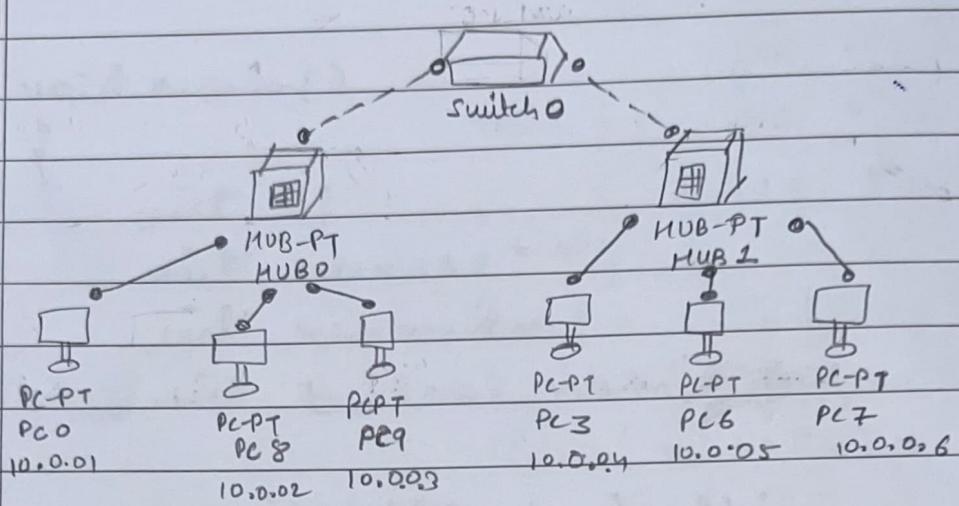


**TASK 1:** Create a topology ~~and~~ involving multiple hubs & a switch connecting them to stimulator with simple PDU [Packet data unit]

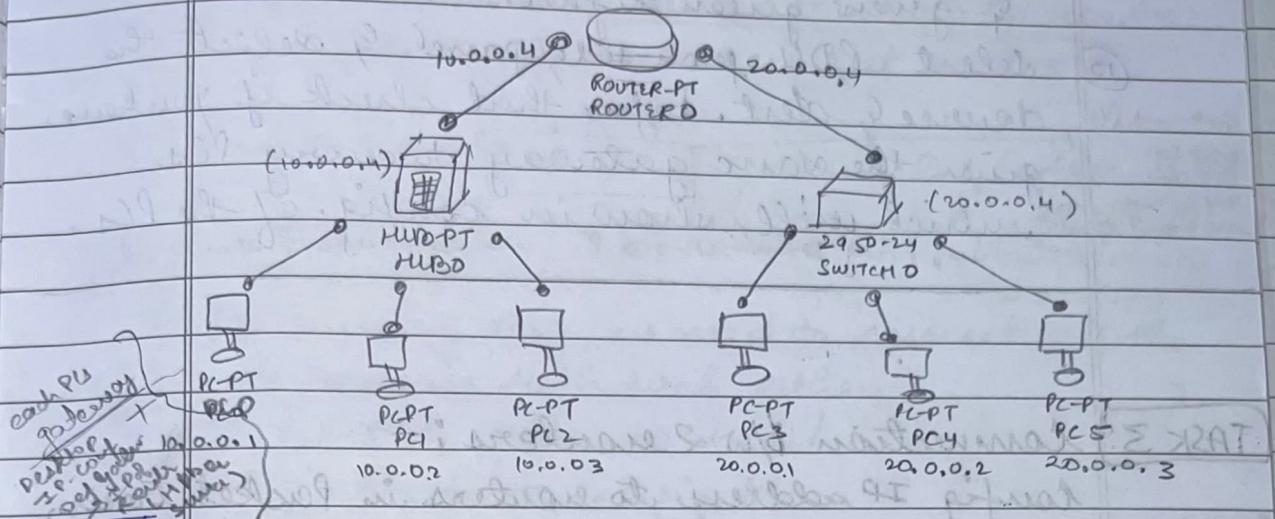


Steps:

- ① drag switch from the switch panel & drop it.
- ② drag hub from hubs panel as hub is connecting device so, no need to configure & drop to the window
- ③ Connect - Switch & Hub - through Copper-Cross Over
- ④ drag PCs from end device config & give IP address to them. Then connect with HUB - through Copper - straight - through
- ⑤ Select PDU from side panel & select the source destination place
- ⑥ Observe the packet transition in - Simulation Mode
- ⑦ for ping - command, go to desktop → → Cmd-Rouupt → Run.  
ping - 10.0.0.1

## Task 2

: Configure IP Address to routers in packet tracer using Hubs & switch



**Steps:** ① Drag Router - from Router panel, name Router-PT & drop it.

- ② Drag HUB - from HUB panel & drop it.
- ③ Drag Switch - from Switch panel & drop it.
- ④ Drag PCs - from End devices & drop it.
- ⑤ Connect - Routers, Hub & ~~Router~~ Switch - from Copper-Straight Through - present in Connection Panel
- ⑥ Click on the PCs → Configure → IP Address  $\frac{10.0.0.1}{10.0.0.2}$   
→ then connect them with Hub
- ⑦ Click on the PCs → Configure → IP Address  $\frac{20.0.0.1}{20.0.0.2}$   
→ then connect them with Switch
- ⑧ Connect - Router & Hub with FastEthernet 0/0  
& Router & Switch with FastEthernet 1/0
- ⑨ Give IP address to router. It should be different  
For HUB give 10.0.0.4 & switch give 20.0.0.4
- ⑩ Go to CLI & give command for Enabling Router  
Router>enable  
Router>config terminal

Router(config)# interface fastethernet 0/0

Router(config-if)# ip address 10.0.0.4 255.0.0.0

Router(config-if)# no shutdown

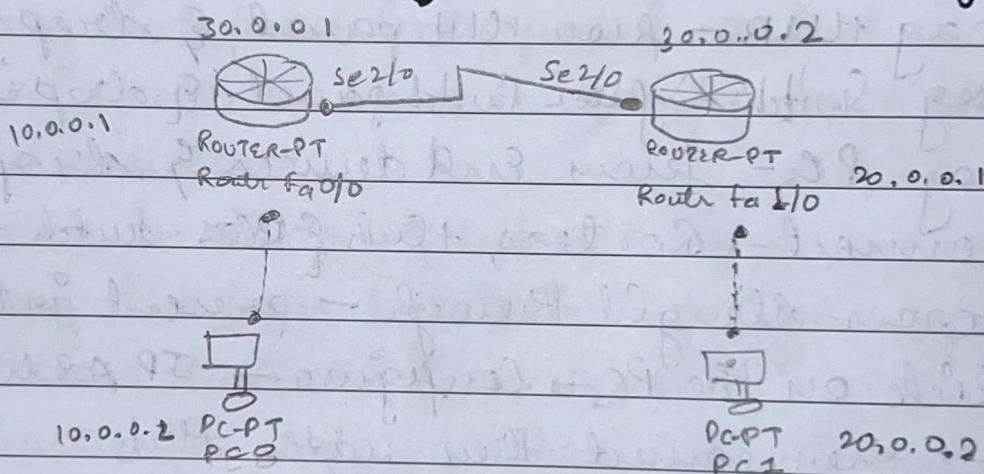
Router(config-if)# exit

For Switch

- ⑪ If the command is ✓, then monitor will enable & gives green signal.
- ⑫ Select PDU from side panel & select the source & dest. before that click if you have given the same gateway to every PCs which will show in config. of the PCs.  
→ then Ping Dest ask → Comp the capture / forward

### TASK 3: Connection b/w 2 routers :-

Config IP address to routers in Packet tracer, explore the follows. msg - Ping response, Destination- Unreachable, Request time out, reply



Steps: ① Select 2 Routers - from Router panel - name Router PT & drop it.

② Drag 2 PCs - from Cordless Device, convert it with 2 diff routers using Copper-cross over & Config.  
1<sup>st</sup> PC - as 10.0.0.2 & 2<sup>nd</sup> PC as 20.0.0.2  
+ gateway = R = 30.0.0.1 → (R<sub>1</sub>) = 30.0.0.2

③ Connect both routers with Serial DTE - connection with Se 2/0. # add serial Port in Physical to both Router, then add connection

④ We need to give 2 diff ip address as there are 2 connections

— One for fastethernet 0/0 - give 10.0.0.1 & enable it or give CLT command

— One for serial 2/0 - give 30.0.0.1 & enable it or give CLT command. Save for another Router

Router en,  
R > config t

R(c) # interface fastethernet 0/0  
R(c) # ip address 10.0.0.1 255.255.255.0

# no shutdown

# exit

R(c) # interface serial 0/1/0

# ip address 30.0.0.2 255.255.255.0

# no shutdown

R.

- ⑤ Go to CLI exit it & come to Router  
give command & Router# show ip route  
Gateway of last resort set if not set

give [ 10.0.0.0/8 is directly connected, fastethernet 0/0  
30.0.0.0/8 is directly connected, serial 2/0  
→ [ 20.0.0.0/8 ; " " , fastethernet 0/0  
" " , serial 2/0 ] ]

- ⑥ For sending msg, we need to connect all the IP-Address with each other.

For that, give command

Router# config t

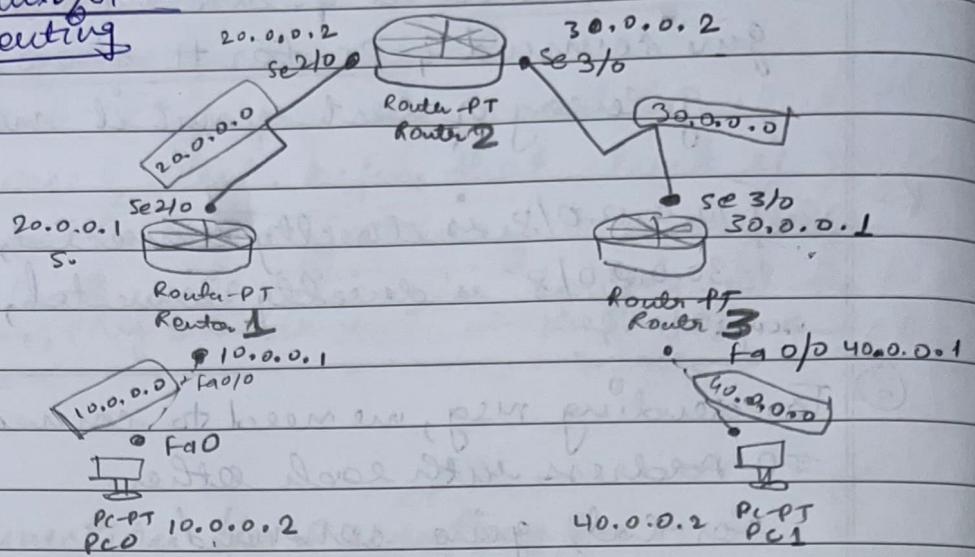
Router(config)# ip route 20.0.0.0 255.0.0.0 30.0.0.2  
→ do same thing for another Router

Router(config)# ip route 10.0.0.0 255.0.0.0 30.0.0.1

- ⑦ Now, send the PDU, it will show successful

**TASK 4:**

Configure default route, static route to the router  
default / static  
Routing



**Steps :**

- ① drag 3 Routers - from Router panel & connect it with Serial DTE connection
- ② drag the PC1 - from Endless Panels & connect it with copper-cross-over with the Router.
- ③ Config every Router from CLI
- from Router 1:

Router # config t

Router(Config)# ip address 10.0.0.1 255.0.0.0 20.0.0.2

Router(Config)# ip address 40.0.0.1 255.0.0.0 40.0.0.2

Router(Config)# exit

Router# show ip route

• from Router 2:

Router# config t

Router(Config)# ip route 0.0.0.0 255.0.0.0 20.0.0.1

Router(Config)# ip route 40.0.0.0 255.0.0.0 30.0.0.1

Router(Config)# exit

Router# show ip route

✓ • from Router 3:

Router# config t

Router(Config)# ip route 0.0.0.0 255.0.0.0 20.0.0.2

Router(Config)# ip route 10.0.0.0 255.0.0.0 30.0.0.2

Router(Config)# exit

Router# show ip route

for Router

+ give IP address to face/Ser. & on them.

- (4) Give the gateways for the PC 10.0.0.1 and 40.0.0.1
- (5) Now send the PDU, it will show successful

→ For Default Routing -

Give CLI Command -

From Router 3 to Router 2

Router # Config t

Router (Config) # ip route 0.0.0.0 0.0.0.0 30.0.0.2

Router (Config) # exit  
Router # show ip route { 30.0.0.1/8 drw, PEG/0  
40.0.0.1/8 drw, Ser}

From Router 1 to Router 2

Router # Config t

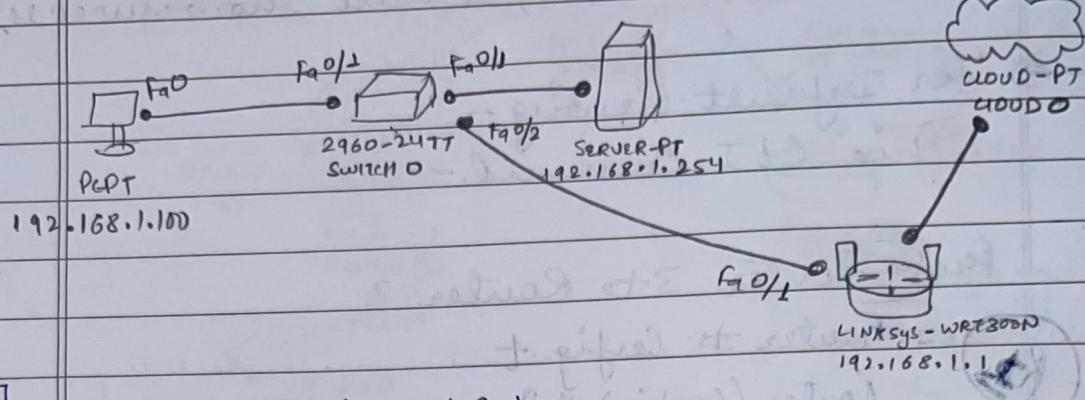
Router (Config) # ip route 0.0.0.0 0.0.0.0 20.0.0.2

Router (Config) # exit  
Router # show ip route { 40.0.0.1/8 drw, PEG/0  
20.0.0.1/8 drw, Ser}

(within a LAN)

## Demonstrate WEB server & DNS using Packet Tracer

### TASK 5



Steps: ① First, create a LAN

② Connect PC, switch 2960, generic Server with Copper straight-through cable

③ To enable the server to the switch & PC, label them with  $\rightarrow$  192.168.1.254 = Server, 192.168.1.100 for PC, label it also make sure that your port status is ON under static ip-address, enter 192.168.1.100

④ We will use ping - from PC - to - the server to make sure we have connection

⑤ for checking, go to the desktop  $\rightarrow$  cmd  $\rightarrow$   $\rightarrow$  ping .192.168.1.254

⑥ Now, click on the http to observe the web browser & look to the html, there type ip address 192.168.1.254  
Now you are connected to the server

⑦ Connect Router & Cloud Pt with each other & give ip address to Router 192.168.1.1

⑧ lets configure wireless router

⑨ Click on desktop tab  $\rightarrow$  cmd  $\rightarrow$  ping 192.168.1.1, now configure default gateway for the PC.

⑩ Go to IP Configuration, give it gateway as 192.168.1.1

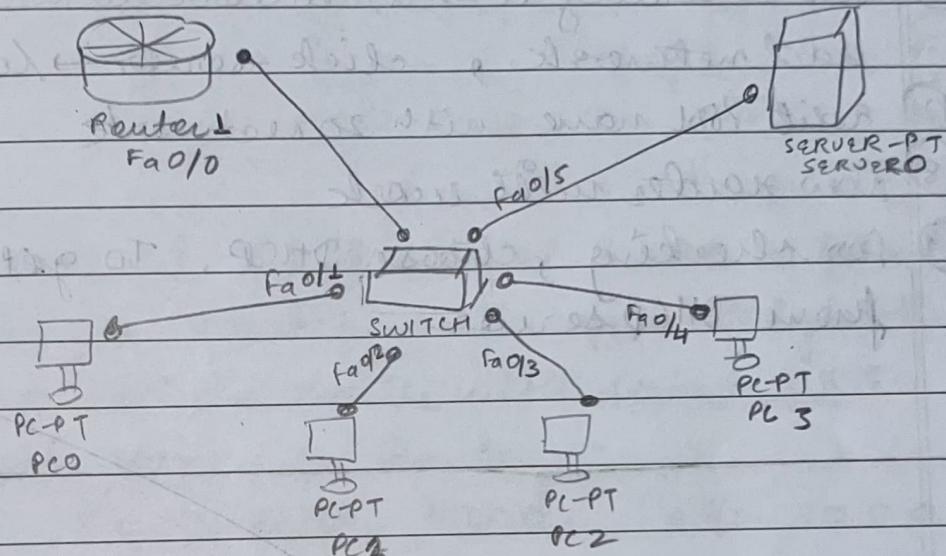
⑪ For connecting more server  $\rightarrow$ , we need to setup DNS

⑫ Click on DNS & make sure that it is turned ON

⑬ Under address, Type: 192.168.1.254 so superyahoo.com resolves to the ip address 192.168.1.254

- (14) Now, click on your PC & click IP configuration & under that server we see our server.
- (15) do now if we open up our web browser & type in superyahoo.com , we should see our new webpage displayed

## TASK 6]: Configure DHCP within a LAN in a packet Tracer



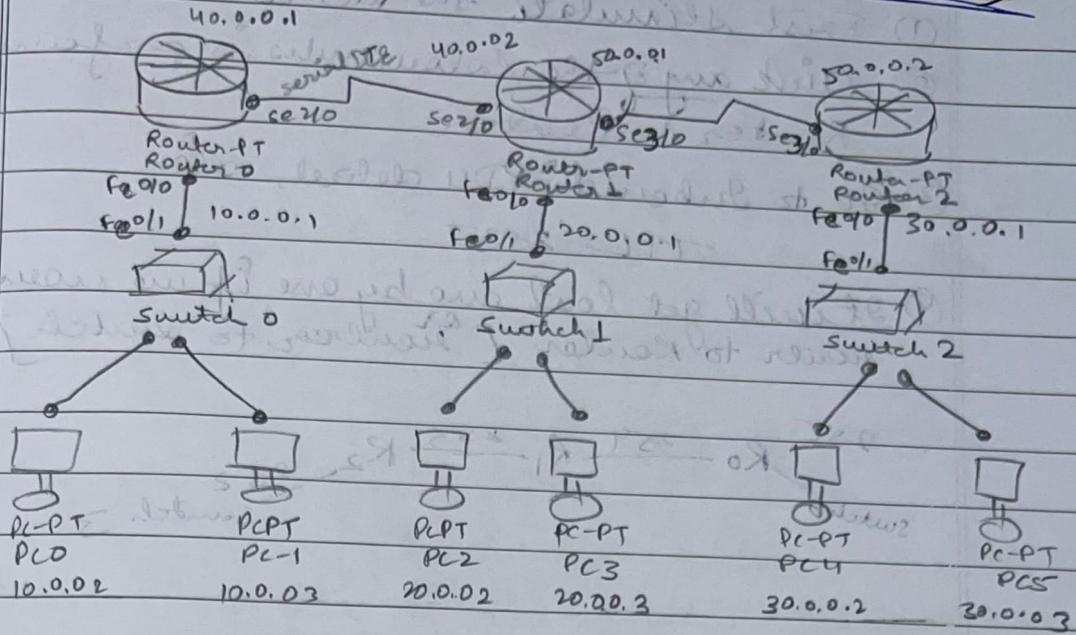
Steps ① Create a LAN

- ② Configure router interface with ip 10.0.0.1 & subnet mask 255.0.0.0
- ③ Click on server → config. the assign gateway 10.0.0.1
- ④ Click on fastEthernet & assign ip as 10.0.0.2 and subnet mask 255.0.0.0 for our server
- ⑤ Click on DHCP, give default gateway 10.0.0.1 and DNS server, 10.0.0.2
- ⑥ ~~edit the start ip address 10.0.0.10~~
- ⑦ ~~give maximum no of user, give 500~~
- ⑧ ~~Assign TFTP server ip address with 10.0.0.2~~
- ⑨ Click save
- ⑩ Click on any of the PC → Click Desktop → IP Config – choose DHCP wait for some time, if your DHCP request failed, try some more time.

- ⑪ Create a topology with another switch
- ⑫ Assign the center interface fast Ethernet 0/0 & fast Ethernet 0/1 with IP address
- ⑬ Give the gateway ip. address is 10.0.0.1
- ⑭ Click the fast Ethernet assign IP address 10.0.0.2 then DHCP server automatically assign 10 network
- ⑮ Click on PC in LAN → desktop → Ip config → choose DHCP you will get up ip from DHCP server from this PC
- ⑯ For observing PC in a network without server add network, click server → config → DHCP edit Pool name with 20 network
- ⑰ Now render will work
- ⑲ For checking, choose DHCP, To get ip address from DHCP server

## Task 7

# Routing Information Protocol (RIP)



- Steps:**
- ① Connect all PCs from end devices to Switches switch to center with Copper straight-through wire
  - ② connect all the Kernels with serial DTE
  - ③ give IP address for Router -

R<sub>0</sub> → Se2/0 40.0.0.1 Fe0/0 10.0.0.1

R<sub>1</sub> → Se2/0 40.0.0.2 Fe0/0 20.0.0.1 Se3/0 50.0.0.1

R<sub>2</sub> → Se3/0 50.0.0.2 Fe0/0 30.0.0.1

- ④ Configure IP address & give gateway to PCs :

PC0 → IP Address 10.0.0.2 Gateway 10.0.0.1

PC2 → 20.0.0.2 20.0.0.1

PC4 → 30.0.0.2 30.0.0.1

Click the first Router 0,

① Go to CLI

Type enable → Router>enable

Router# config terminal

# Router# ip route rip

(for R0) # network 10.0.0.0 # networks 40.0.0.0,

(for R1) # network 40.0.0.0 # networks 20.0.0.0,

# networks 50.0.0.0

1.1.1.1 (for R2) # network 50.0.0.0 # network 30.0.0.0

# exit

Router# show ip route

- ⑤ Go to PC → CMD → Ping it

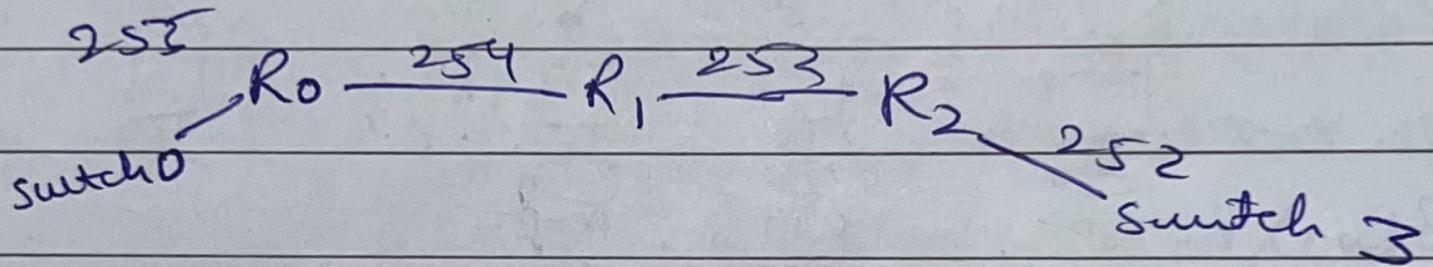


steps to observe TTL →

- ① First stimulate the packet
- ② Click any of the colour who is performing Router to Router
- ③ Go to Inbound PDU details see

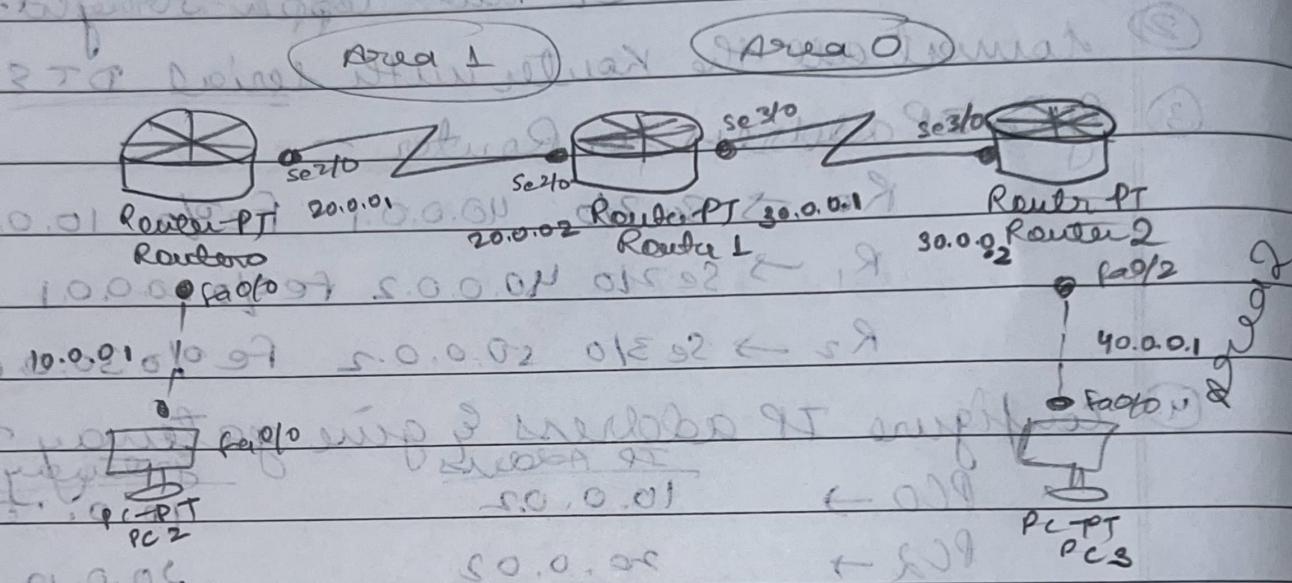
$$\text{TTL} = 255$$

- ④ It will get lost one by one if we move Router to Router [ Router to switch ]



## Task 8

Open Short Path First (OSPF) - Routing protocol & connect Area



- Steps:
- ① Connect all the routers with each other with serial DCE & PCs through Copper bypass lines.
  - ② Configure IP address to all interfaces.
  - ③ Then, whenever it is clock, give clock rate as 64000.
  - ④ Enable IP routing by configuring OSPF routing protocol in all routers → In Router R1,

R1(config)# router OSPF 1

R1(config-router)# router-id 1.1.1.1

R1(config-router)# network 10.0.0.0 0.255.255.255 area 3

20.0.0.0 0.255.255.255 area 1

R1(config-router)# exit

→ similarly for R2 & R3  
area 1-0 area 0-2

⑤ Now check Routing table of all Router —  
Router # show ip route.

- Here all the devices are linked together.
- There must be interface to keep OSPF process up.  
to configure loop back address to router

R1 (config-if) # interface loopback 0

# ip add 172.16.1.253 255.255.0.0

# no shutdown

Similar for R2 & R3

$\begin{cases} 253 - R_2 \text{ for} \\ 254 - R_3 \text{ for} \end{cases}$

⑥ Check Routing table for R3  
R3 # show ip route

→ Here R3 doesn't know the area 3, so we create virtual link.

In Router R1

R1 (config) # router ospf 1

R1 (config-router) # area 1 virtual link 2.2.2.2

In Router R2,

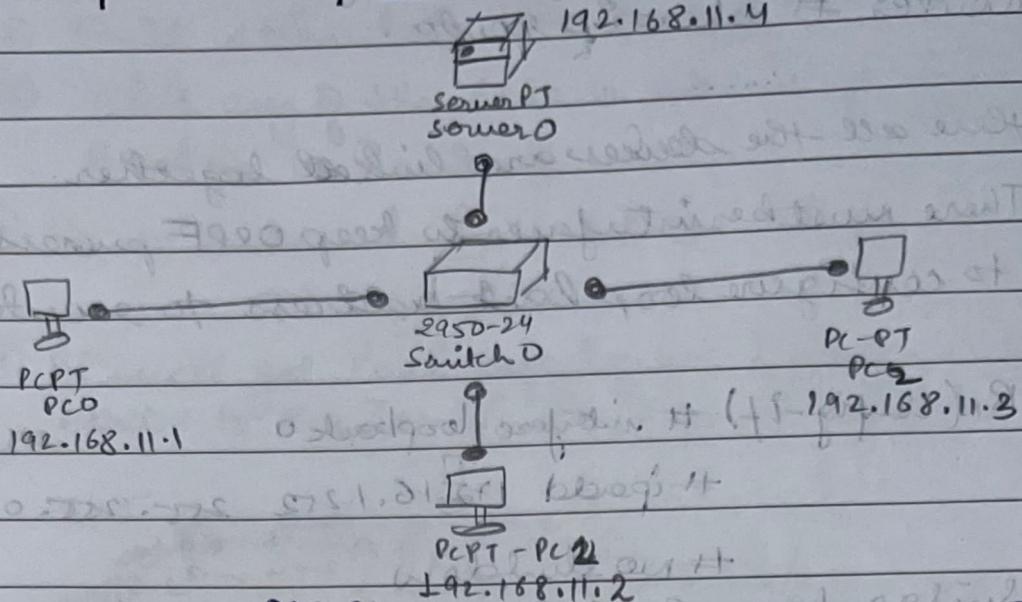
R2 (config Router) # area 1 virtual link 1.1.1.1

⑦ R2 & R3 get update about area 3, check it.

⑧ Check connectivity b/w host 10.0.0.10 to 40.0.0.10  
by giving ping command  
ping 40.0.0.10

## TASK 9

### Simple LAN & (ARP) Address Resolution Protocol



Steps: ① Assign IP address to all PCs & Server with 192.168.11.1/24

② Assign IP address to all PCs from end user device. Connect all with switch using copper straight through wire

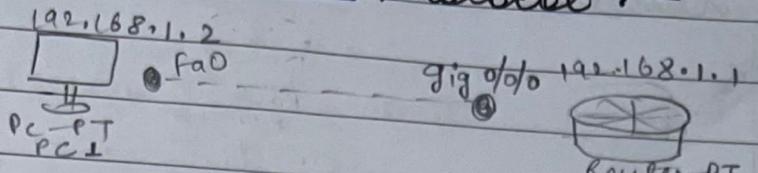
- ③ Assign IP address to all PCs & Server with 192.168.11.1/24
- ④ Go to Switch Panel, click on Inspect [Q] & right click on PC (ie search option)
- ⑤ After clicking on PC, → click on ARP Table & Notice that entries are there or not in ARP Table. Repeat same for Server
- ⑥ Click on PC → go to Command, → Type arp -a

{ Initially there are no ARP entries }

- ⑦ Try pinging from PC to Server → give command → ping 192.168.11.4
- ⑧ Send package from PC to Server; Notice 2 packets → ICMP = created
- ⑨ Click on ARP packet & then click on Capture button to start the stimulation
- ⑩ Ping request from PC to switch will happen
- ⑪ Click on capture to see the ICMP packet movement
- ⑫ After doing everything, click on inspect option & click to PC to check ARP Table

→ Now you can see that there is entries present.

Task 10: Constraint TELNET Protocol :-



Steps: ① drag PC & Router PT from the panel & connect both with copper cross wires.

② Configure IP Field + Gateway for PC.  
Go to desktop → IP →

③ Konfigurieren Router in CLI

Kontext erweite

Renter # config t

Rentier (Kaufg) + bestehende RL

R2 (Config) # enable secret ~~webexp~~

$R_2$  (contd) # int goto 0

R2 (config-if) # ip address 192.168.1.1 255.255.

$R_L$  (coupling)  $\neq$  no shutters.

R1 (config-if)# line vty 0 5

R+ (cyclic-life) # login

R<sub>1</sub> (candy - ~~top~~) # passim

R1 (config-line) # exit

R1 (config-line) # exit R1 # wr

Building castles

Building configuration [tower below] R+  
[OK]

• natural superficial

to PC → desktop → Command prompt .

ping 192.168.1.1

>>> telnet 192.168.1.1 user@pi (go-free) actual

Passwort: ~~top~~ wechselt zu 4t (zi-phus) nicht

Rides ~~not last~~ # (is still) return

~~passenger % exp~~ ~~Passenger % (B. p. f. f.) return~~

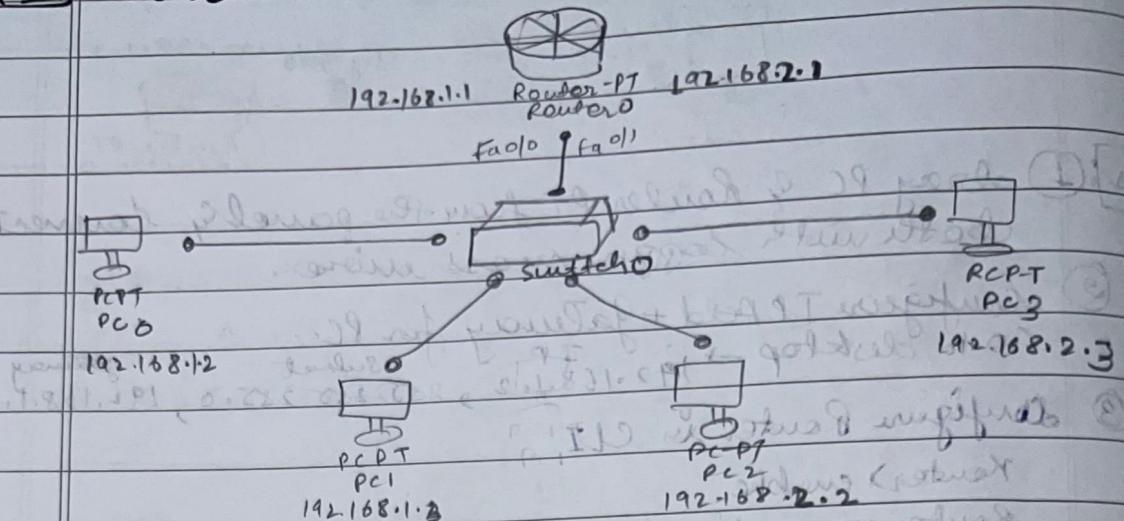
~~R1# 100 ft section if C is not taken~~

KI#

The password is set.

## Task 11

## Construct Virtual LAN :-



- (Steps):
- ① drag Router, PCs & switch & connect with Copper Straight wire
  - ② Configure all PCs & Router - IP, Subnet, Gateway
  - ③ Configure Switch:

2° Open switch → Configure → Fa0/5 (which is connected to routers)  
  → Replace access with Trunk (which is available in drop down list)

1. Go to VLAN Database → VLAN Number: 2

VLAN Name: VLAN → Add

3. Click on Fa0/3, Fa0/4 (connected to diff network)  
  → Under VLAN select 2

{ Basically STP spanning allows switches to forward frames from different VLANs over a single link called Trunk }

### ④ Configure Router.

Router > enable

Router# Config t

Router (config) # interface Fa0/0

Router (config-if) # ip address 192.168.1.1 255.255.255.0

Router (config-if) # no shutdown

~~Router (config-if) # exit fastethernet~~

~~Router (config) # interface Fa0/0.1~~

~~Router (config-if) # encapsulation dot1q 2~~

~~Router (config-subif) # ip address 192.168.2.1 255.255.255.0~~

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

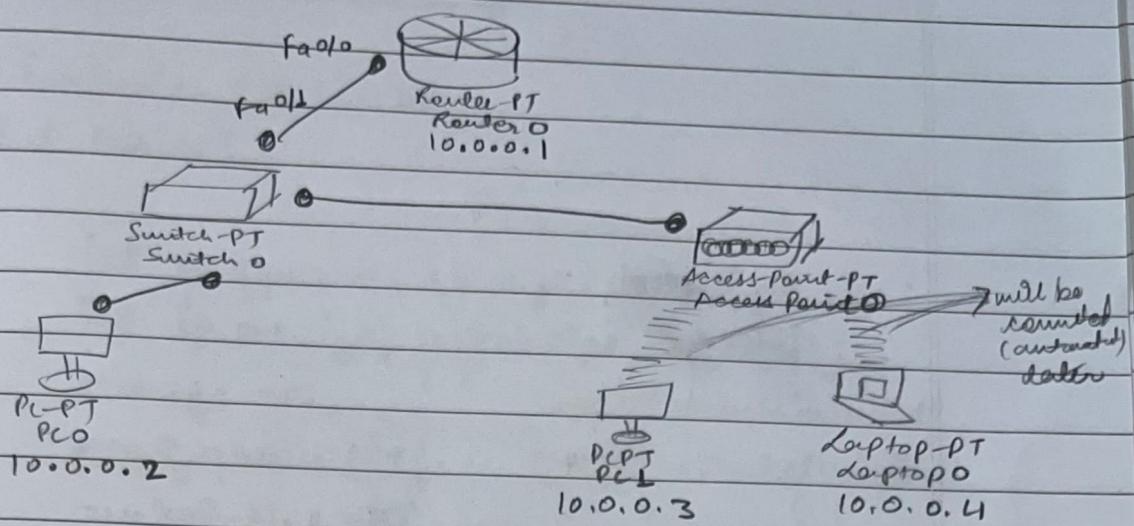
~~Router (config-subif) # exit~~

⑤ Check connectivity b/w networks

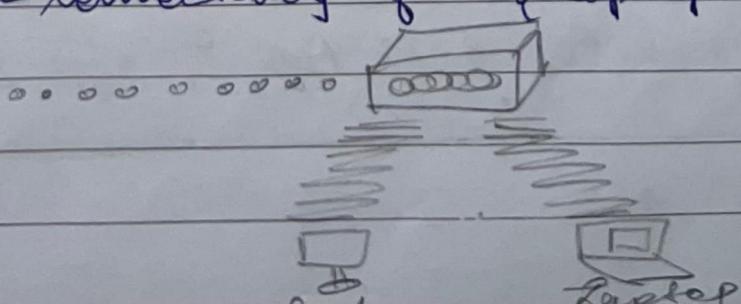
→ { Encapsulation dot1q 2 : It is the networking standard that supports VLANs on an IEEE 802.3 Ethernet network  
  → most widely used encapsulation method for VLAN tagging }

## TASK 12:

Construct WLAN & make nodes communicate wirelessly.



- Steps:**
- ① Drag switch, PC & Router & configure it after connecting with copper-straight through wire
  - ② Draw AccessPoint from the wireless device & drop it also, drag 1 PC by 1 Laptop.
  - ③ for Router - IP Add: 10.0.0.1, PC → IPAdd 10.0.0.2 Gateway 10.0.0.1
  - ④ Connect switch & Access point with copper-straight wire
  - ⑤ for AccessPoint configure give:  
SSID: WLAN & then change it to WEP & give  
WEP: 1234567890
  - ⑥ Configure PC1:  
Go to PC1 → Physical → Switchoff  
PC → remove PT-Host-NM-1... → ADD Drag WMP300N to the entry port.
  - ⑦ In config tab → Wireless  
SSID: WLAN  
WEP: 1234567890 For Laptop 10.0.0.4  
Static IPv4 Address: (10.0.0.3)  
Subnet mask: 255.0.0.0
  - ⑧ Configure Laptop: Same as PC1  
Just, IPv4 address: 10.0.0.4
  - ⑨ Check connectivity of PC & Laptop i.e.



## \* Programs:

Task 1:

### Leaky Bucket Algo:

```
#include < stdio.h >
int main()
{
    int no_of_queries, storage, ip_pkt_size;
    int ip_pkt_size, bucket_size, size_left;
    storage = 0; // initial packets in bucket
    no_of_queries = 4; // no of time bucket checked
    bucket_size = 10;
    ip_pkt_size = 4; // no of packets that enter in bkt at a time
    op_pkt_size = 1; // if ip_pkt_size <= size_left
    for(int i=0; i<no_of_queries; i++)
    {
        size_left = (bucket_size - storage);
        if(ip_pkt_size <= size_left)
        {
            storage += ip_pkt_size; // update storage
            cout << "Packet loss = %d \n", ip_pkt_size;
        }
        else
            cout << "Packet loss = %d \n", ip_pkt_size;
        storage -= op_pkt_size;
    }
    return 0;
}
```

Off  
Buffer size = 4 out of bucket size = 10  
" " = 7 " " = 10  
" " = 10  
Packet loss = 4 + 0 + 1 = 5  
Buffer size = 9 out of bucket size = 10  
" " = 7 " " = 10

## TASK 2:

Error detection using CRC-CCITT (16 Bits)

#include &lt;stdio.h&gt;

#include &lt;string.h&gt;

```

int crc(char *ip, char *op, char *poly, int mode)
{
    strcpy(op, ip); // copy ip to op
    if (mode)
        { // append 0 to op
            for (int i=1; i<strlen(poly); i++)
            {
                strcat(op, "0");
            }
        }
    // perform XOR
    for (int i=0; i<strlen(ip); i++)
    {
        if (op[i] == '1')
            { // for (int j=0; j<strlen(poly); j++)
                for (int j=0; j<strlen(poly); j++)
                    { // if (op[i+j] == poly[j])
                        if (op[i+j] == poly[j])
                            { // op[i+j] = '0'
                                op[i+j] = '0';
                            }
                        else
                            { // op[i+j] = '1'
                                op[i+j] = '1';
                            }
                    }
            }
    }
}

```

i (eg 10101010) // check for error

```

for (int i=0; i<strlen(op); i++)
{
    if (op[i] == '1')
        { // returns 0
            return 0;
        }
    return 1; // however
}

```

~~int main()~~ // tested for two offset - 910

```

{
    char ip[50], op[50], recv[50];
    char poly[] = "1000100000100001";
    printf("Enter IP msg in binary:");
    scanf("%s", ip);
}

```

```

    // calculate CRC & get transmission msg
    crc(ip, op, poly, 1);
    printf("The transmitted msg = %s\n", op);
    printf("Enter received msg in binary : ");
    scanf(" %s", recv);

    // check for error in received msg
    if (crc(recv, op, poly, 0))
    {
        printf("No error in data\n");
        // 10001110 - without error
    }
    else
    {
        printf("Error in data transmission\n");
        // 00001110 - transmission error
    }
    return 0;
}

```

C/P Enter ip msg in binary: 1111  
 The transmission msg = 00000111000111101110  
 Enter the received msg in binary: 1111  
 Error in data transmission has occurred.

② Enter ip msg in binary = 1111  
 The transmission msg = 100000111000111101110  
 Enter the received msg in binary: 10111102  
 No error in data. 00001110 - transmission error

### TASK 3

## Socket Programming :-

```
i = input("What is your name? ")  
if i == "":  
    print("User did not input anything")  
else:  
    print("Hello " + i)
```

⇒ client (TCP.py) →

from socket import \* →

serverName = "127.0.0.1" →

serverPort = 12000 →

clientSocket = socket(AF\_INET, SOCK\_STREAM) →

clientSocket.connect((serverName, serverPort)) →

sentence = input("\nEnter file name: ") →

clientSocket.send(sentence.encode()) →

fileContents = clientSocket.recv(1024).decode() →

print("From Server: \n") →

print(fileContents) →

clientSocket.close() →

⇒ Server TCP.py →

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") →

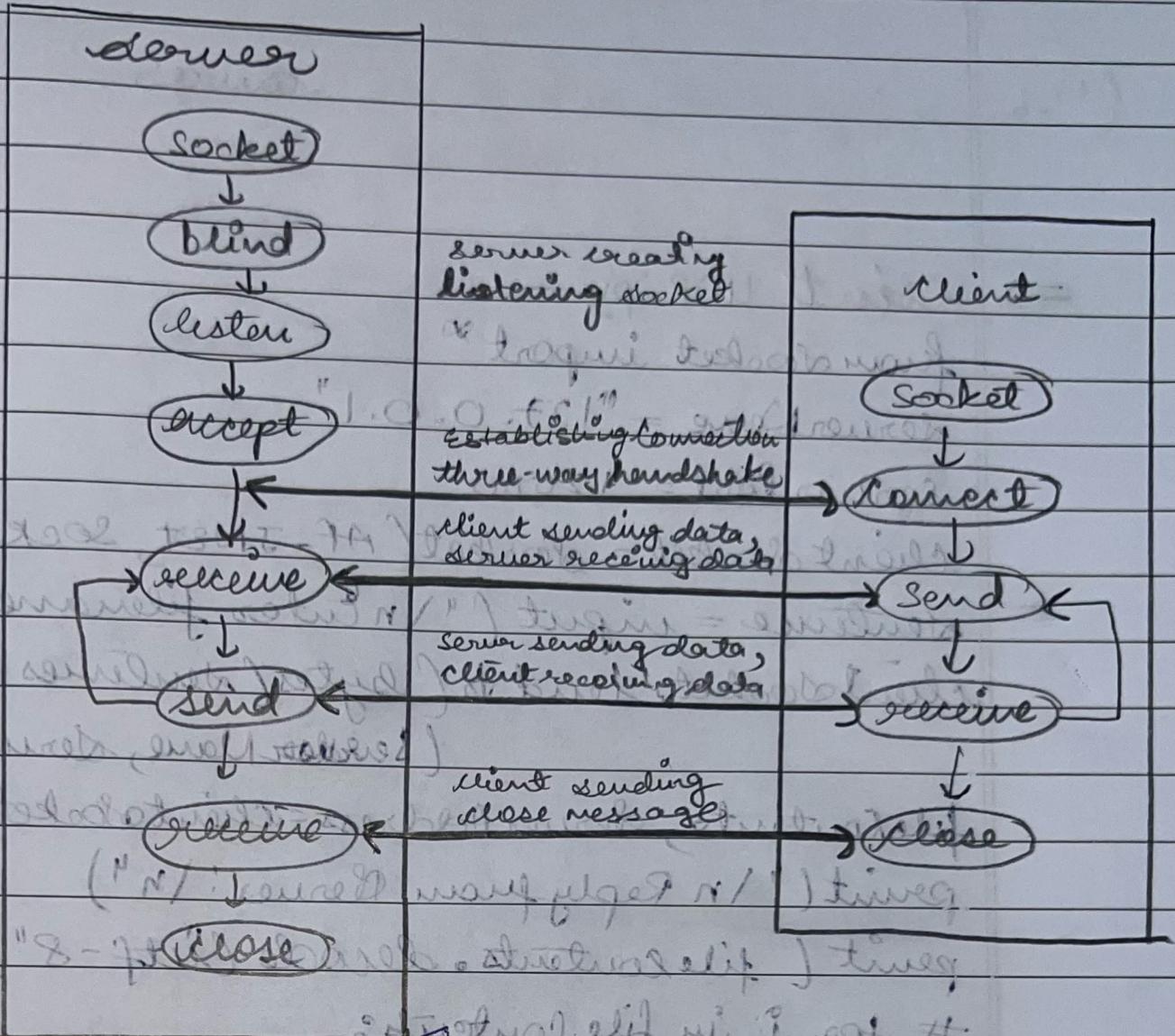
I = file.read(1024) →

connectionSocket.send(I.encode()) →

print("\nSent contents of " + sentence) →

file.close() →

connectionSocket.close() →



⇒ Client UDP.py -

```
from socket import *
serverName = "127.0.0.1"
serverPort = 12000
clientSocket = socket(AF_INET, SOCK_DGRAM)
sentence = input("\n Enter filename")
clientSocket.sendto(sentence.encode("utf-8"), (serverName, serverPort))
fileContents, serverAddress = clientSocket.recvfrom(2048)
print("\n Reply from Server:\n")
print(fileContents.decode("utf-8"))
# for i in fileContents:
#     print(str(i), end="")
clientSocket.close()
clientSocket.close()
```

⇒ Server UDP.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")
    con = file.read(2048)
    serverSocket.sendto(con.encode("utf-8"), clientAddress)
```

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

23 22