
17/11/22

LAB-2

2) Configuring IP address to Routers in Packet Tracer. Explore the following messages: ping response, Destination unreachable, Request timed out, Reply.

PC-PT-PC0 → 10.0.0.1
PC-PT-PC1 → 20.0.0.2

Router will have 9+1 own interfaces
Router → CLI
If we need to config. → say no



Step1: Router enable

default gateway

↳ Go to PC → gate

↳ for the first device

ipaddress → 10.0.0.1

Gateway → 10.0.0.1 → another 20.

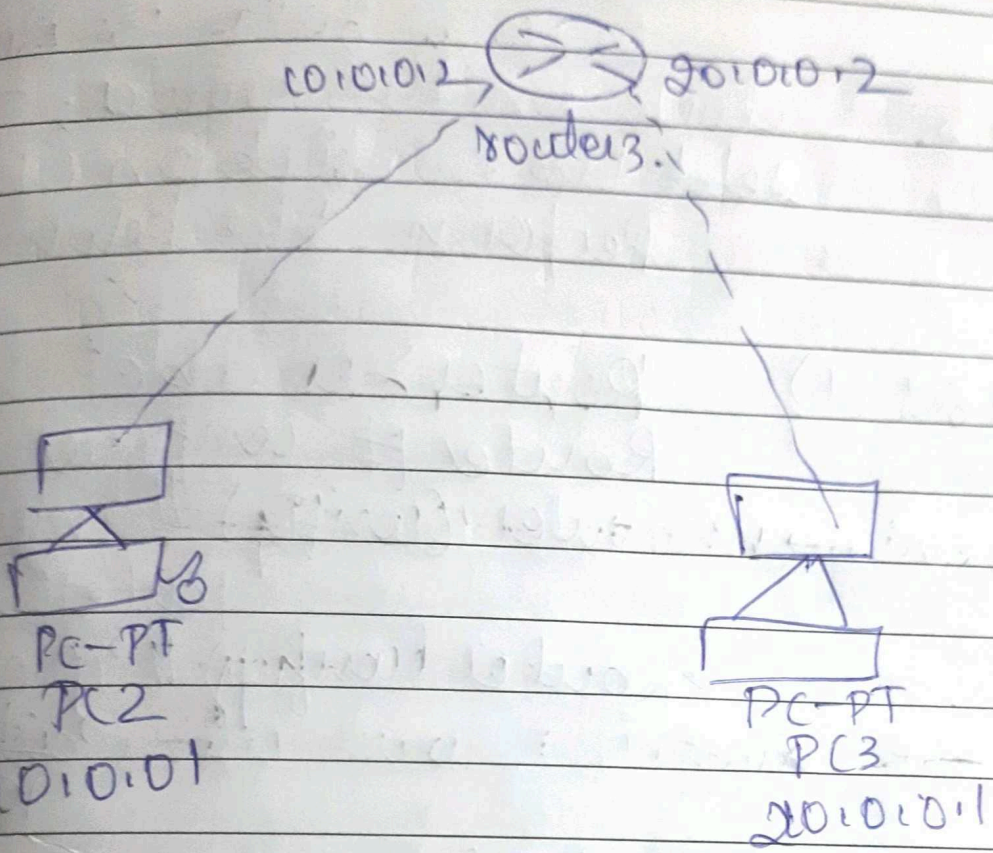
IPaddress → PC0 10.0.0.1

IPaddress → PC1 → 10.0.0.2

→ #Router → 10.0.0.2 → PC1

→ # Router → 20.0.0.2 → PC

Router:



Step1: Select the router and the place it and then select the end devices and place it.

Step2: Connect the end devices to the router.

Step3: Go to PC-PT PC2, Go to → config → fastethernet0 → Set IP address.

Date: / /

Step 4: ~~rep~~ do the same for PC-PTPC3 set I[illegible]

Step 5: click on the router, select the click on C[illegible] Perform the below

Step 6: i) Router>enable
Router# configure [illegible]
Router (config)# interface FastEthernet[illegible]
Router (config)# ip address 10.[illegible]

~~Router(config-if)# no shutdown~~

~~i, then the connection between the router and the end device turns to green.~~

Router(config-if)#ip address 20.10.10.[illegible] 255.10.10.10

Router(config-if)# no shutdown.

Step6:- Set the gateway IP address for both the end devices

Step7) Go to simulation, add simple PDU, then click on the source and the destination. to see the movement of packets sent click on capture forward.

In command prompt.

PC>ping 10.0.0.2 (router — —)

Reply from 10.0.0.2 byte=32 time=12ms TTL=12
Reply from 10.0.0.2 byte=32 time=0ms TTL=12
Reply from 10.0.0.2 byte=32 time=1ms TTL=12
Reply from 10.0.0.2 byte=32 time=0ms TTL=12

Packet Set=4, Received=4 Lost=0.

Minimum=0ms Maximum=1ms Average=0ms

papergrid

Date:

PC > Ping 20.0.0.2

Reply from 20.0.0.2 bytes=32 time
Reply from 20.0.0.2 bytes=32
Reply from 20.0.0.2 bytes=32
Reply from 20.0.0.2 bytes=32

packets Set=4, Received=4, lo

Minimum=0 maximum=1ms
and average=0ms

Ping 20.0.0.1 with 32 byte
of data:
Request timed out.
Request timed out.
Request timed out.
Request timed out

24/4/22

papergrid

Date: / /

TTL=128
TTL=128
TTL=128
TTL=128

Lab-3

Aim:

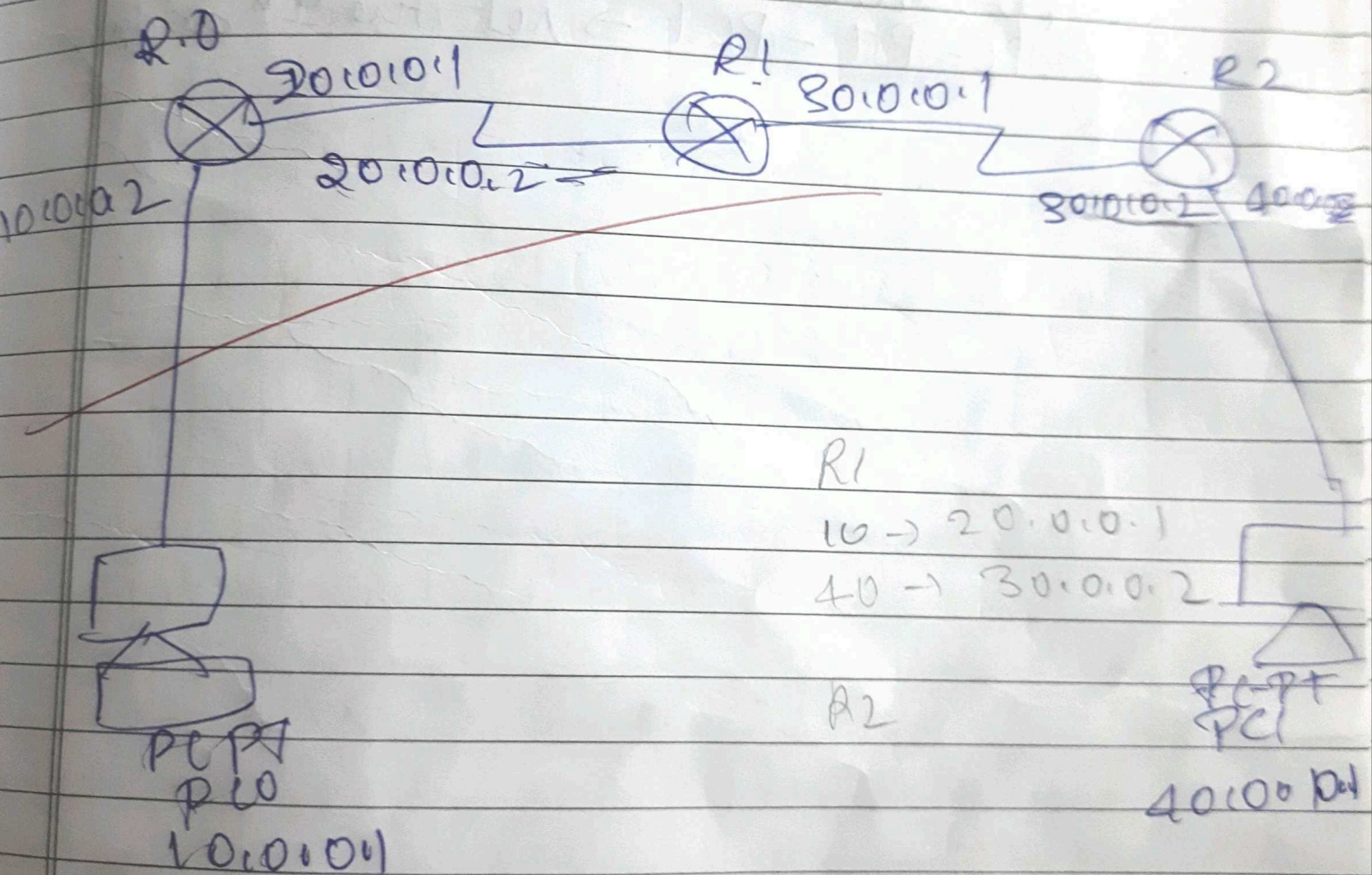
Configuring default, static Router to Router.

PC0 - IP address : 10.0.0.1

PC1 - IP address → 20.0.0.1

Router 0 → 10.0.0.2

Router 2 → 40.0.0.2



Step1:- Select the 3 routers and build connect the connection.
router.
with serially connect the devices to the one router node end device to the router.

Step2:- click on PC-PT → PC0 → IP address do the same process PC-PT-PC1 → Set the IP address

Step3: Router - ① configuration.

Router(config)# interface fastethernet 0/0.
Router(config)# ip address 10.0.0.2
255.0.0.0

R(config)# no shutdown.
Router(config)# exit
Router(config)# interface serial 2/0.
Router(config)# ip address 20.0.0.1
255.0.0.0.

Router(config-if)# no shut down.
Router(config)# ip route 30.0.0.0
255.0.0.0.
20.0.0.0

Router(config)# ip route 40.0.0.0
255.0.0.0 20.0.0.2.

Note: syntax ip route destination subnet nexthop address mask address.

Router # show ip route.
C 10.0.0.0/8 is directly connected.
C 20.0.0.0/8 is directly connected 2/0
S 30.0.0.0/8 [1/0] via 20.0.0.2
S 40.0.0.0/8 [1/0] via 20.0.0.2

Router - 1 configuration.

Router(config)# interface serial 2/0
Router(config)# ip address 20.0.0.2 255.0.0.0
Router(config-if)# no shutdown.
Router(config-if)# exit.

Router(config)# interface serial 3/0
Router(config-if)# ip address 30.0.0.2 255.0.0.0
Router(config-if)# no shutdown.
Router(config-if)# exit.
Router(config)# ip route 10.0.0.0 255.0.0.0
20.0.0.1

Router(config)# ip route 40.0.0.0 255.0.0.0
30.0.0.1

Router - 2 configurations

continue with configuration dialog
[yes/no]? no

Router > enable.
Router # config terminal.

Router(config)# interface serial 3/0
Router(config-if)# ip address 30.0.0.
2

255.0.0.0

Router(config-if)# no shutdown
Router(config-if)# exit
Router(config)# interface serial 4/0
Router(config-if)# ip address 30.0.0.1
255.0.0.0
Router(config-if)# no shutdown
Router(config-if)# exit

ip route 10.0.0.0 255.0.0.0 20.0.0.1
ip route 40.0.0.0 255.0.0.0 30.0.0.2

PC> ping 40.0.0.1
Pinging 40.0.0.1 with 32 bytes of data

Reply from 40.0.0.1 byte=32 time=4m
Reply from 40.0.0.1 byte=32 time=6m
Reply from 40.0.0.1 byte=32 time=2
Reply from 40.0.0.1 byte=32 time=2

statistic.

cert = 0 : Approximate round trip time in milliseconds
minimum = 9ms, Avg = 1ms

PDU from source end device successfully received by destination end device (10.0.0.1)

Ping 10.0.0.1

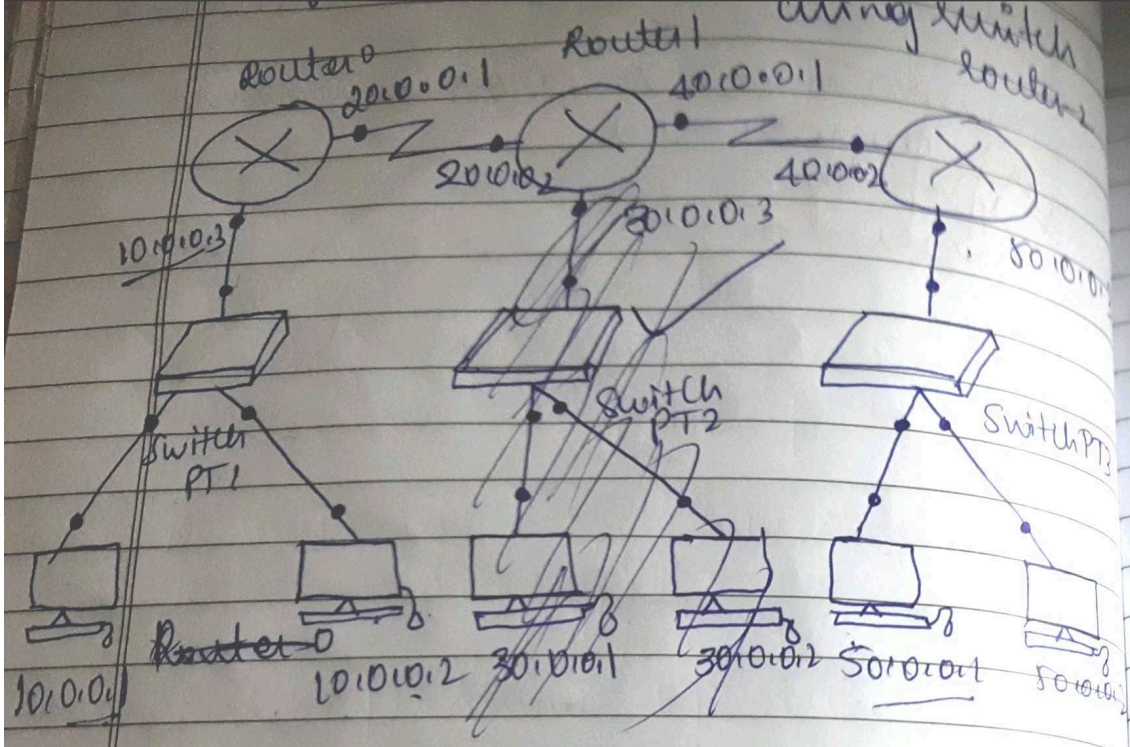
Pinging 10.0.0.1 with 32 bytes of data:

Request timed out.
Request timed out.
Request timed out.
Request timed out.

Ping 10.0.0.1

Pinging 10.0.0.1 with 32 bytes of data:

Reply from 10.0.0.1: bytes=32 time=3ms
Reply from 10.0.0.1: bytes=32 time=1ms
Reply from 10.0.0.1: bytes=32 time=1ms
Reply from 10.0.0.1: bytes=32 time=2ms



Router 0 configuration

Router > enable

Router # config terminal.

Router (config) # interface fastethernet 0/0

Router (config-if) # ip address 10.0.0.1 255.0.0.0

Router (config-if) # no shutdown

Router (config-if) # exit

Router (config) # interface serial 2/0

Router 1 configuration

Router > enable

Router # config terminal.

Router (config) # interface serial 2/0

Router (config-if) # ip address 30.0.0.2 255.0.0.0.

Router (config-if) # no shutdown.

Router (config-if) # exit

Router (config) # interface fastethernet 0/0.

Router (config-if) # ip address 40.0.0.2 255.0.0.0

Router (config-if) # no shutdown.

Router (config-if) # exit

Router (config) # exit.

Router # show ip route

Router # config terminal.

Router (config) # ip route 0.0.0.0 0.0.0.0 30.0.0.1

Router (config) # exit.

Router 2 configuration.

Router > enable

Router # config terminal.

Router (config) # interface serial 3/0

Router (config-if) # ip address 20.0.0.2 255.0.0.0

Router (config-if) # no shutdown.

Router (config-if) # exit.

Router (config) # interface serial 3/0.

Router (config-if) # ip address 30.0.0.1 255.0.0.0

Router (config-if) # no shutdown.

Router (config-if) # exit

Router (config) # exit

Router # show ip route.

Router # config terminal.

Router (config) # ip route 10.0.0.0 255.0.0.0 20.0.0.2

Router (config) # ip route 40.0.0.0 255.0.0.0 30.0.0.2

Router (config) # exit.

Command Prompt:

C:\> ping 10.0.0.1

papergrid

Date:

pinging 10.0.0.1 with 32 bytes of data:

Reply from 10.0.0.1: bytes = 32 time = 15ms TTL = 128
Reply from 10.0.0.1 bytes = 32 time = 4ms TTL = 255
Reply from 10.0.0.1 bytes = 32 time = 4ms TTL = 255
Reply from 10.0.0.1 bytes = 32 time = 6ms TTL = 255
Packets: Sent = 4, Received = 4, Lost = 0 (0% Loss)

pinging 20.0.0.2 with 32 bytes of data:

Reply from 20.0.0.2 bytes = 32 time = 5ms TTL = 255
Reply from 20.0.0.2 bytes = 32 time = 4ms TTL = 255
Reply from 20.0.0.2 bytes = 32 time = 5ms TTL = 255
Reply from 20.0.0.2 bytes = 32 time = 6ms TTL = 255

packets = 4, Received = 4, Lost = 0 (0% loss).

Pinging 10.0.0.1 out with 32 bytes of data.

Request timed out.
Request timed out
Request timed out
Request timed out.

Appropriate round trip times in ms:
minimum = 4ms, maximum = 15ms, Average = 9ms.

observation: we need to give hop address for each of the routers for the PDU to be able to reach any of the end devices.

hop address → Ip address of router that is not connected directly.

8/12/22 default router

Lab 5

DHCP.

Server → Services → DHCP

DHCP service → off → ON

↓ DHCP

start Ip address

↳ 10.0.0 [3]

Subnet mask. 255.0.0.0

max → no of users → default

DNS Server → same ip address as server address.

Trivial file transfer protocol.

TFTP server: 10.0.0.2 → server

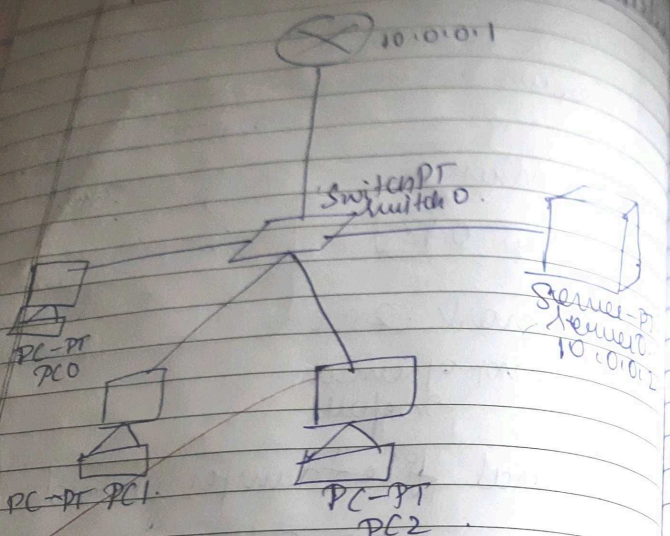
☆ Do not give IP address in the starting

Next.

Desktop → ip configuration → • DCHP ○ Static

D - Discover
O - Offer.
R - Request
A - Acknowledgement

Dynamic host configuration protocol



Aim: configuring DHCP.

Aim:

select generic router, server connect it to the end devices

configure the router → 10.0.0.1
configure the server → 10.0.0.2
click on server → services → DHCP
switch on the service
set the default gateway to the router and address → 10.0.0.1

set DNS server with the ip address of server
↳ 10.0.0.2

start ip address → 10.0.0.3
subnet mask → 255.0.0.0

maximum number of user = 512
TFTP server = 10.0.0.2
→ save.

We use DHCP to automatically assign addresses to the end devices using the DORA technique.
D discover offer request acknowledgement

papergrid
Date: / /

Now we go to the end device
> Desktop > ip configuration
Then click on DHCP to see our ip address automatically assigned.

Observation:-
By adding a server we can ~~other~~ off ip addresses to end devices

papergrid
Date: / /

it selects the best path.
max - 15 hop count
if hop count < 15, if we select the best hop -

RIP

#enable
#config t
#ip address
#encapsulation PPP
#clock rate 64000
#no shutdown
#exit
#config →
#network 10.0.0.0
#network 20.0.0.0
#exit

only one side where symbol enter

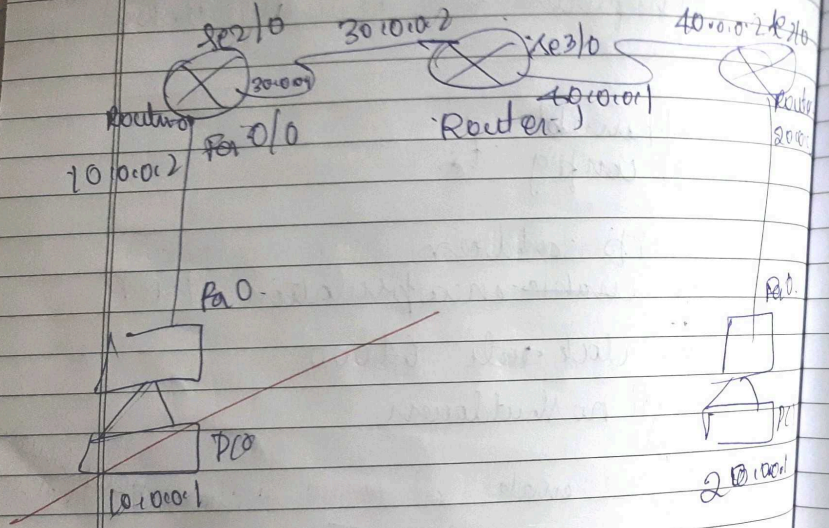
neighbouring network

finds the shortest path (for all the routers)

15/12/22

LAB 5

AIM: Configuring RIP routing Protocol in Router.



connect the end devices and the routers as shown.

Initially configure IP address of the end devices and fast ethernet.

Then for serial configuration in routers. Router the clock

enable
configure terminal
interface serial 2/0
~~interface 2/0~~

Router(config-if)# ip address 30.0.0.1 255.0
Router(config-if)# encapsulation PPP.
clock ~~router~~ rate 64000.

no shutdown

If shown no clock → no need of clock rate
Set the gateway for the PCs.
To set RIP, give ip address of neighbor devices

Router(config)# router rip
network 10.0.0.0
network 30.0.0.0
exit

ping 20.0.0.1

> Ping 20.0.0.1.

Reply from 20.0.0.1 byte = 32 time<1ms TTL=128.

Reply from 20.0.0.1 byte = 32 time=2ms TTL=125

Reply from 20.0.0.1 byte=32 time=2ms TTL=225

Reply from 20.0.0.1 byte=32 time=2ms TTL=225

Pakets: Sent=4, Received = 4, Lost=0 (0% Loss)

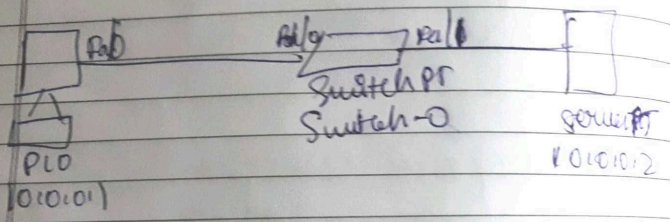
minimum = 2ms, Maximum = 1.9ms, Average = 7ms

Observation:

when we set the trip route, we don't have to give the hop address for each of

LAB-5

AIM: Demonstration of WEB Server and DNS using Packet tracer



connect the end devices, switch and server as shown.

Configure the ip address of the end devices and server.

Then go to server → Services → HTTP → On

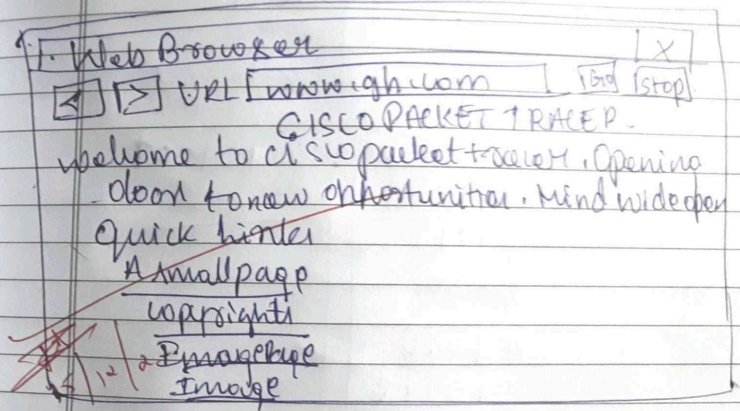
Then go to DNS → DNS service on. Give the Name of the end. address → 10.0.0.2

Click on Add and save it.

Go to Desktop → web browser

→ URL → www.gh.com.

Observation: The URL given in the server can be accessed via the web browsers in the PC.



29/12/22

LAB-6

1. Write a Program for error detecting code using CRC-CCITT (16 bit)

```
#include<stdio.h>
#include<string.h>
#define N strlen(gen_poly)
char data[28];
char check_value[28];
char gen_poly[10];
int data_length,i,j;
void XOR(){
    for(j=1;j<N;j++)
    check_value[j]=((check_value[j]==
                    gen_poly[j])?'0':'1');
}
void receiver(){
    printf("Enter the received data:");
    scanf("%s",data);
    printf("\n-----\n");
    printf("Data received:%s",data);
    recv();
    for(i=0;(i<N-1)&&(check_value[i]!=
                    '1');i++);
```

```
if (i<N-1)
else  pf("\nError detected\n\n");
   pf("\nNo error detected.\n\n");
}

void crc(){
   for(i=0; i<N; i++)
       check_value[i]=data[i];
   do{
     if(check_value[0]=='1')
          XOR();
     for(j=0; j<N-1; j++)
       check_value[j]=check_value[j+1];

     check_value[j]=data[i++];
   }while(i<=data_length+N-1);
}

int main()
{  pf("Enter data to be transmitted:");
   sf("%s", data);
   pf("\n Enter the generating polynomial:");
   sf("%s", gen_poly);
   data_length = strlen(data);
   for(i=data_length; i<data_length+N-1; i++)
       data[i]='0';
   pf("\n----------------\n");
   pf("Data padded with n-1 zeros: %s", data);

   crc();
   pf("CRC or check value is: %s", check_value);
   for(i=data_length; i<data_length+N-1; i++)

       data[i]=check_value[i-data_length];
   pf("\n----------------\n");
   pf("\n Final data to be sent: %s", data);
   pf("\n----------------\n");
   receiver();
   return 0;
}
```

O/P

Enter the data to be transmitted: 1101011011
Enter Generating poly: 10011
Data padded with n-1 zeros: 11010110110000
CRC or check value is: 1110
Final data to be sent: 11010110111110
enter the received data: 11010110111110
Data received: 11010110111110
No error detected.

LAB-9

Date: / /

(2) Implement Dijkstra's algorithm to compute the shortest path for a given topology.

```
#include <limits.h>
#include <stdio.h>
#include <stdbool.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 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);
}

void printSolution(int dist[])
{
    printf("Vertex \t\t Distance from src\n");
    for (int i = 0; i < V; i++)
        printf("%d \t\t %d\n", i, dist[i]);
}

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} };
```

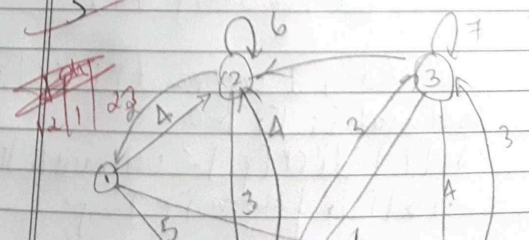
```
{0,4,14,10,0,2,0,0,0},{0,0,0,0,0,
2,0,1,6},{8,11,0,0,0,0,1,0,7},
{0,0,2,0,0,0,6,7,0}};
dijkshtra(graph,0);
return 0;
}
```

olb

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

(or)

```
void main()
{
  int i,j;
  printf("\n Enter the no of vertices:");
  scanf("%d",&n);
  printf("\n Enter the cost matrix:\n");
  for(i=1;i<=n;i++)
  {
    for(j=1;j<=n;j++)
    {
      scanf("%d",&c[i][j]);
    }
  }
  printf("\n Enter the source node:");
  scanf("%d",&src);
  dijkshtra();
}
```



LAB-10

```
#include <stdio.h>
struct node
{
    unsigned dist[20];
    unsigned from[20];
} rt[10];
int main()
{
    int costmat[20][20];
    int nodes, i, j, k, count=0;
    pf("Enter the no. of nodes:");
    sf("%d", &nodes);
    pf("Enter the cost matrix:\n");
    for(i=0; i<nodes; i++)
    {
        for(j=0; j<nodes; j++)
        {
            sf("%d", &costmat[i][j]);
            costmat[i][i]=0;
            rt[i].dist[j]=costmat[i][j];
            rt[i].from[j]=j;
        }
    }
    do
    {
    count=0
    for(i=0; i<nodes; i++)
    for(j=0; j<nodes; j++)
    for(k=0; k<nodes; k++)
        if(rt[i].dist[j]>costmat[i][k]+rt[k].dist[j])
    {
        rt[i].dist[j]=rt[i][k]+rt[k].dist[j];
        rt[i].from[j]=k;
        count++;
    }
    } while(count!=0);
    for(i=0; i<nodes; i++)
    {
        printf("\n\n For router %d\n", i+1);
        for(j=0; j<nodes; j++)
        {pf("node %d via %d distance %d", j+1, rt[i].from[j]+1, rt[i].dist[j]);
    }
    pf("\n\n");
    print(" cost is %d", cos
    getch();
```

o/p:- Enter the number of nodes 3
Enter the cost matrix
0 2 7
2 0 1

LAB-11

We can't transfer the packets which is greater than leak rate

n = leak rate = 1000
data = 400
400 <= n ∴
n = 1000 - 400
bucket = [500, 100], 500 <= 600
n = (600 - 500) = 100

```
#include <iostream>
using namespace std;

int main()
{
    int capacity=0, bsize=0, packet=0,
        rate=0;
    char ans = 'y';
    cout << "Enter the bucket size";
    cin >> capacity;
    cout << "Enter the leaking rate";
    cin >> rate
    while (ans == 'y')
    {
        cout << "Enter the packet size";
        cin >> packet;
```

Date: / /

LAB-12 [TCP] Socket:

Server.py

```
from socket import *
serverName = 'DESKTOP-9CJOB77'
serverPort=12530
serverSocket=socket(AF_INET,SOCK_STREAM)
serverSocket.bind((serverName,serverPort))
serverSocket.listen(1)
print("The server is ready to receive")
while(1):
    connectionSocket,addr=serverSocket.accept()
    sentence=connectionSocket.recv(1024).decode()
    file=open(sentence,"r")
    l=file.read(1024)
    connectionSocket.send(l.encode())
    file.close()
    connectionSocket.close()
```

papergrid
Date / /

LAB-12 [TCP] Socket:

Server.py

```
from socket import *
serverName = 'DESKTOP-9CJOB7I'
serverPort = 12530
serverSocket = socket(AF_INET, SOCK_STREAM)
serverSocket.bind((serverName, serverPort))
serverSocket.listen(1)
print("The server is ready to receive")
while(1):
    connectionSocket, addr = serverSocket.accept()
    sentence = connectionSocket.recv(1024).decode()
    file = open(sentence, "r")
    l = file.read(1024)
    connectionSocket.send(l.encode())
    file.close()
    connectionSocket.close()
```

client.py

```
from socket import*
ServerName = 'DESKTOP-9CJOB77'
ServerPort=12530
clientSocket=socket(AF_INET,SOCK_STREAM)
clientSocket.connect((ServerName,ServerPort
Sentence =input("Enter fileName")
clientSocket.send(send.encode())
fileconents=clientSocket.recv(1024).decode
print('From Server:', fileconents)
clientSocket.close()
```

Date: / /

UDP

Server.py

```
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('\n Sent contents of', end=' ')
    print (sentence)
    # for i in sentence:
    # print (str(i), end=' ')
    file.close()
```

client.py

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