

21BDS0340

Abhinav Dinesh Srivatsa

Computer Networks Lab

Lab Experiment – I

Question 1

1. Ethernet Hub

This piece of hardware emits/broadcasts data to every computer/node connected to the hub. These computers/nodes are connected with the RJ45 cables.



This hub can connect to 4 computers (labelled as 1, 2, 3, 4) with RJ45 cables.

2. RJ45 Jack

This piece of hardware is an 8-pin cable that is used to connect computers/nodes on an ethernet based local area network, also known as LAN. These wires are made of copper.



3. LAN Switch

This piece of hardware links locations on a LAN. After a successful connection to a computer by its IP address, the switch will only broadcast data to the computer, instead of to all Lan connected computers.



This switch can connect to 24 computers (labelled as 1, 2, 3, ..., 24) with RJ45 cables.

4. Router

This piece of hardware links multiple local area networks (LAN), the source and destination may not share the same IP sub address.



This router can connect to 3 LANs and 1 WAN (wide-area network) with RJ45 cables.

5. Bridge

This piece of hardware is specifically used to connect 2 LANs together, or to connect a LAN to a larger network.



This bridge can connect between 4 different LAN networks.

6. Wireless Access Point

This piece of hardware can convert a physical network to a wireless one, which can connect to wireless devices.



This WAP can connect to multiple wireless devices by giving an input RJ45 cable to connect it to a particular LAN.

7. Modem

This piece of hardware connects you to the internet, essentially connecting you to other servers hosted elsewhere.



8. Fiber Optic Cable

This piece of hardware is used for data transmission through the use of light. These cables are the newest addition to networking technology with massively improved speeds compared to copper wires.



9. Repeater

This piece of hardware is used to boost the amplitude/power of a signal, allowing it to travel a further distance and to combat attenuation.



10. Gateway

This piece of hardware sends local traffic to other networks/LANs.



Question 2

Byte Stuffing

Code: Python

```
print('Enter bits, start with frame with f, end frame with e:')
raw = input()
allowed = ['0', '1', 'e', 'f']

# checking for correct format (f has to be starting)
if raw[0] != 'f':
    print('Invalid format!')
    exit()

# checking for correct formatting (e after f)
e_flag = False
for i in raw:
```

```

if i not in allowed:
    print('Invalid character!')
    break

if e_flag and i == 'f':
    e_flag = False
if e_flag and i != 'f':
    print('Invalid format!')
    exit()
if i == 'e':
    e_flag = True

# creating output frame (e before all f and e)
output = ''
for i in raw:
    if i == 'e' or i == 'f':
        output += 'e' + i
    else:
        output += i
print('The output frame is:')
print(output)

```

Input: f010110101ef10101001e

Output: ef010110101eeef10101001ee

Sample Image:



```

Enter bits, start with frame with f, end frame with e:
f010110101ef10101001e
The output frame is:
ef010110101eeef10101001ee

```

Bit Stuffing

Code: Python

```

# default flag = 01111110
print('Enter bits:')
raw = input()
allowed = ['0', '1']

# checking for correct formatting (only 0's and 1's)
for i in raw:
    if i not in allowed:
        print('Invalid format!')
        exit()

# creating output frame (stuff 0 after 5 consecutive 1's)
output = ''

```

```

count = 0
for i in row:
    output += i
    if i == '1':
        count += 1
    else:
        count = 0
    if count == 5:
        output += '0'
        count = 0
print('The output frame is:')
print(output)

# output with the f = 01111110
print('Th output frame with f is:')
print(f'01111110{output}01111110')

```

Input: 110101111101011111101011111110

Output: 110101111100101111101010111110110

Sample Image:

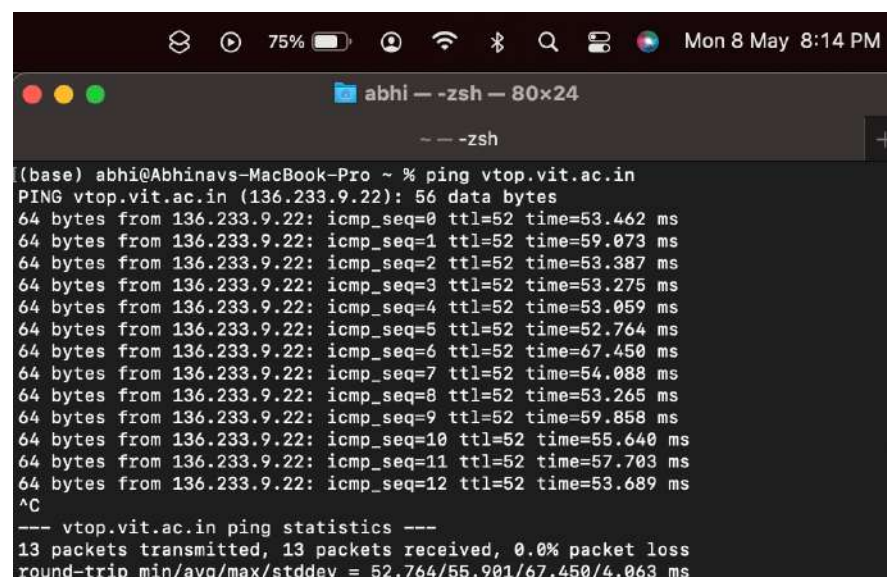
```

Enter bits:
110101111101011111101011111110
The output frame is:
110101111100101111101010111110110
Th output frame with f is:
0111111011010111110010111110101011111011001111110

```

Question 3

- ping

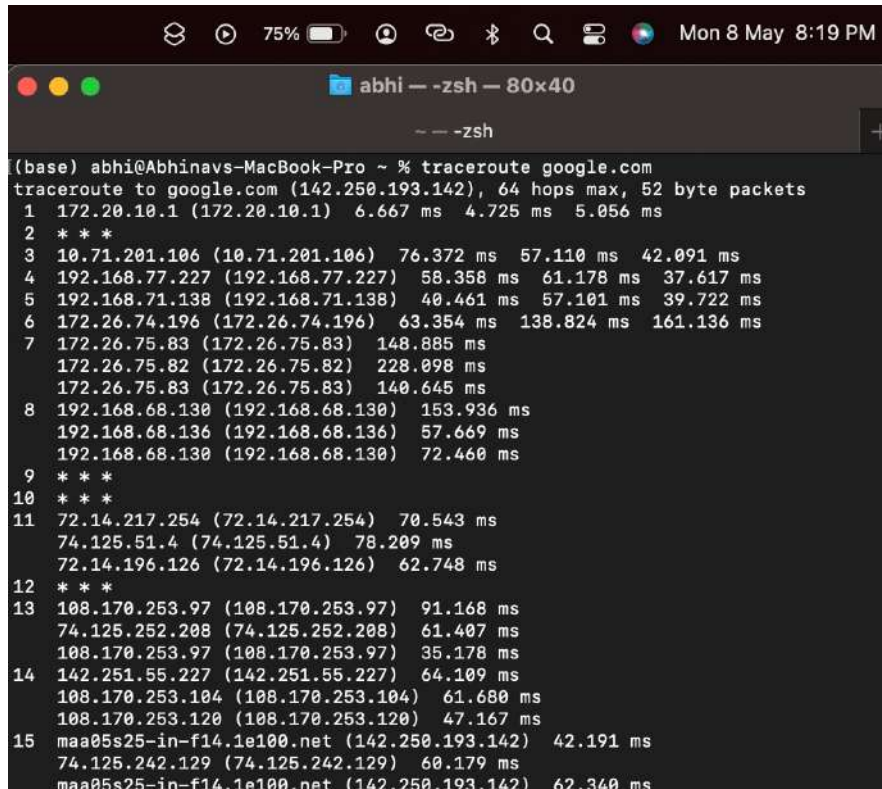


```

Mon 8 May 8:14 PM
abhi -- -zsh -- 80x24
~ -- -zsh
(base) abhi@Abhinavs-MacBook-Pro ~ % ping vtop.vit.ac.in
PING vtop.vit.ac.in (136.233.9.22): 56 data bytes
64 bytes from 136.233.9.22: icmp_seq=0 ttl=52 time=53.462 ms
64 bytes from 136.233.9.22: icmp_seq=1 ttl=52 time=59.073 ms
64 bytes from 136.233.9.22: icmp_seq=2 ttl=52 time=53.387 ms
64 bytes from 136.233.9.22: icmp_seq=3 ttl=52 time=53.275 ms
64 bytes from 136.233.9.22: icmp_seq=4 ttl=52 time=53.059 ms
64 bytes from 136.233.9.22: icmp_seq=5 ttl=52 time=52.764 ms
64 bytes from 136.233.9.22: icmp_seq=6 ttl=52 time=67.450 ms
64 bytes from 136.233.9.22: icmp_seq=7 ttl=52 time=54.088 ms
64 bytes from 136.233.9.22: icmp_seq=8 ttl=52 time=53.265 ms
64 bytes from 136.233.9.22: icmp_seq=9 ttl=52 time=59.858 ms
64 bytes from 136.233.9.22: icmp_seq=10 ttl=52 time=55.640 ms
64 bytes from 136.233.9.22: icmp_seq=11 ttl=52 time=57.703 ms
64 bytes from 136.233.9.22: icmp_seq=12 ttl=52 time=53.689 ms
^C
--- vtop.vit.ac.in ping statistics ---
13 packets transmitted, 13 packets received, 0.0% packet loss
round-trip min/avg/max/stddev = 52.764/55.901/67.450/4.063 ms

```

- traceroute

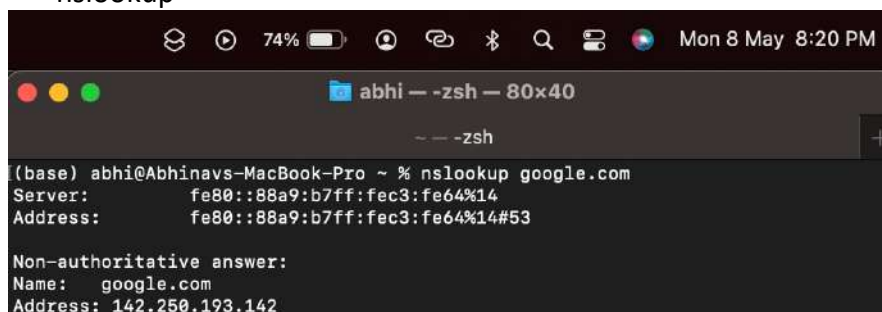


```

(base) abhi@Abhinavs-MacBook-Pro ~ % traceroute google.com
traceroute to google.com (142.250.193.142), 64 hops max, 52 byte packets
 1  172.20.10.1 (172.20.10.1)  6.667 ms  4.725 ms  5.056 ms
 2  * * *
 3  10.71.201.106 (10.71.201.106)  76.372 ms  57.110 ms  42.091 ms
 4  192.168.77.227 (192.168.77.227)  58.358 ms  61.178 ms  37.617 ms
 5  192.168.71.138 (192.168.71.138)  40.461 ms  57.101 ms  39.722 ms
 6  172.26.74.196 (172.26.74.196)  63.354 ms  138.824 ms  161.136 ms
 7  172.26.75.83 (172.26.75.83)  148.885 ms
   172.26.75.82 (172.26.75.82)  228.098 ms
   172.26.75.83 (172.26.75.83)  140.645 ms
 8  192.168.68.130 (192.168.68.130)  153.936 ms
   192.168.68.136 (192.168.68.136)  57.669 ms
   192.168.68.130 (192.168.68.130)  72.460 ms
 9  * * *
10  * * *
11  72.14.217.254 (72.14.217.254)  70.543 ms
   74.125.51.4 (74.125.51.4)  78.209 ms
   72.14.196.126 (72.14.196.126)  62.748 ms
12  * * *
13  108.170.253.97 (108.170.253.97)  91.168 ms
   74.125.252.208 (74.125.252.208)  61.407 ms
   108.170.253.97 (108.170.253.97)  35.178 ms
14  142.251.55.227 (142.251.55.227)  64.109 ms
   108.170.253.104 (108.170.253.104)  61.680 ms
   108.170.253.120 (108.170.253.120)  47.167 ms
15  maa05s25-in-f14.1e100.net (142.250.193.142)  42.191 ms
   74.125.242.129 (74.125.242.129)  60.179 ms
   maa05s25-in-f14.1e100.net (142.250.193.142)  62.340 ms

```

- nslookup



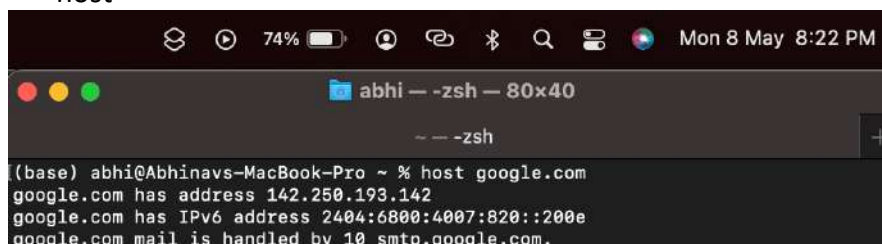
```

(base) abhi@Abhinavs-MacBook-Pro ~ % nslookup google.com
Server:         fe80::88a9:b7ff:fec3:fe64%14
Address:        fe80::88a9:b7ff:fec3:fe64%14#53

Non-authoritative answer:
Name:   google.com
Address: 142.250.193.142

```

- host



```

(base) abhi@Abhinavs-MacBook-Pro ~ % host google.com
google.com has address 142.250.193.142
google.com has IPv6 address 2404:6800:4007:820::200e
google.com mail is handled by 10 smtp.google.com.

```


- arp

```

Mon 8 May 8:23 PM

abhi — -zsh — 80x40
-- -zsh

(base) abhi@Abhinavs-MacBook-Pro ~ % arp -a
? (172.20.10.1) at 8a:a9:b7:c3:fe:64 on en0 ifscope [ethernet]
? (172.20.10.5) at c8:89:f3:b5:b6:72 on en0 ifscope permanent [ethernet]
? (172.20.10.15) at ff:ff:ff:ff:ff:ff on en0 ifscope [ethernet]
? (224.0.0.251) at 1:0:5e:0:0:fb on en0 ifscope permanent [ethernet]
? (239.255.255.250) at 1:0:5e:7f:ff:fa on en0 ifscope permanent [ethernet]

```

- netstat

```

Mon 8 May 8:24 PM

abhi — -zsh — 80x56
-- -zsh

(base) abhi@Abhinavs-MacBook-Pro ~ % netstat
Active Internet connections
Proto Recv-Q Send-Q Local Address           Foreign Address         (state)
tcp4      0      0 172.20.10.5.55136       13.67.9.5.https        ESTABLISHED
tcp4      0      0 172.20.10.5.55135       13.67.9.5.https        ESTABLISHED
tcp6      0      0 2409:4072:411:fe.55134  maa05s23-in-x0a..https ESTABLISHED
tcp6      0      0 2409:4072:411:fe.55133  maa03s35-in-x01..https ESTABLISHED
tcp6      0      0 2409:4072:411:fe.55126  maa05s21-in-x03..https ESTABLISHED
tcp6      0      0 2409:4072:411:fe.55125  maa05s28-in-x03..https ESTABLISHED
tcp6      0      0 2409:4072:411:fe.55124  maa03s31-in-x0e..https ESTABLISHED
tcp6      0      0 2409:4072:411:fe.55123  maa05s24-in-x0e..https ESTABLISHED
tcp4      0      0 172.20.10.5.55121       www.archive.org.https  ESTABLISHED
tcp6      0      0 2409:4072:411:fe.55120  maa05s21-in-x0e..https ESTABLISHED
tcp6      0      0 2409:4072:411:fe.55099  maa03s44-in-x0d..https ESTABLISHED
tcp6      0      0 2409:4072:411:fe.55098  maa05s24-in-x0e..https ESTABLISHED
tcp6      0      0 2409:4072:411:fe.55097  maa03s43-in-x04..https ESTABLISHED
tcp6      0      0 2409:4072:411:fe.55096  maa03s43-in-x04..https ESTABLISHED
tcp4      0      0 172.20.10.5.55081       ec2-13-127-122-2.https ESTABLISHED
tcp6      0      0 2409:4072:411:fe.55080  maa05s28-in-x16..https ESTABLISHED
tcp4      0      0 172.20.10.5.55066       17.57.145.119.6223    ESTABLISHED
tcp6      0      0 2409:4072:411:fe.55065  maa05s14-in-x0e..https ESTABLISHED
tcp6      0      0 abhinavs-macbook.54710  abhinavs-iphone.50223 ESTABLISHED
tcp4      0      0 172.20.10.5.55058       whatsapp-cdn-shv.https ESTABLISHED
tcp6      0      0 abhinavs-macbook.black  fe80::8b38:d496:1027  ESTABLISHED
tcp6      0      0 abhinavs-macbook.1024   fe80::8b38:d496:1024  ESTABLISHED
tcp6      0      0 2409:4072:411:fe.55112  2405:200:1630:ff.https TIME_WAIT
tcp6      0      0 2409:4072:411:fe.55109  2405:200:1630:b1.https TIME_WAIT
tcp4      0      0 172.20.10.5.55137       52.109.44.80.https    TIME_WAIT
tcp4      0      0 172.20.10.5.55138       52.109.44.80.https    TIME_WAIT
tcp6      0      0 2409:4072:411:fe.55141  2403:300:a41:606.https TIME_WAIT
tcp6      0      0 2409:4072:411:fe.55140  2403:300:a41:606.https TIME_WAIT
tcp4      0      0 172.20.10.5.55093       maa05s21-in-f14..https TIME_WAIT

```

- getmac

```

Mon 8 May 8:33 PM

abhi — -zsh — 80x56
-- -zsh

(base) abhi@Abhinavs-MacBook-Pro ~ % ifconfig en1 | awk '/ether/{print $2}'
36:e9:35:f5:74:80

```