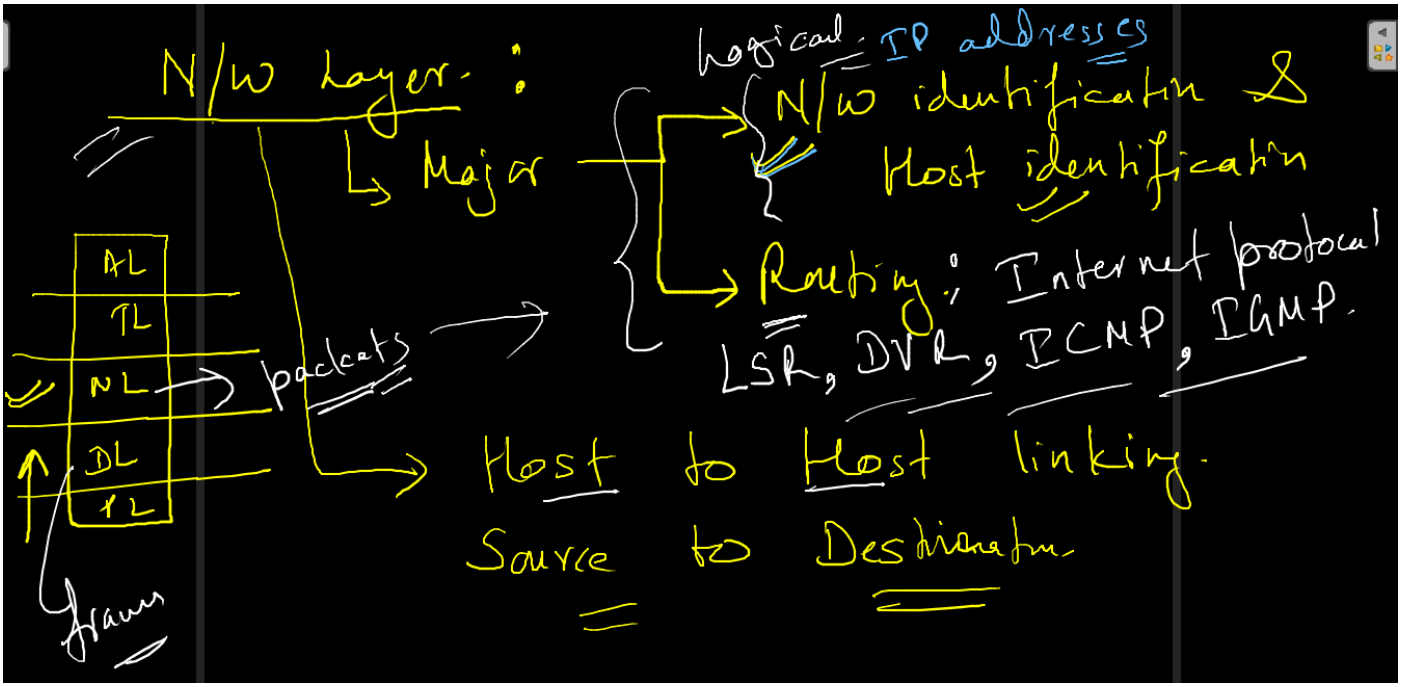
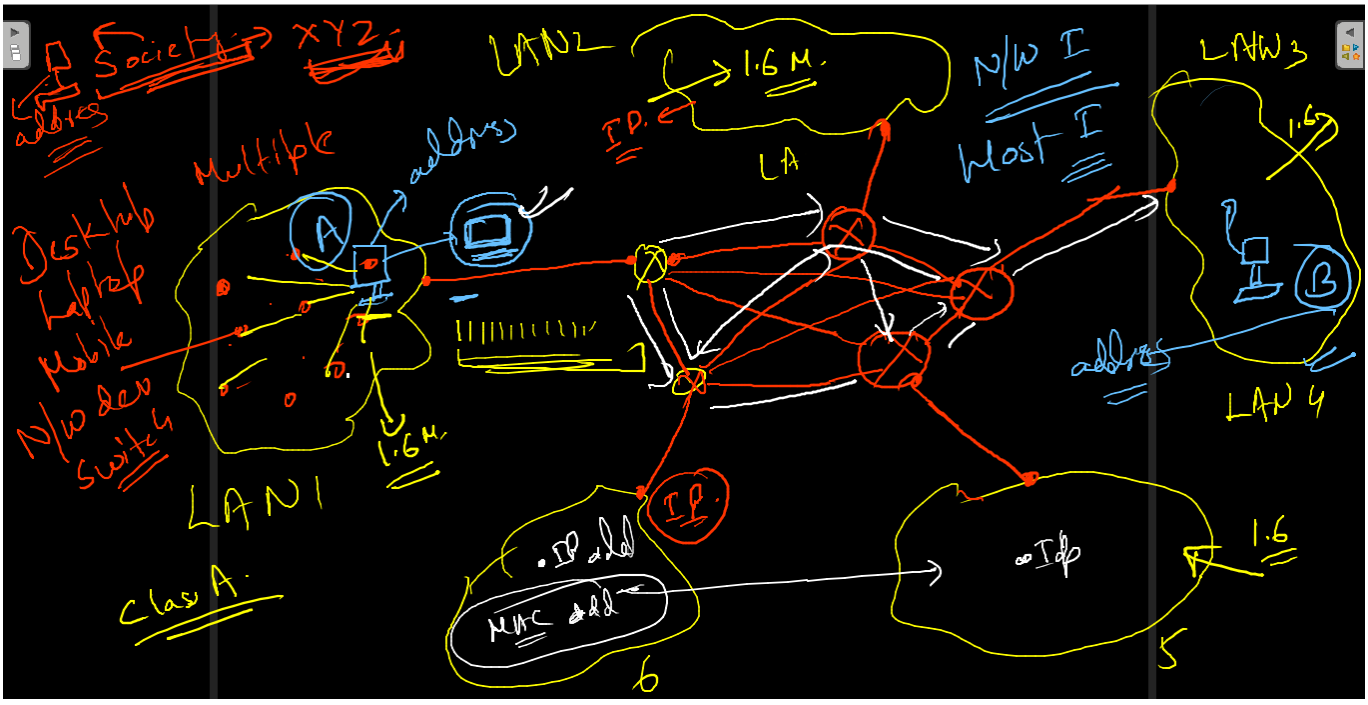
UNIT – III


N/w layer:
Major
logical IP addresses
N/w identification &
Host identification
Routing: Internet protocol
LSR, DVR, ICMP, IGMP.
AL
TL
NL
DL
YL
packets
frames
Host to Host linking.
Source to Destination.


Society
XYZ
address
Multiple
address
LAN2
1.6 M.
IP.
N/W I
Host I
LA
LAN3
1.6
Desktop
Laptop
Mobile
N/w dev
Switch
A
B
address
LAN 4
1.6 M.
LAN1
Class A.
IP.
IP add
MAC add
Idp
1.6
6
5

Logical Addressing — MAC

↳ actual physical.

↳ Logical addresses that are assigned to different nodes/hosts in the n/w.

Laptop → A ⇒ MAC permanent

IP address can change

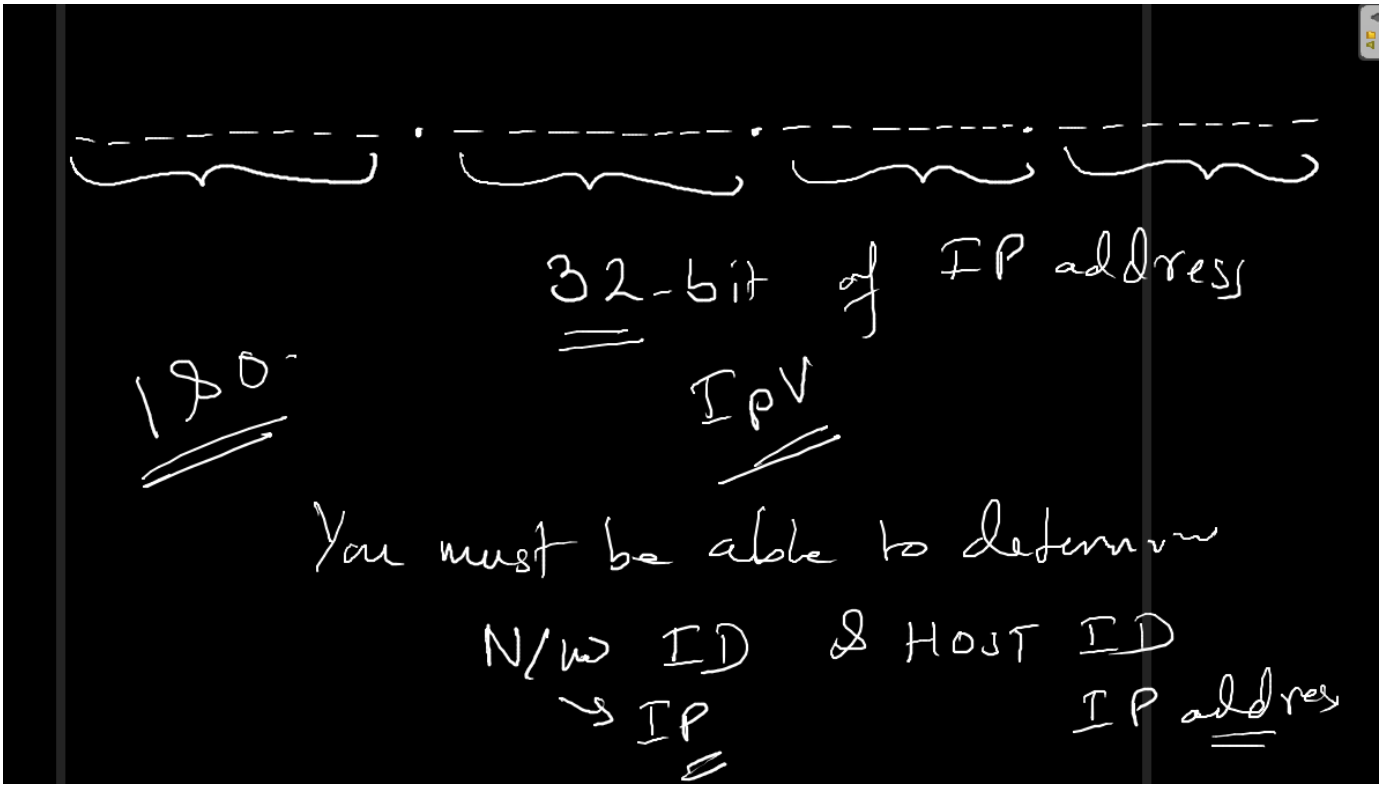
Classful IP addressing Scheme

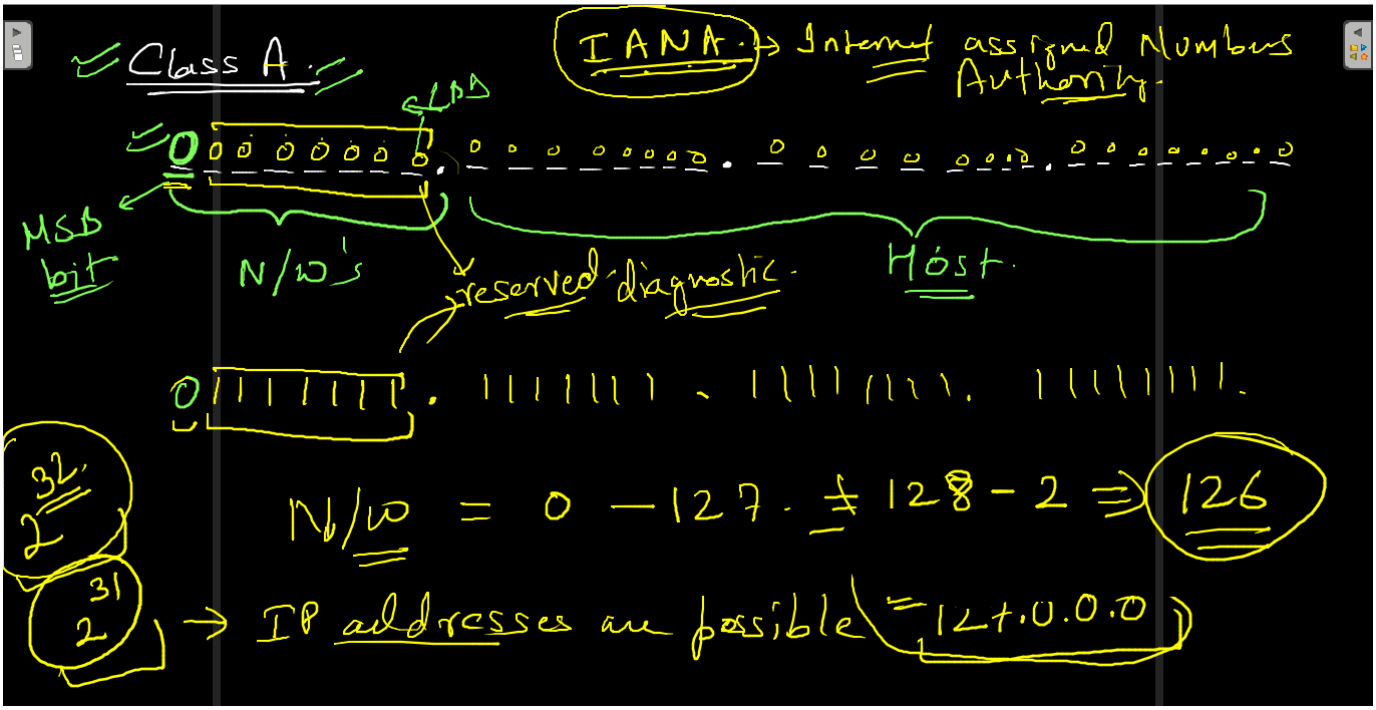
Classless "

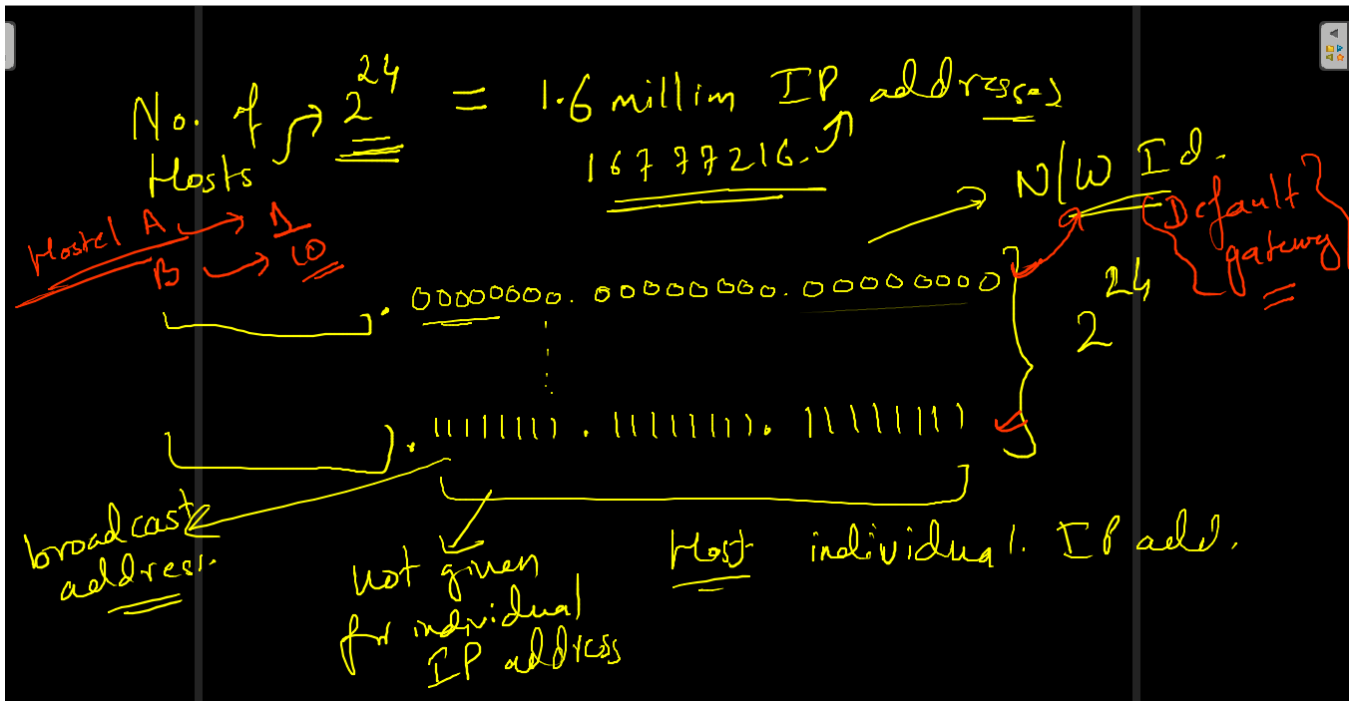
IPV4.

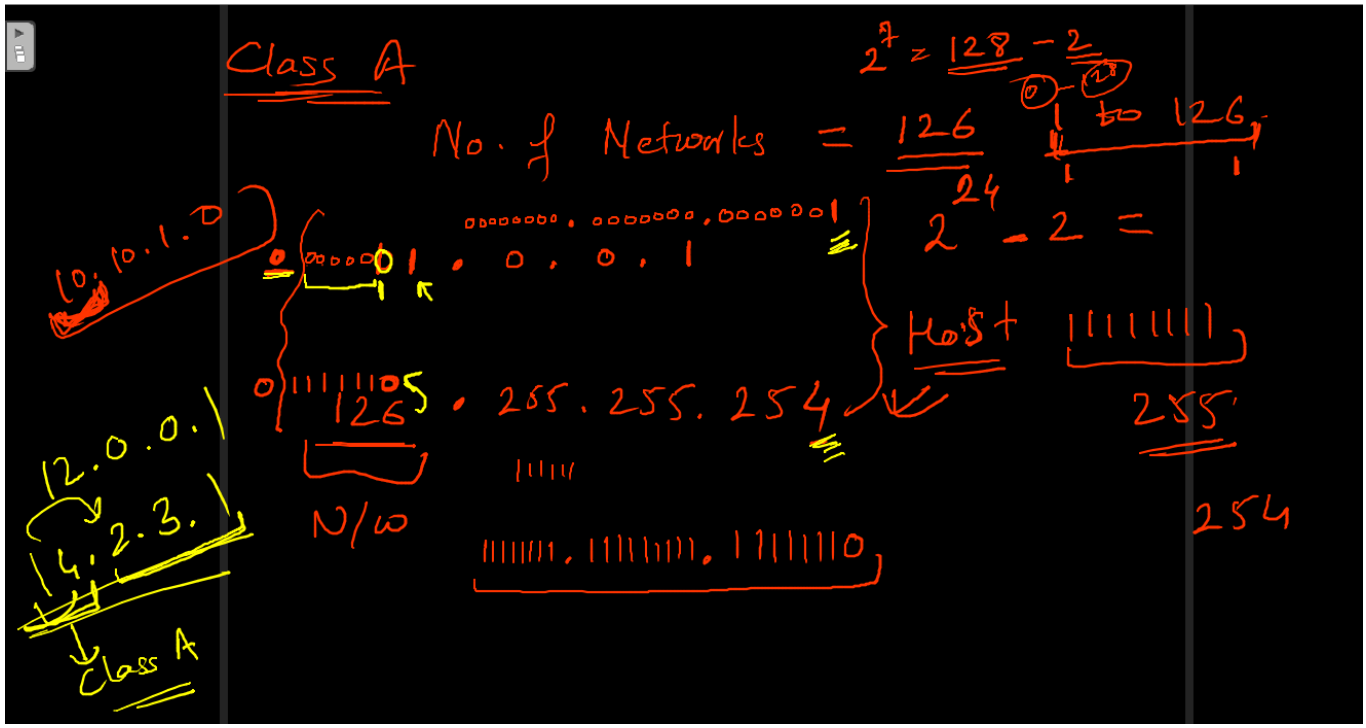
Class A
Class B
Class C
Class D
Class E

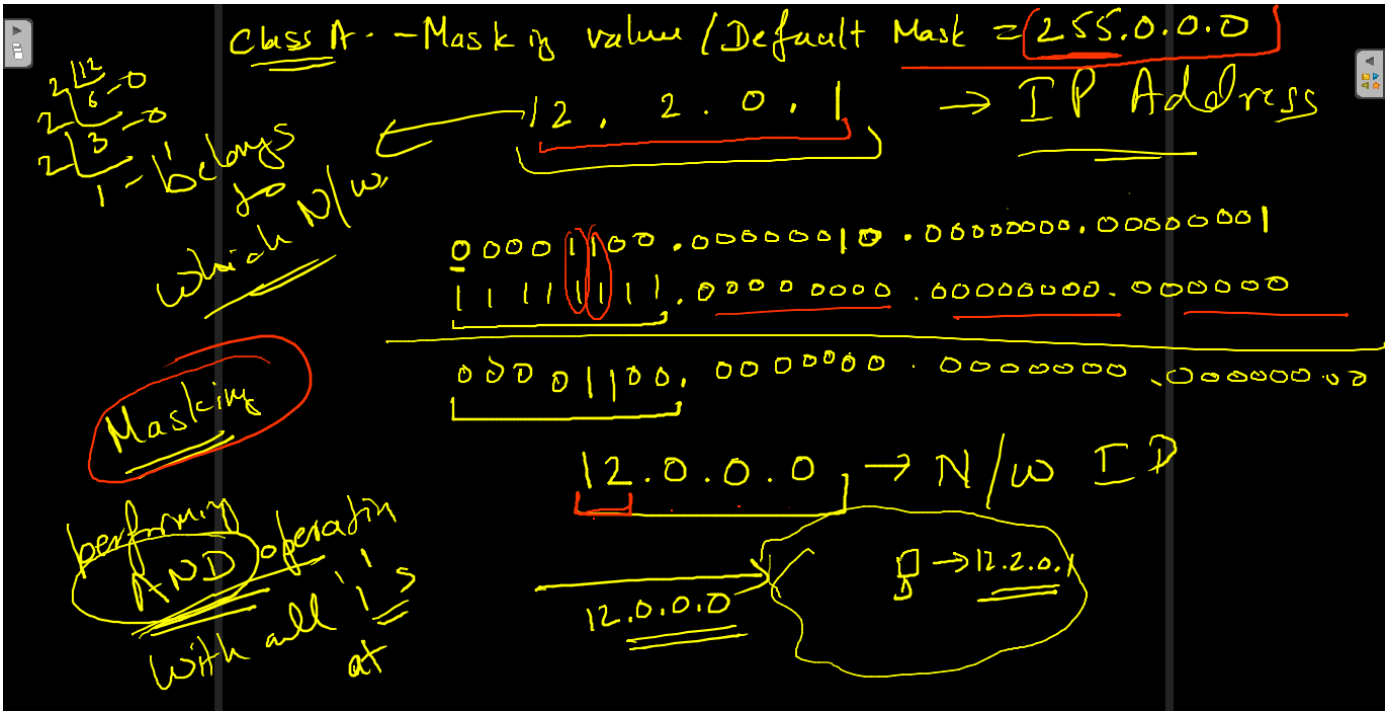
↳ CIDR
↳ VLSM

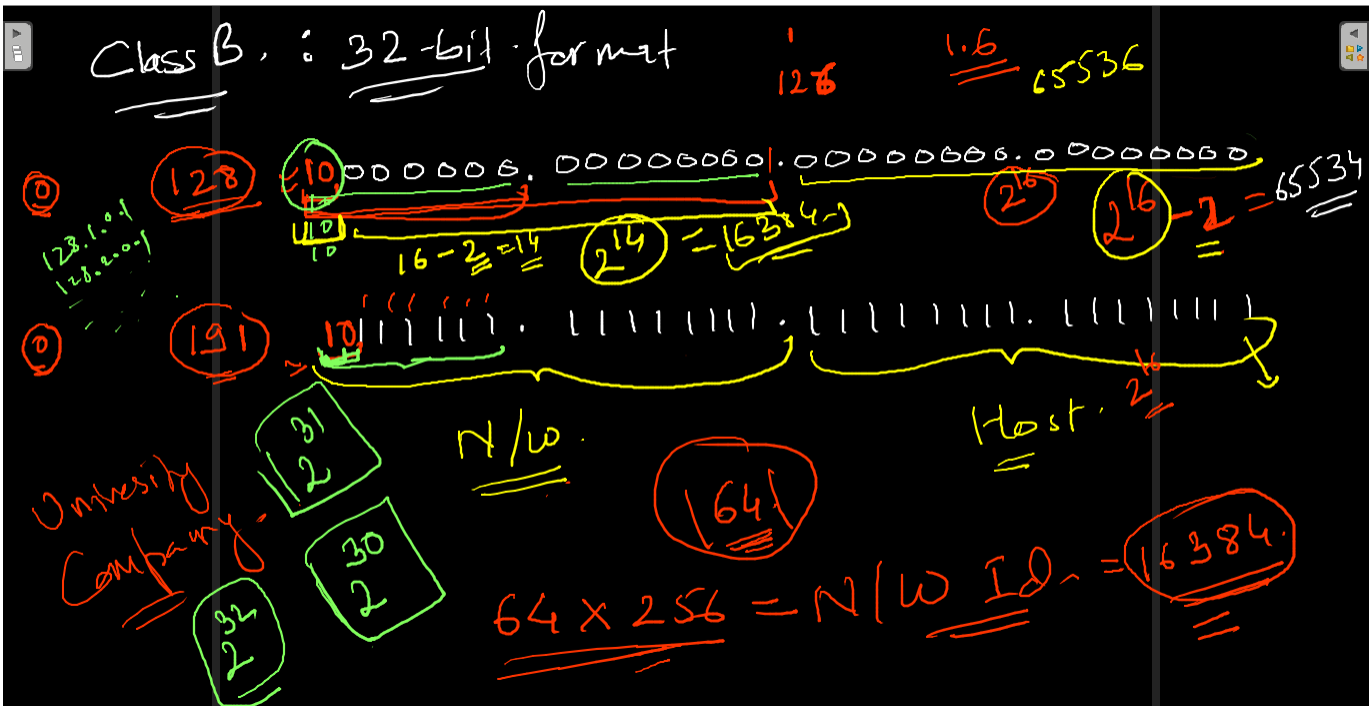

32-bit of IP address
IPV
You must be able to determine
N/W ID & HOST ID
IP address

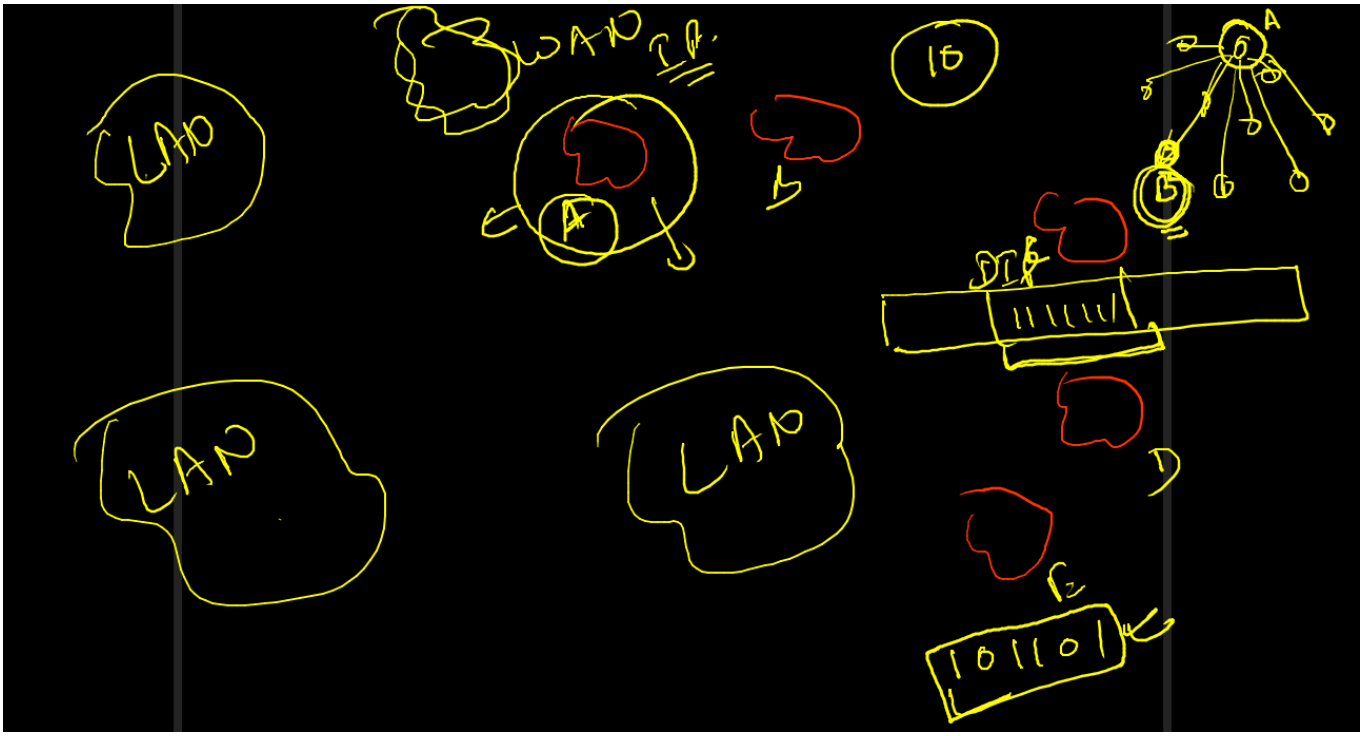

Class A
IANA → Internet assigned Numbers Authority
MSB bit
N/W's
reserved diagnostic
Host
N/W = 0 — 127 ∴ 128 - 2 ⇒ 126
2^32
2^31 → IP addresses are possible
127.0.0.0

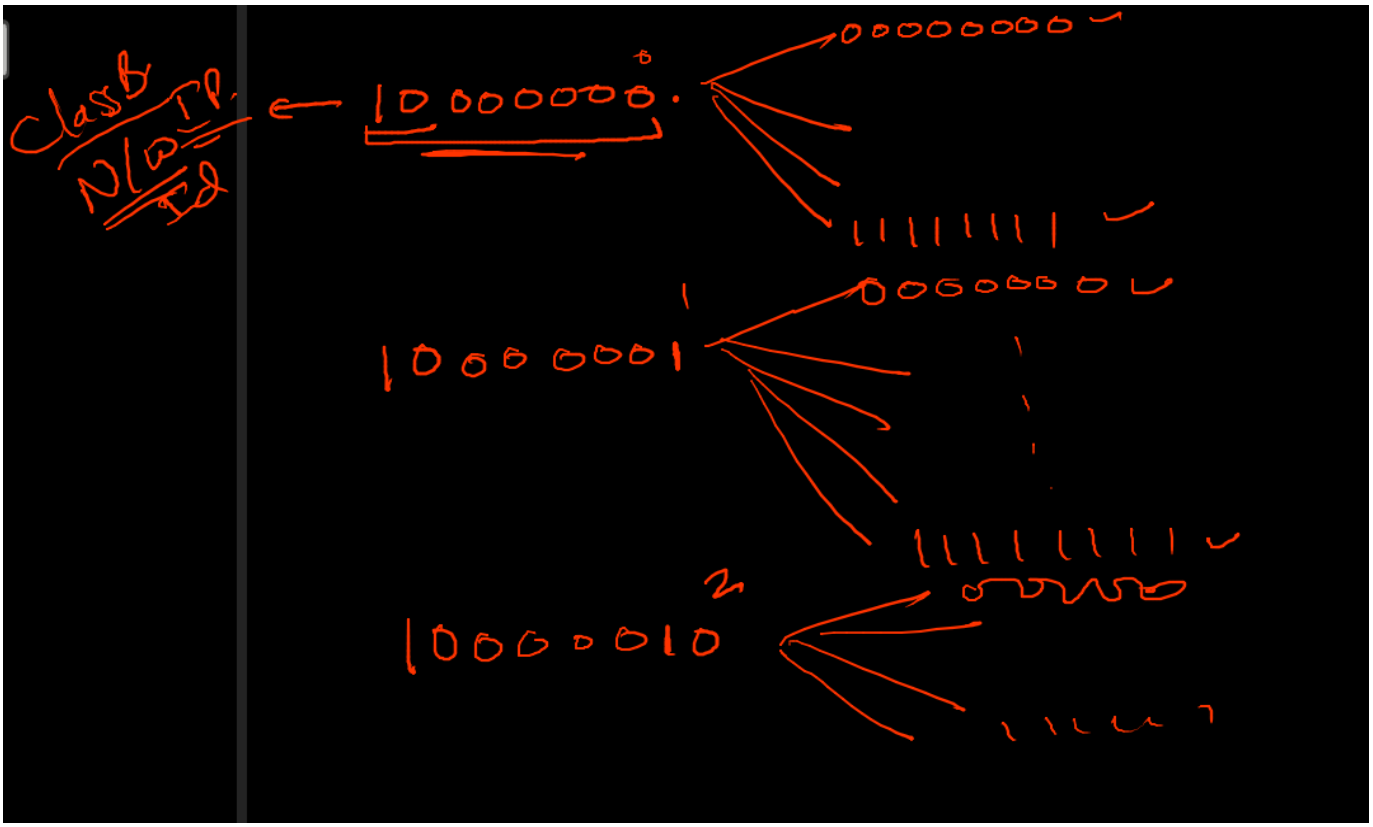

No. of Hosts
2^{24} = 1.6 millim IP addresses
16777216
Hostel A → A
B → 10
N/W Id.
default gateway
24
2
broadcast address.
Not given for individual IP address
Host individual IP add.

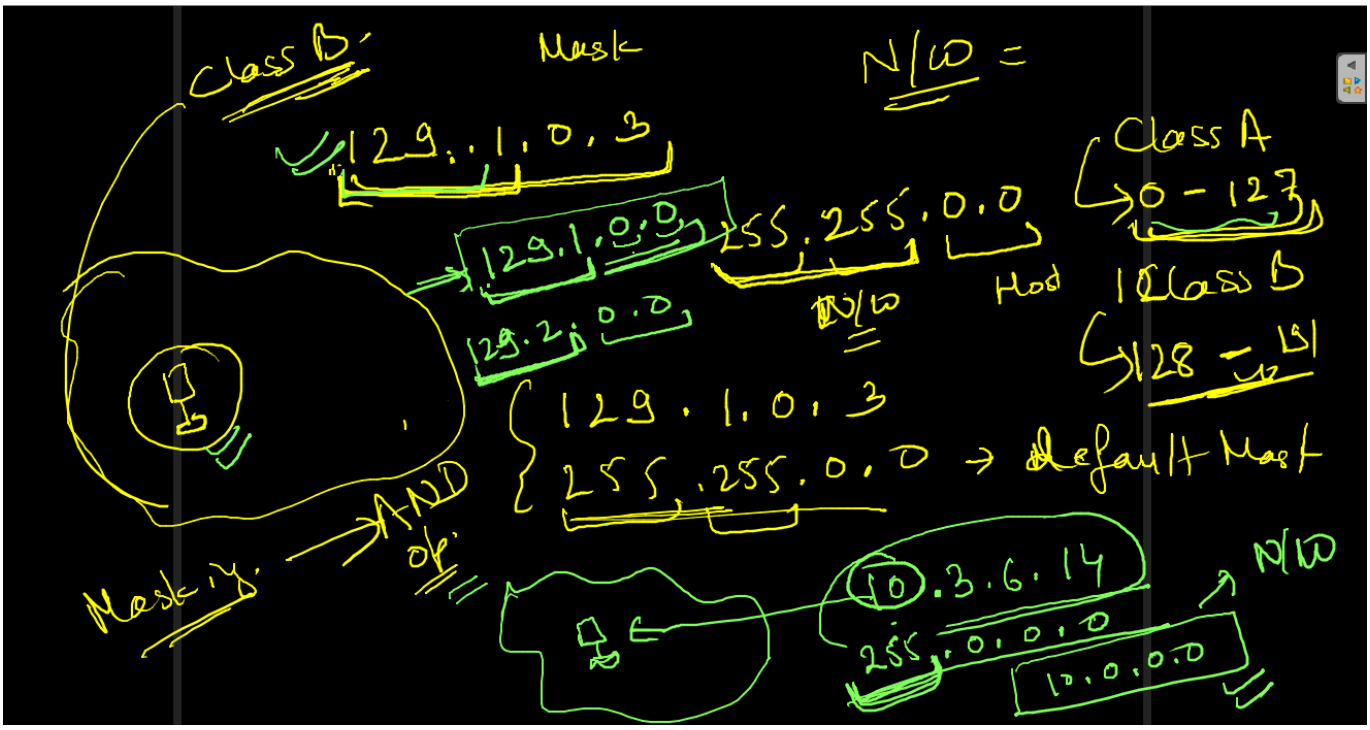

Class A
2^7 = 128 − 2
0 to 127
No. of Networks = 126
$2^{24} - 2$ =
Host
255
254
10.10.1.0
12.0.0.1
14.2.3.1
Class A
N/w

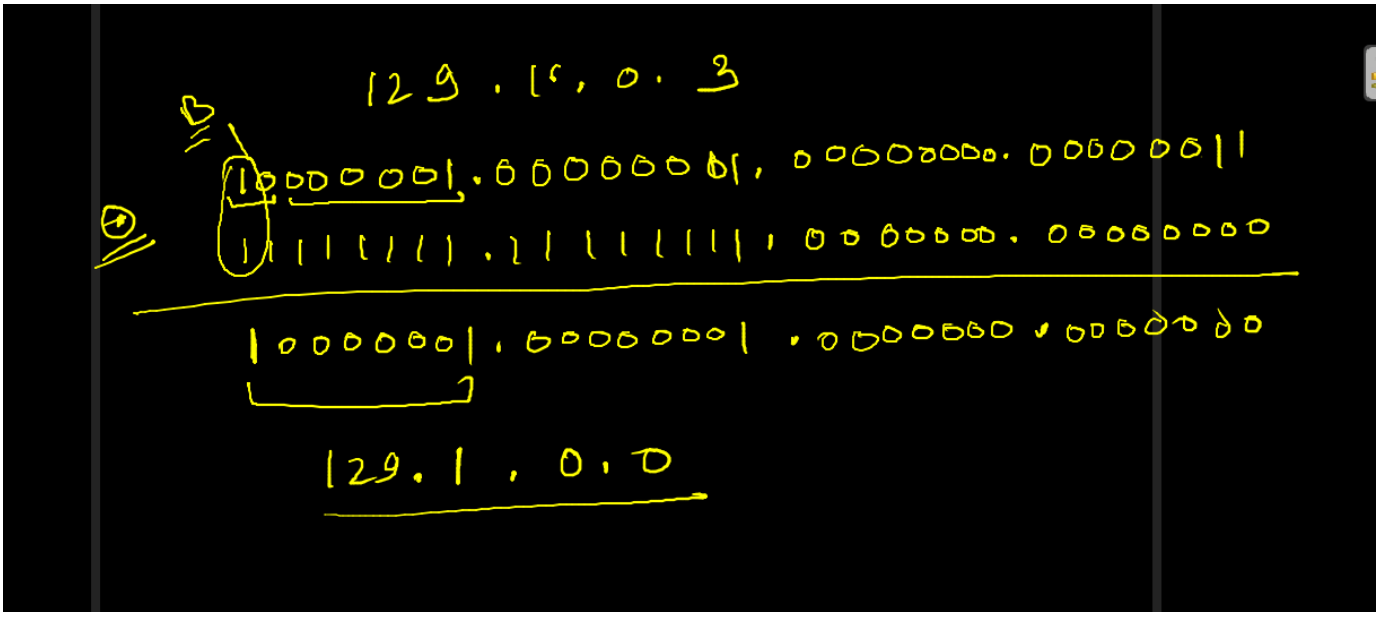

Class A. – Mask is value / Default Mask = 255.0.0.0
12 . 2 . 0 . 1 → IP Address
belongs to which N/w
Masking
performing AND operation with all 1's at
12.0.0.0 → N/w ID
12.0.0.0
12.2.0.1

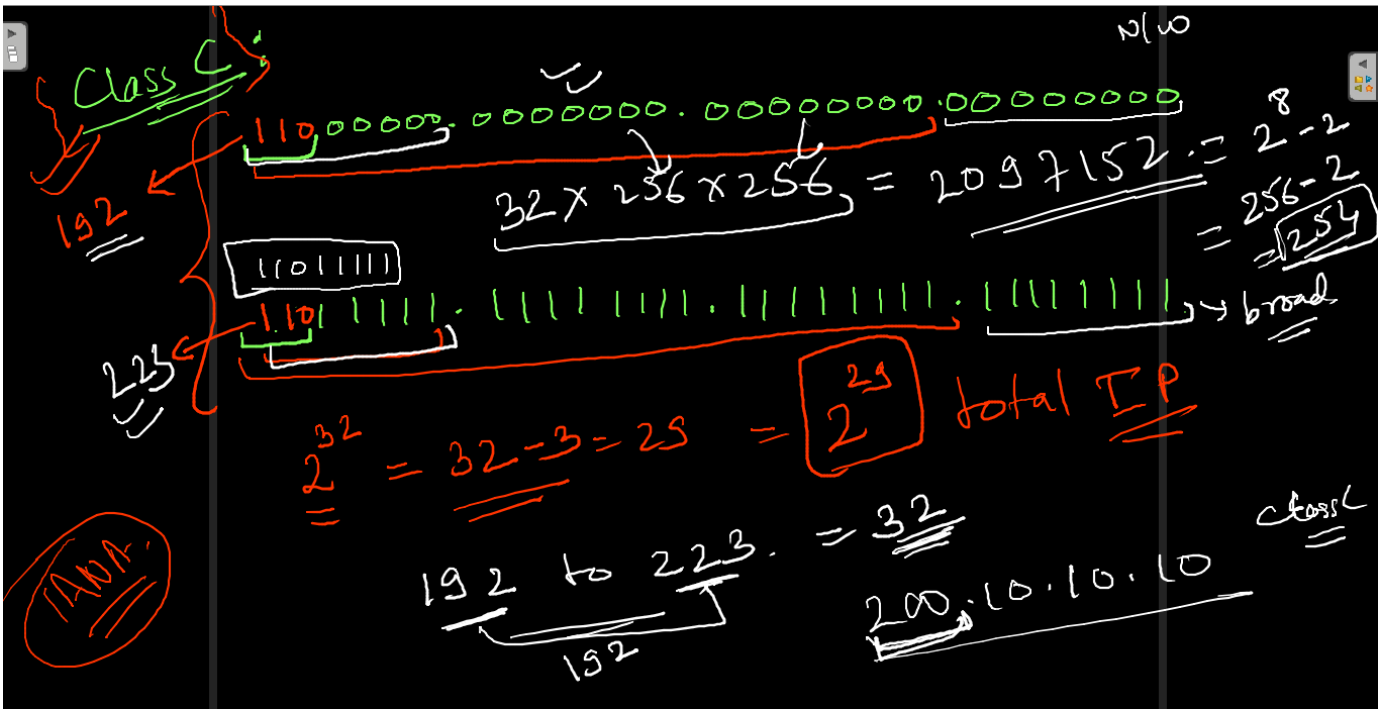

Class B. : 32-bit format
128
1.6
65536
128
10
16 – 2 = 14
2^{14} = 16384
2^{16} – 2 = 65534
191
N/w
Host
University
Company
64
64 × 256 = N/w Id. = 16384

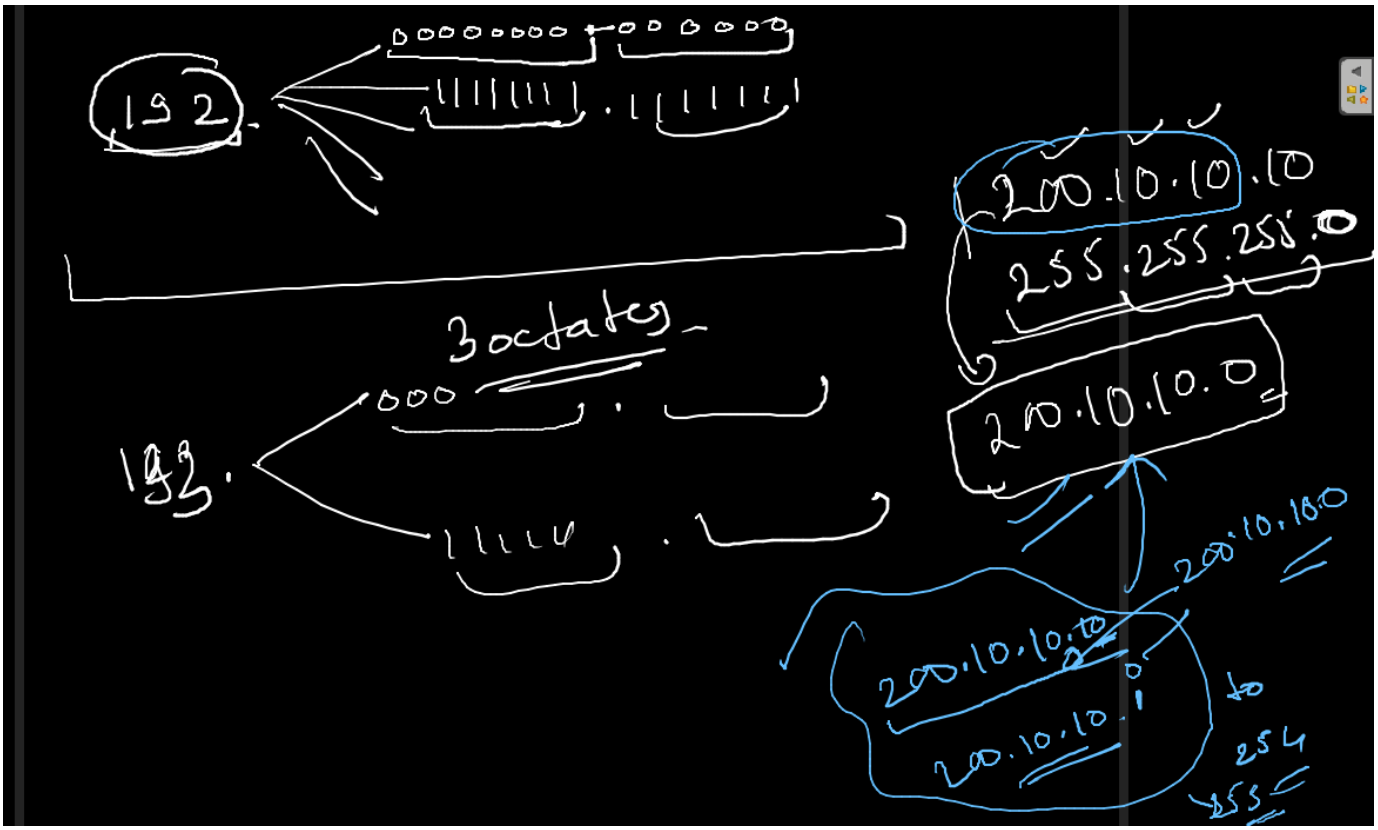

LAN
LAN
LAN
A
B
D
10110

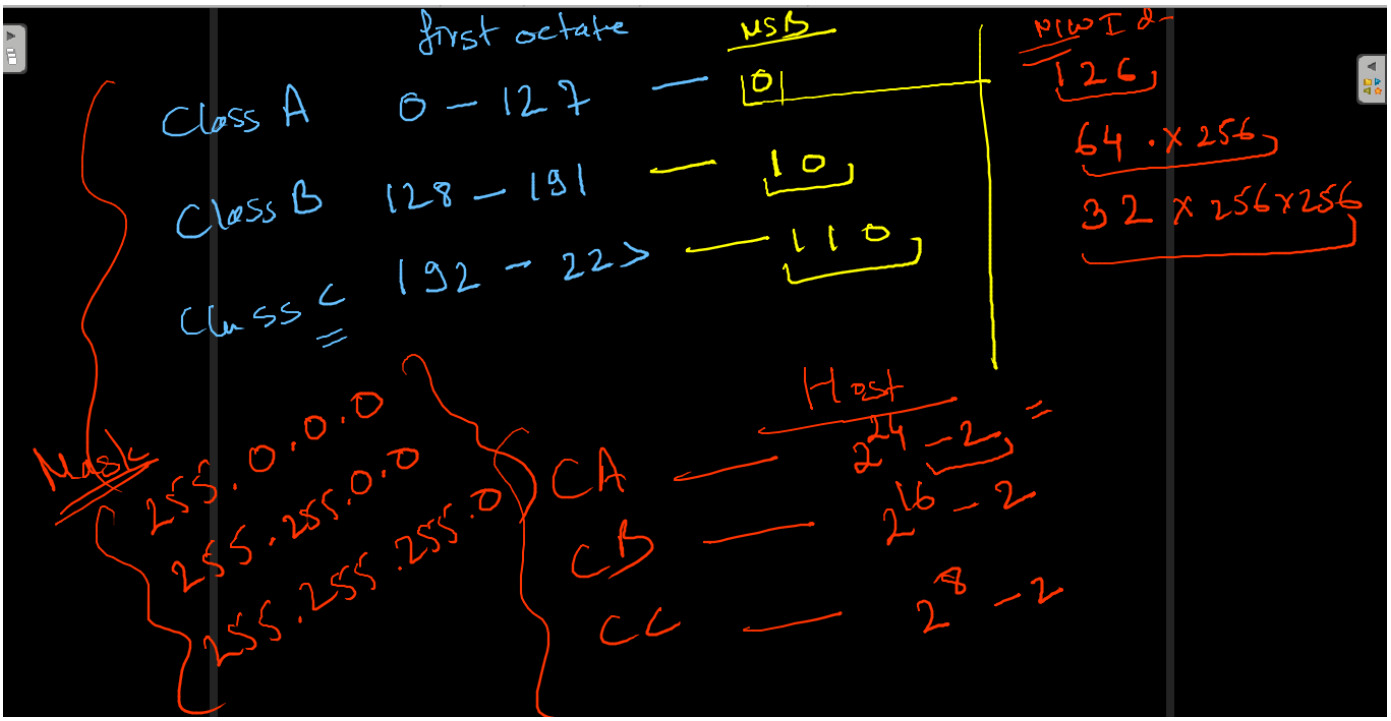

10000000
10000001
10000010

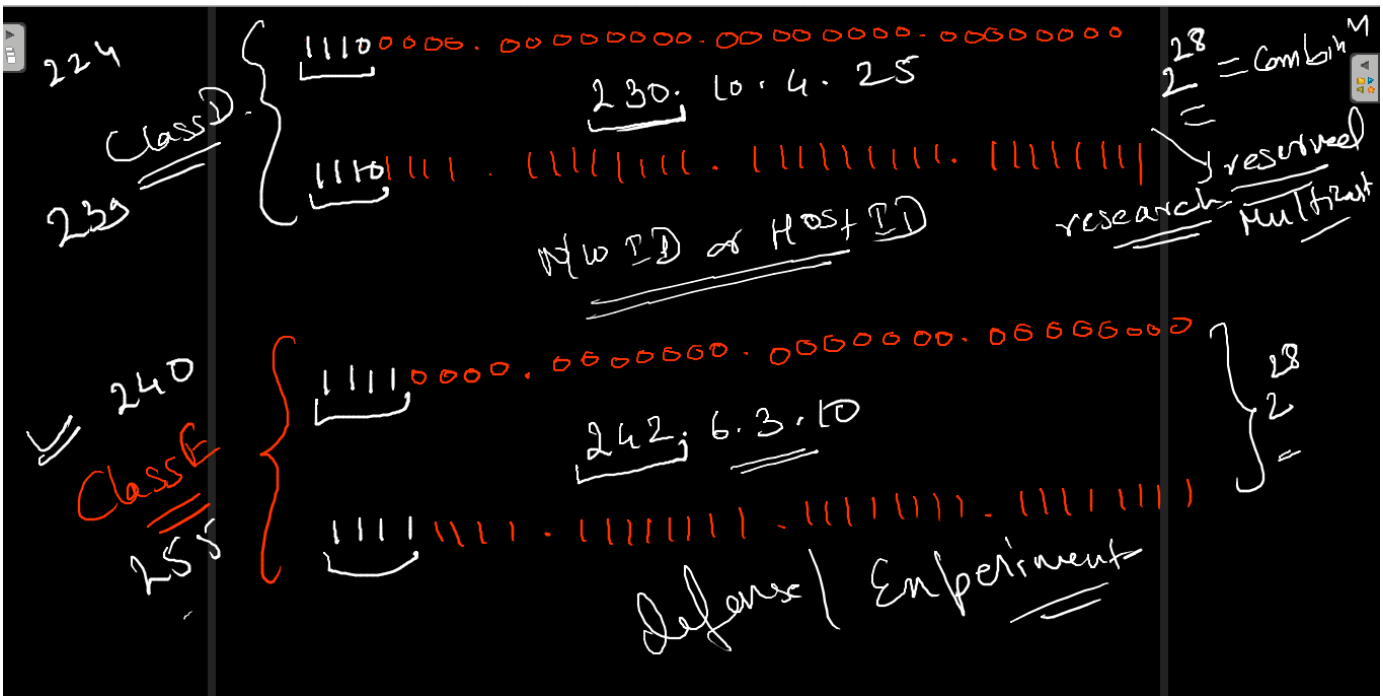

Class B
Mask
N/W =
129.1.0.3
129.1.0.0
255.255.0.0
N/W
Host
Class A
0 - 127
Class B
128 - 191
129.2.0.0
129.1.0.3
255.255.0.0 → default Mask
Mask 'ki'
AND op.
10.3.6.14
255.0.0.0
10.0.0.0
N/W

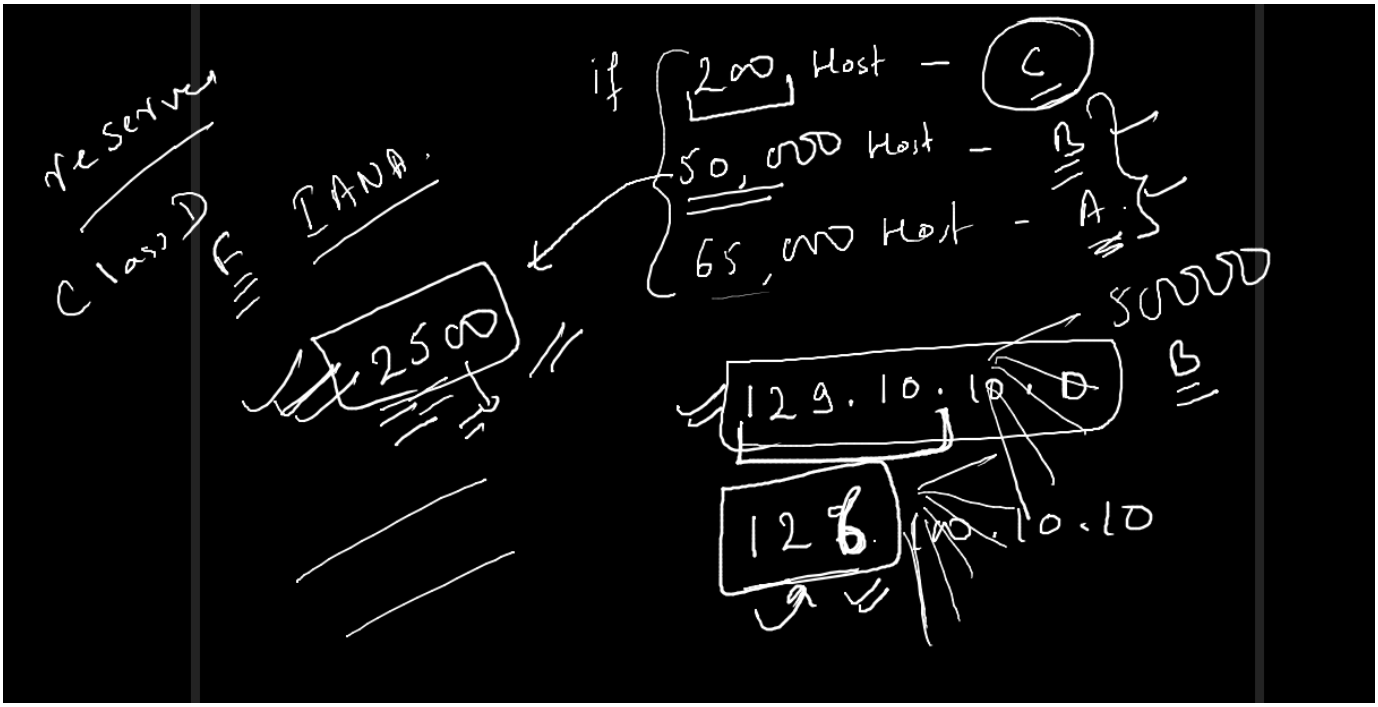

129.1.0.3
B
10000001.00000001.00000000.00000011
11111111.11111111.00000000.00000000
10000001.00000001.00000000.00000000
129.1.0.0

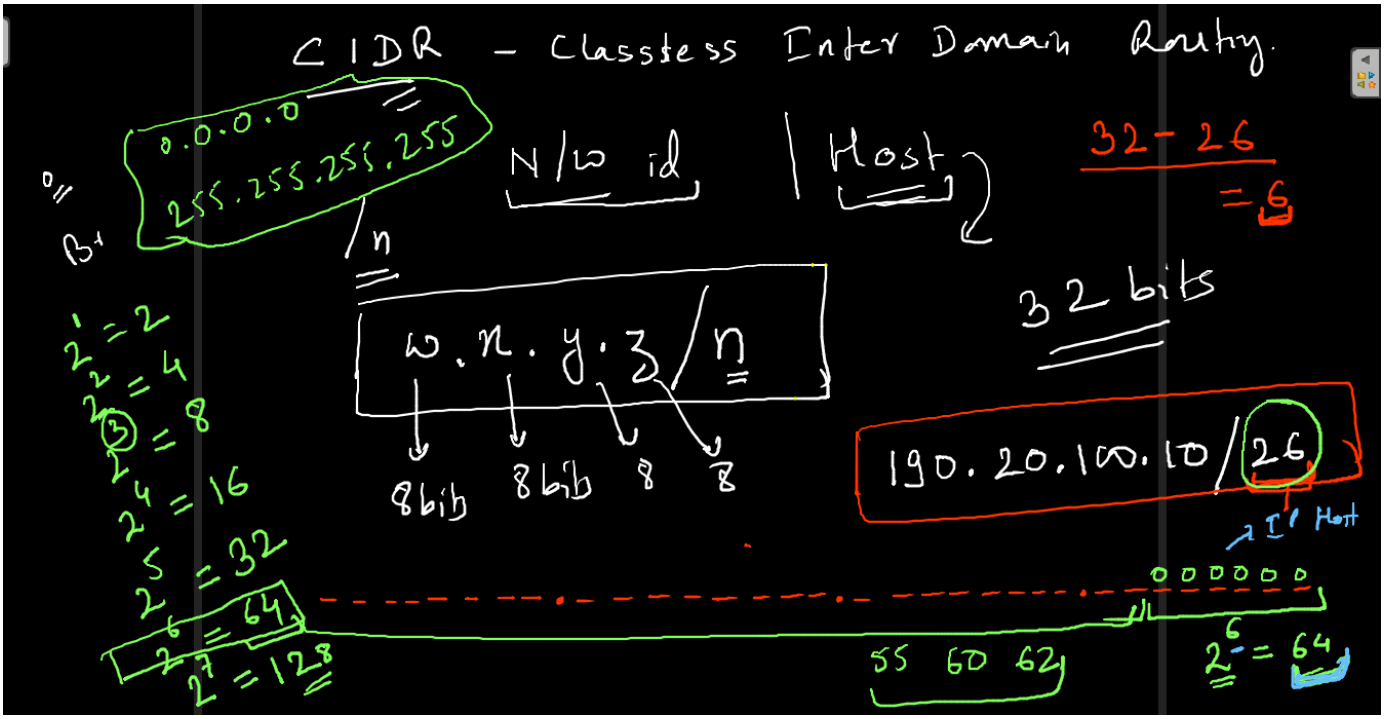

Class C
192
223
32 x 256 x 256 = 2097152
= 256 - 2
= 254
broad
total IP
2^{32} = 32 - 3 = 25 = 2^{29}
192 to 223 = 32
20.10.10.10
class C
IANA

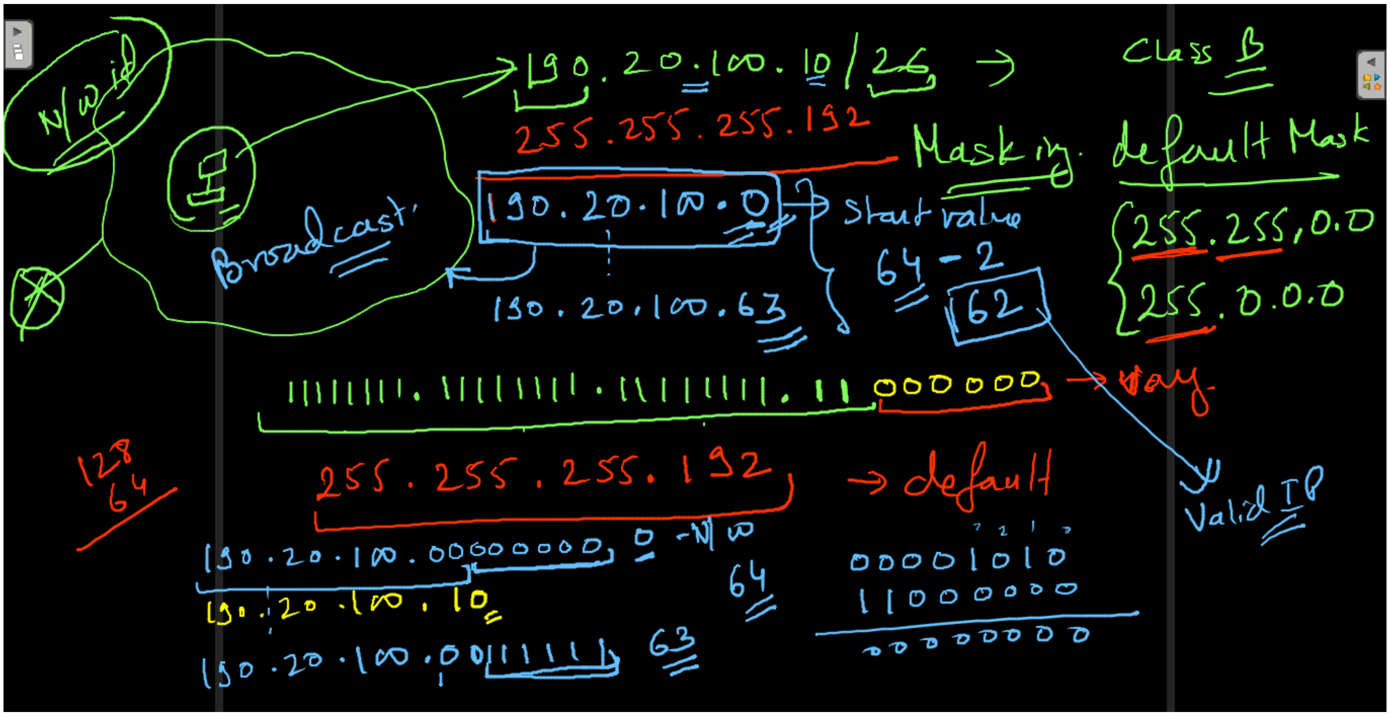

192.
3 octets
193.
20.10.10.10
255.255.255.0
20.10.10.0
20.10.10.0
20.10.10.1
to
254

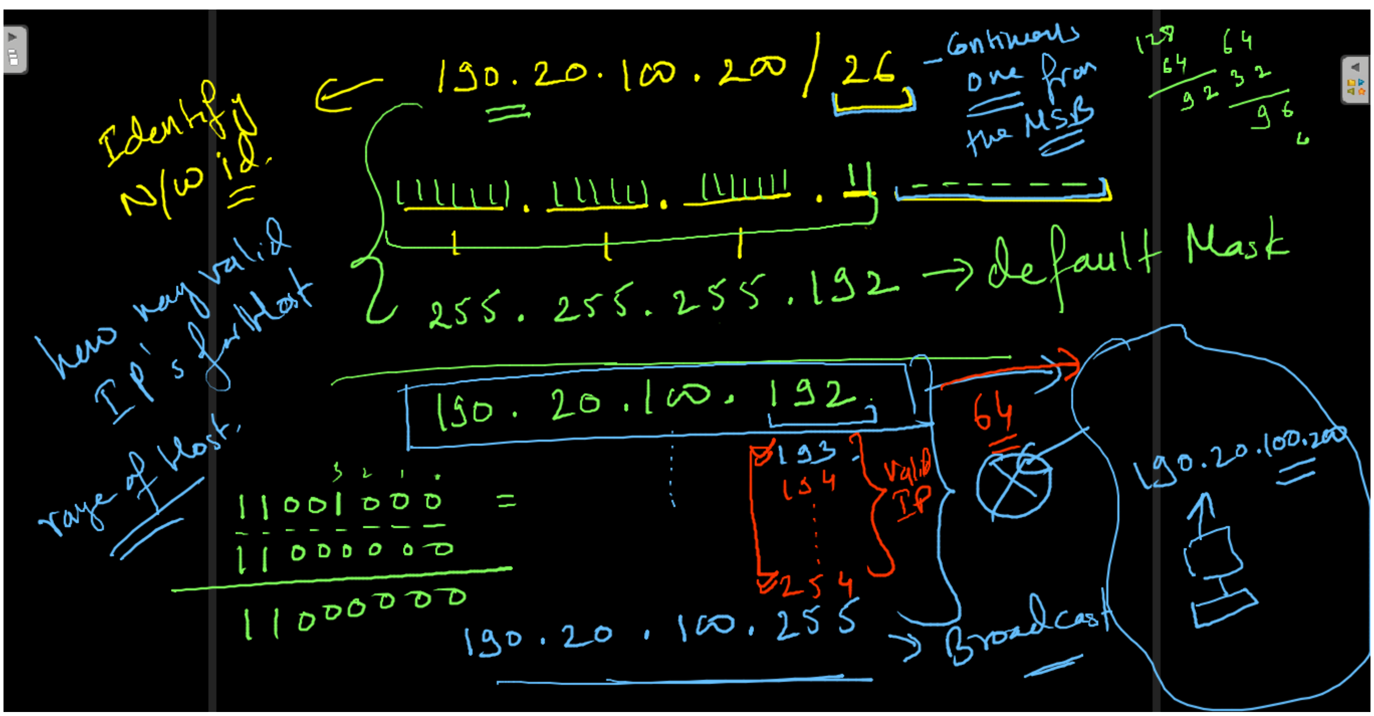

first octate
MSB
Class A 0 – 127 — 0
Class B 128 – 191 — 10
Class C 192 – 223 — 110
NW ID
126
64 · X 256
32 X 256 X 256
Mask
255.0.0.0
255.255.0.0
255.255.255.0
Host
CA — 2^24 – 2
CB — 2^16 – 2
CC — 2^8 – 2

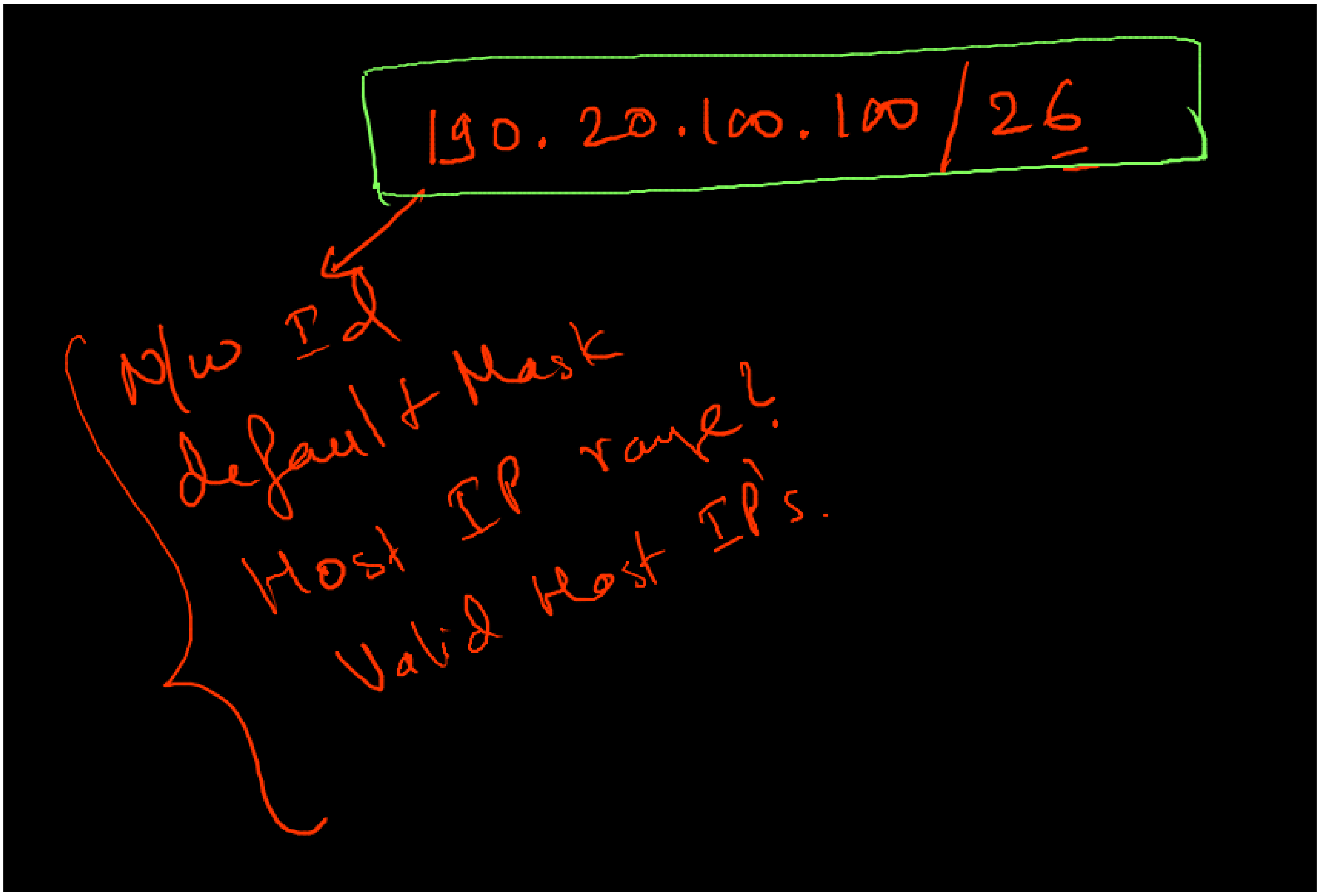

224
Class D
239
11100000.00000000.00000000.00000000
230. 10. 4. 25
1110 1111 . 1111 1111 . 1111 1111 . 1111 1111
NW ID or Host ID
2^28 = Combin^N
reserved
research
Multicast
240
Class E
255
1111 0000 . 0000 000 . 0000 000 . 0000 0000
242. 6. 3. 10
1111 1111 . 1111 1111 . 1111 1111 . 1111 1111
2^28
Defense / Experiment

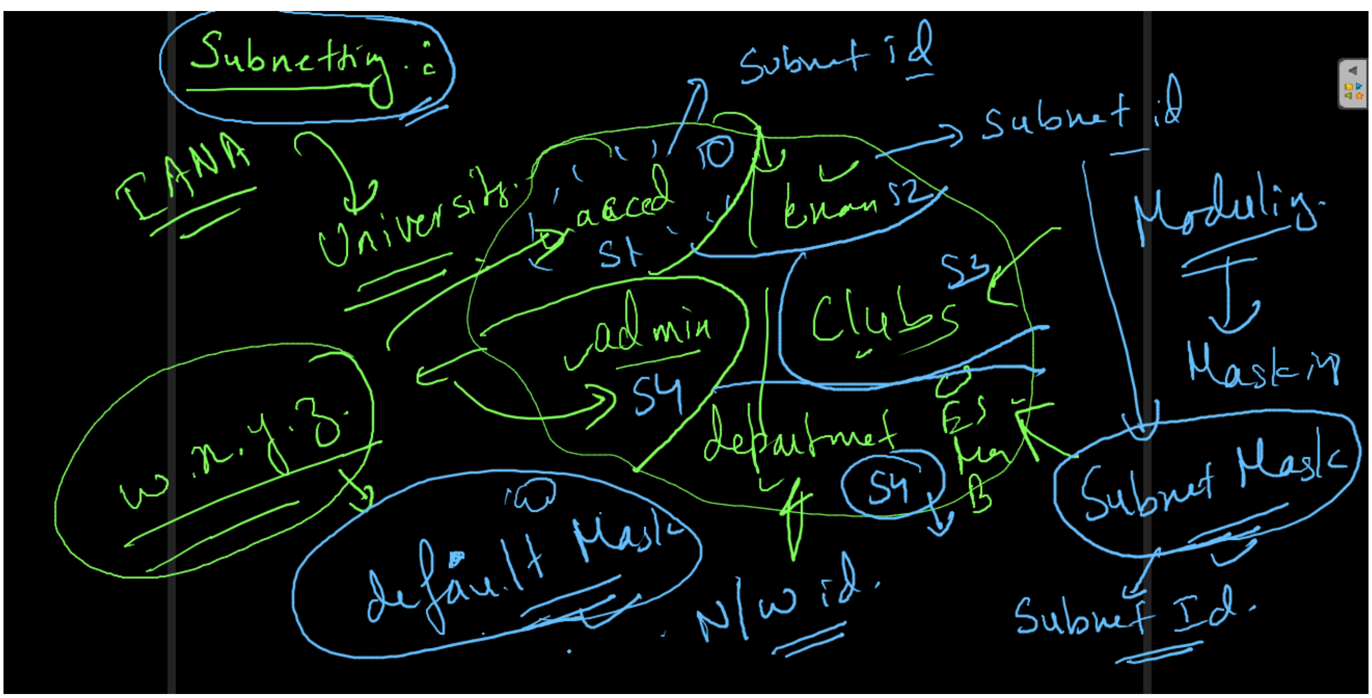

reserved
Class D & IANA.
if
200 Host - C
50,000 Host - B
65,000 Host - A
2500
129.10.10.0
126.10.10.10
65000
B

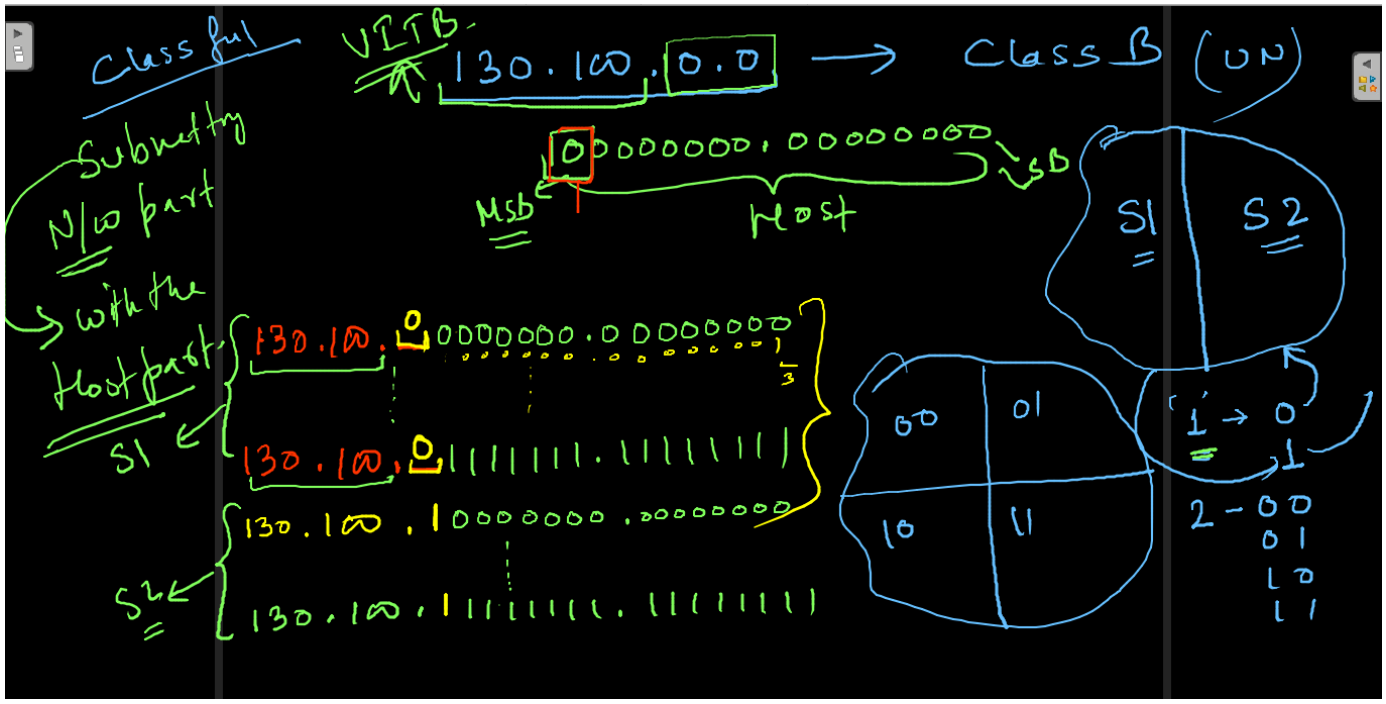

CIDR - Classless Inter Domain Routing.
0.0.0.0
255.255.255.255
Network id
Host
32 - 26 = 6
32 bits
n
w.x.y.z/n
8bit 8bit 8 8
190.20.100.10/26
Host
000000
55 60 62
2^6 = 64
2^1 = 2
2^2 = 4
2^3 = 8
2^4 = 16
2^5 = 32
2^6 = 64
2^7 = 128

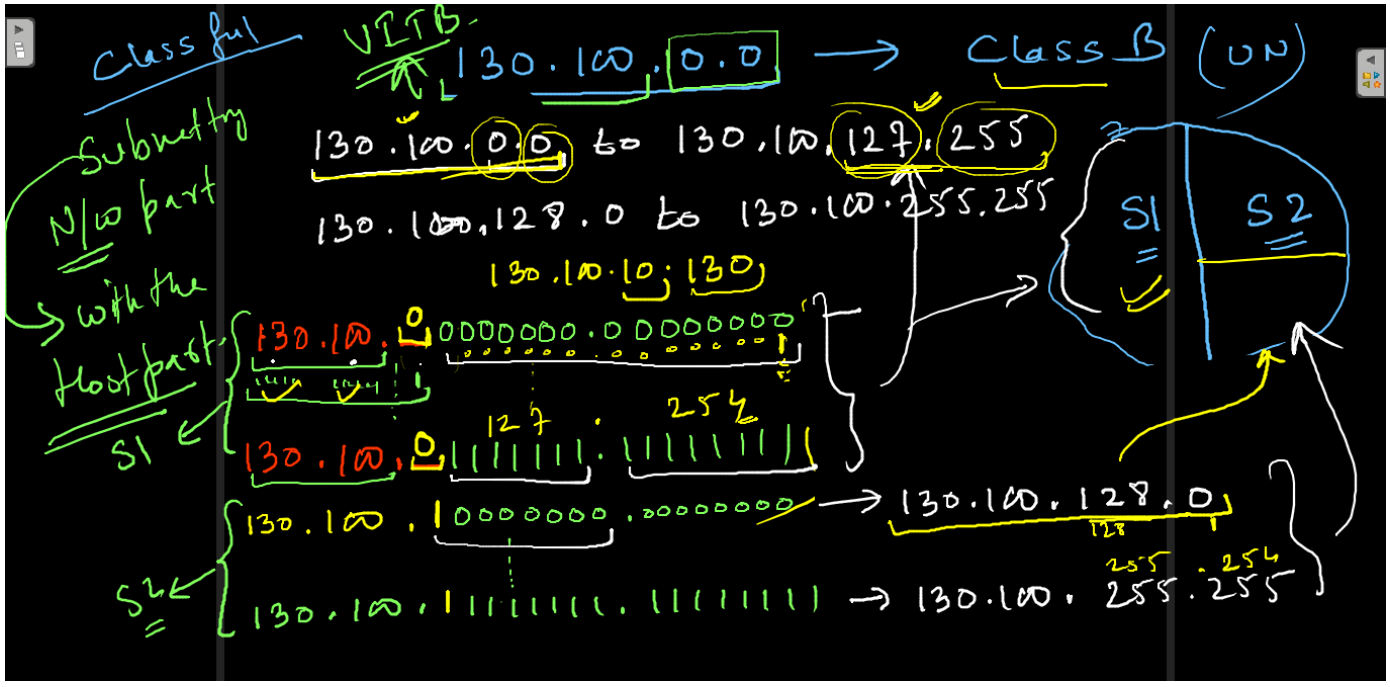

N/w id
Broadcast
190.20.100.10/26 →
Class B
255.255.255.192
Masking.
default Mask
255.255.0.0
255.0.0.0
190.20.100.0 → Start value
190.20.100.63
64 − 2
62
11111111.11111111.11111111.11000000 → way
255.255.255.192 → default
Valid IP
128
64
190.20.100.00000000 0 - N/w
190.20.100.10
190.20.100.00111111 63
64
00001010
11000000
00000000

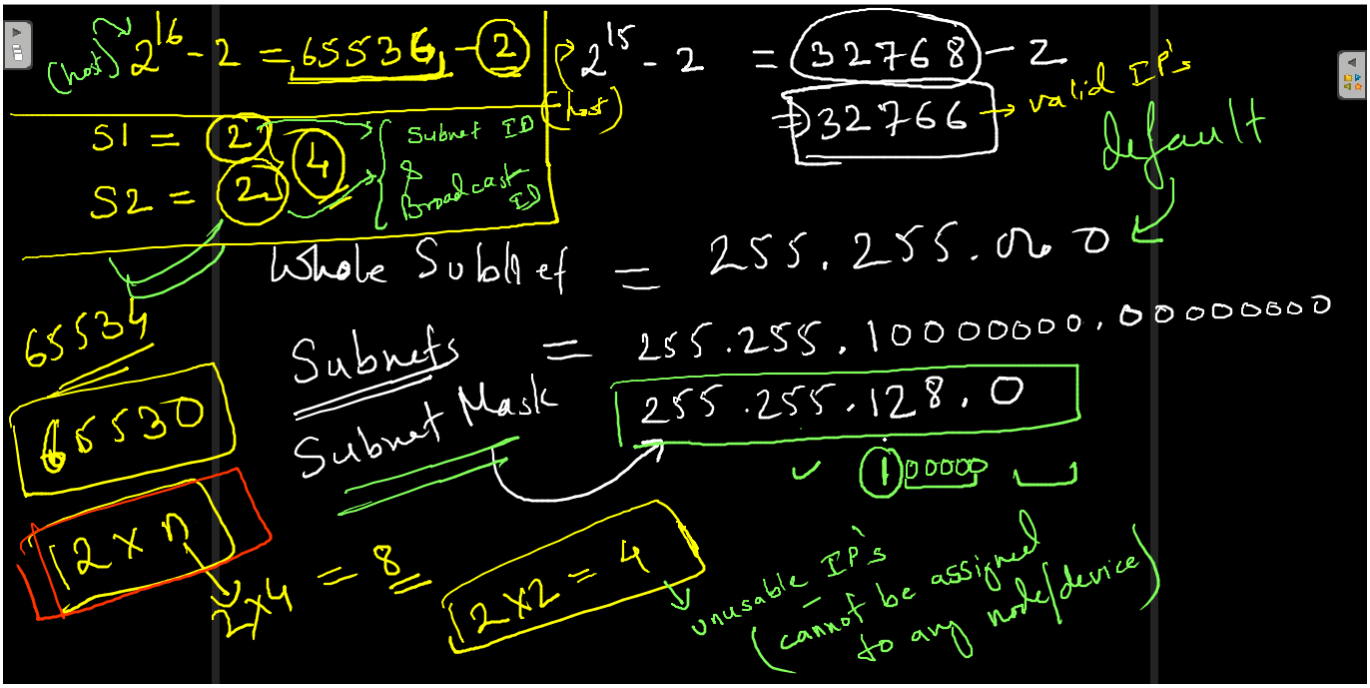

190.20.100.200/26
Continuous one from the MSB
Identify N/w id
how many valid IP's for Host
range of Host.
255.255.255.192 → default Mask
190.20.100.192
64
193
194
254
Valid IP
11001000
11000000
11000000
190.20.100.255 → Broadcast
190.20.100.200

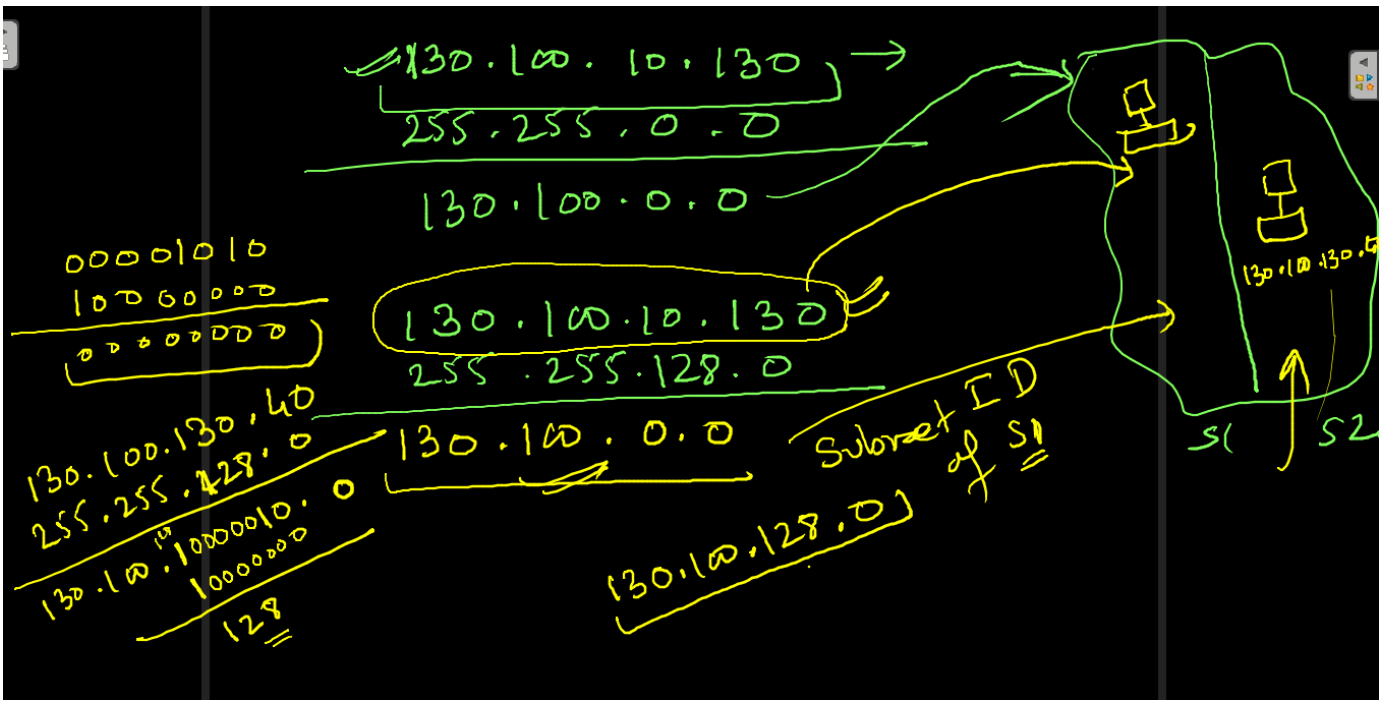

190.20.100.100/26
N/w Id
default Mask
Host IP range?
Valid Host IP's.


Subnetting:
IANA
University
Subnet id
Subnet id
placed St
exam S2
Clubs S3
admin S4
department
S4
Modulig.
Mask id
w.x.y.z.
default Mask
N/w id.
Subnet Mask
Subnet Id.


Classful
VITB
130.100.0.0
Class B (UN)
Subnetting
N/w part
with the
Host part
MSb
LSb
Host
S1
S2
130.100.0000000.00000000
130.100.11111111.11111111
130.100.10000000.00000000
130.100.11111111.11111111
00
01
10
11
1 → 0
1
2-00
01
10
11


Classful
VITB
130.100.0.0
Class B (UN)
Subnetting
N/w part
with the
Host part
130.100.0.0 to 130.100.127.255
130.100.128.0 to 130.100.255.255
130.100.10;130
S1
S2
130.100.0000000.00000000
130.100.11111111.11111111
127
254
130.100.10000000.00000000 → 130.100.128.0
130.100.11111111.11111111 → 130.100.255.254

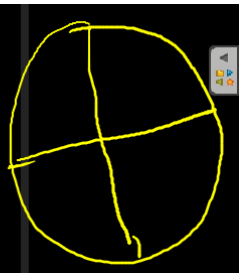

(host) $2^{16} - 2 = 65536$ — (2)
$2^{15} - 2 = 32768 - 2$
(host)
$= 32766$ → valid IP's
S1 = (2) (4) Subnet ID & Broadcast ID
S2 = (2) (4)
default
Whole Subnet = 255.255.0.0
65534
Subnets = 255.255.10000000.00000000
65530
Subnet Mask
255.255.128.0
(1)0000
2 X 1
2X4 = 8
2X2 = 4
unusable IP's (cannot be assigned to any node/device)


130.100.10.130
255.255.0.0
130.100.0.0
00001010
10060000
00000000
130.100.10.130
255.255.128.0
130.100.130.40
255.255.128.0
130.100.10000010.0
10000000
128
130.100.0.0
Subnet ID of S1
130.100.128.0
130.100.130.5
S1
S2

130.100.0.0 - CB

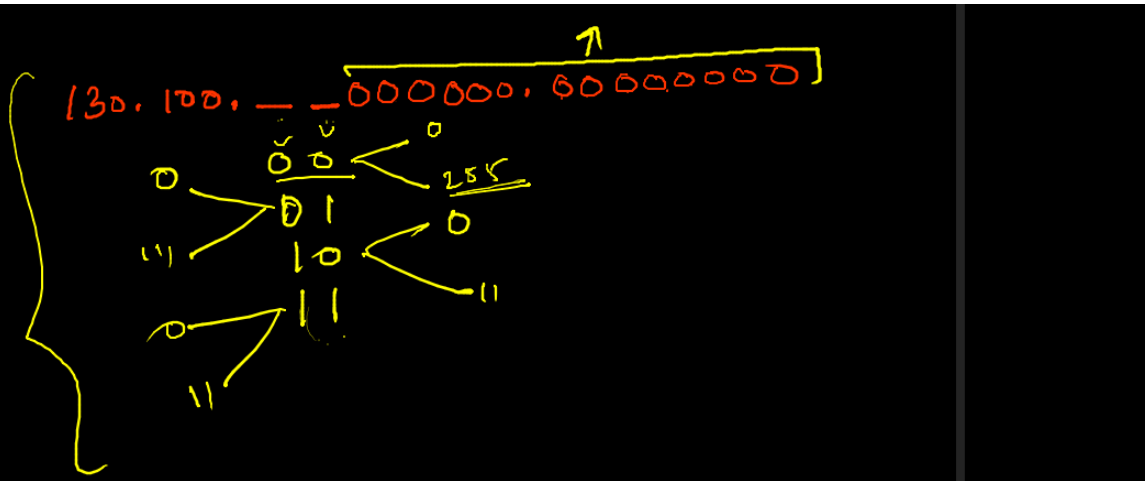
130.100.___000

130.100.0.0 — IP

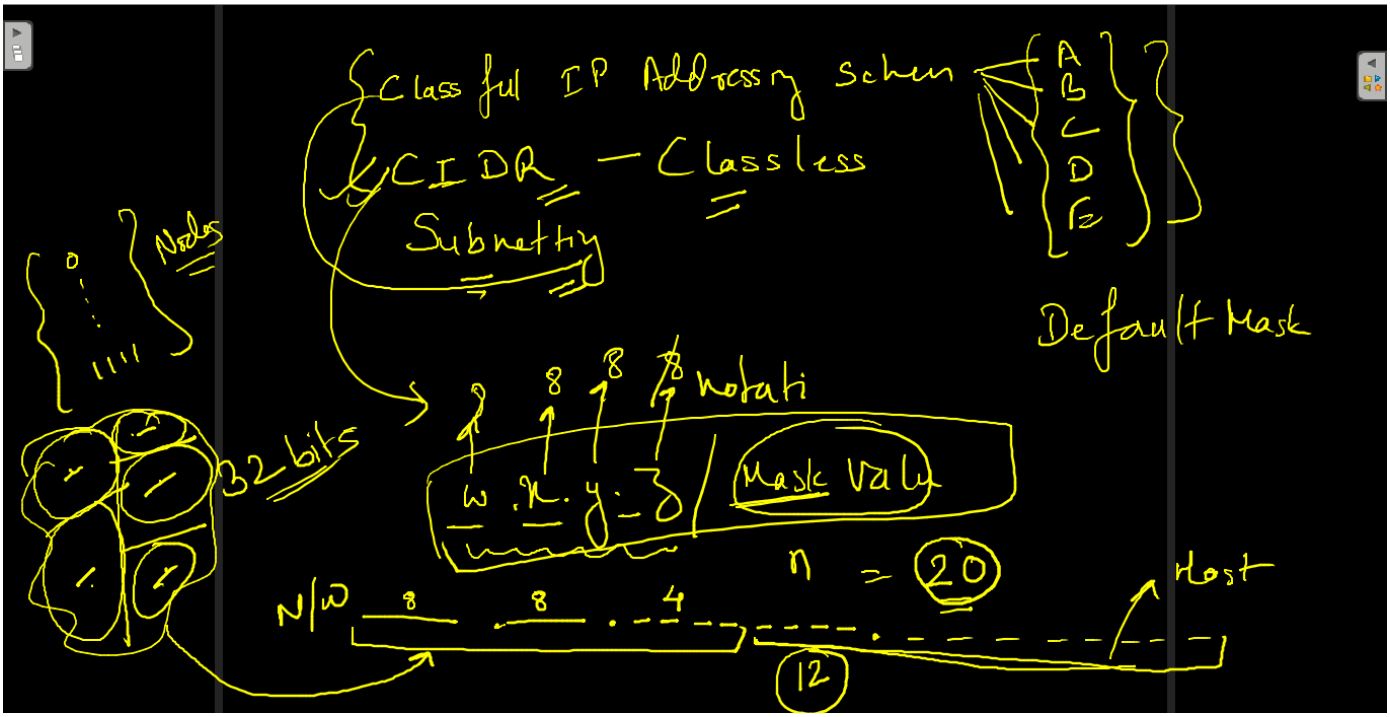


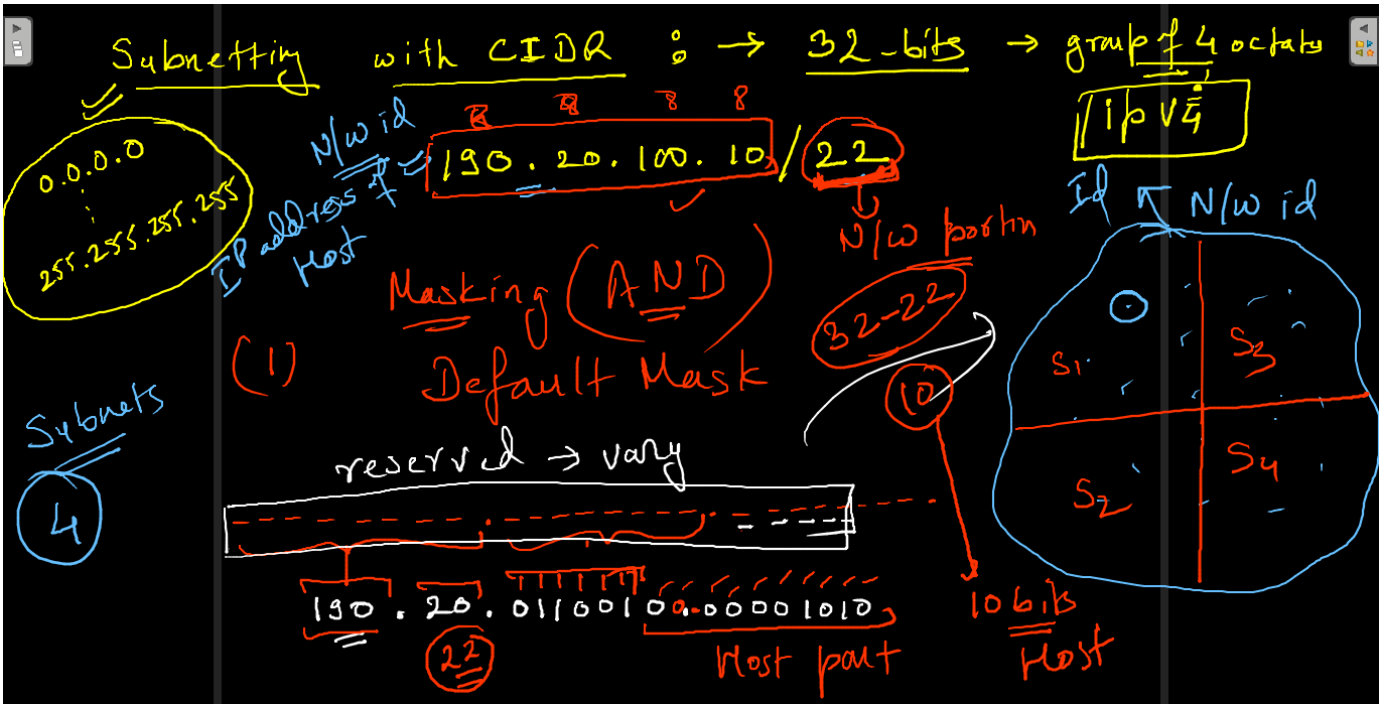
4 Subnets

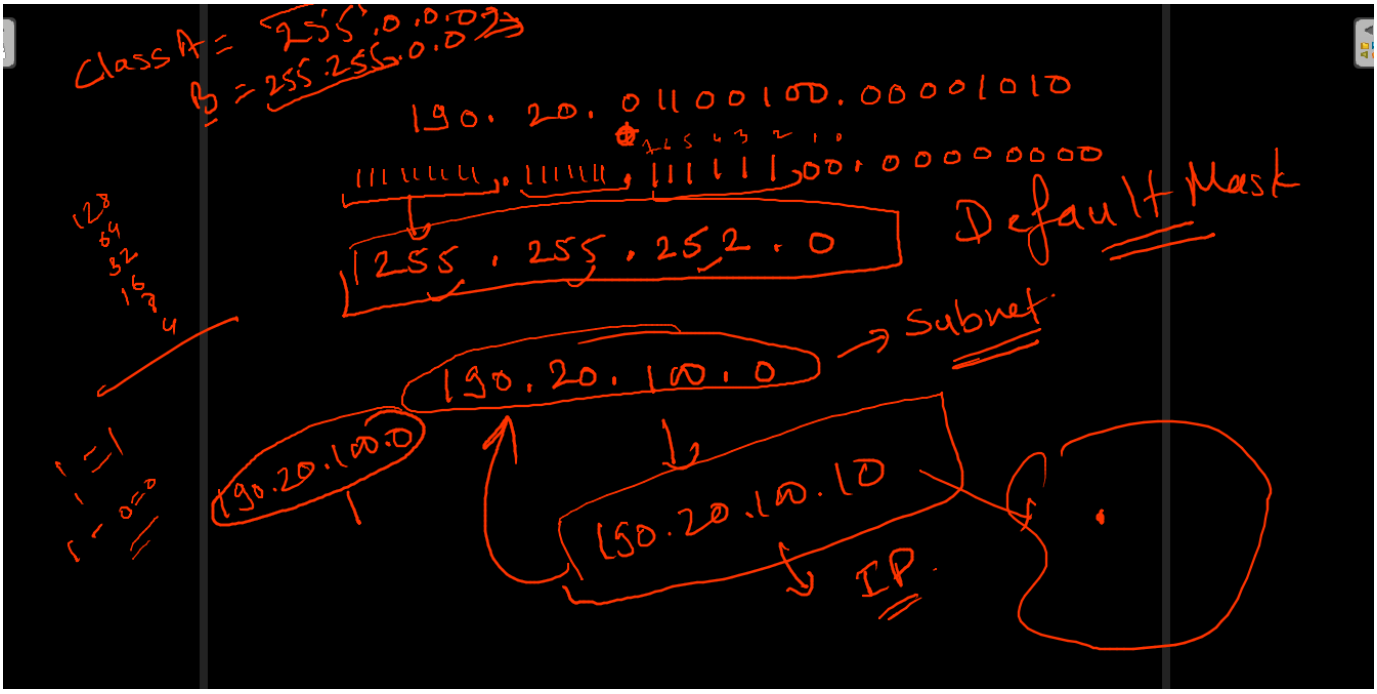
1. What will be the range of each Subnet
2. Subnet Mask
3. What will be the valid IP addresses in each subnet
4. How many IP's will not be utilize

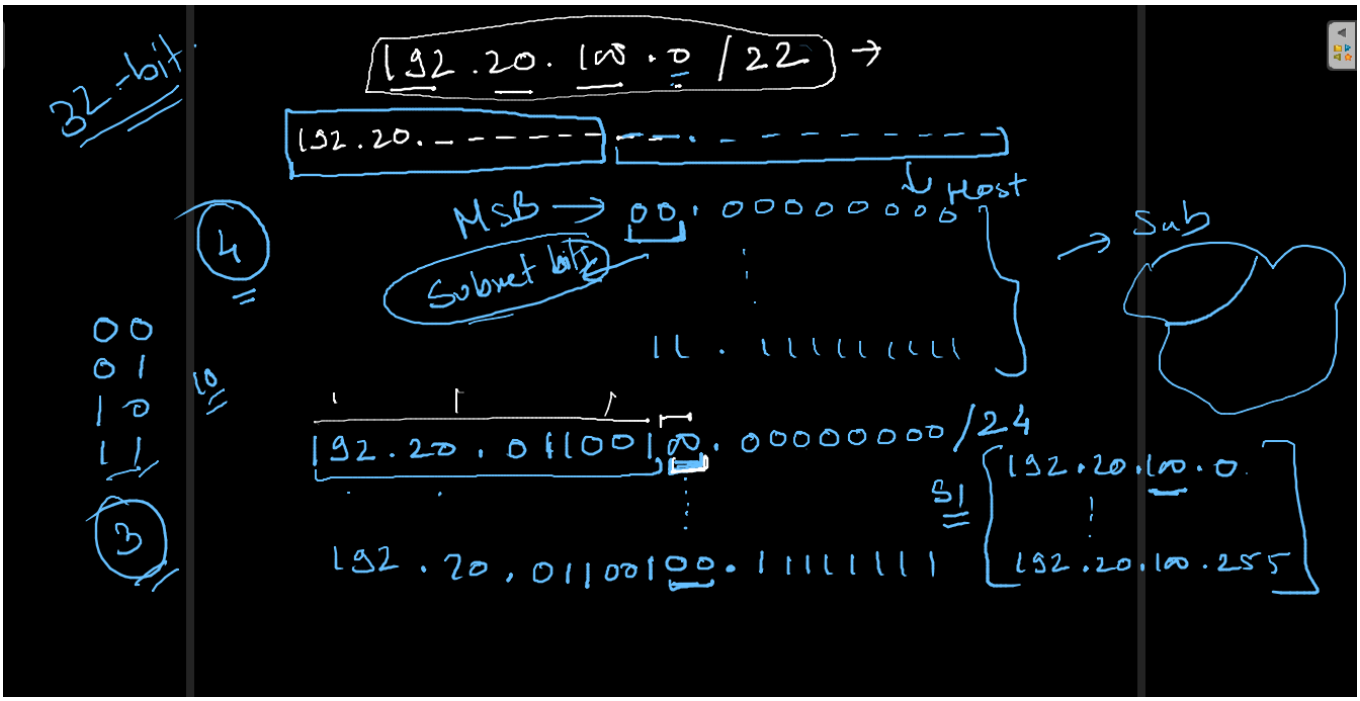


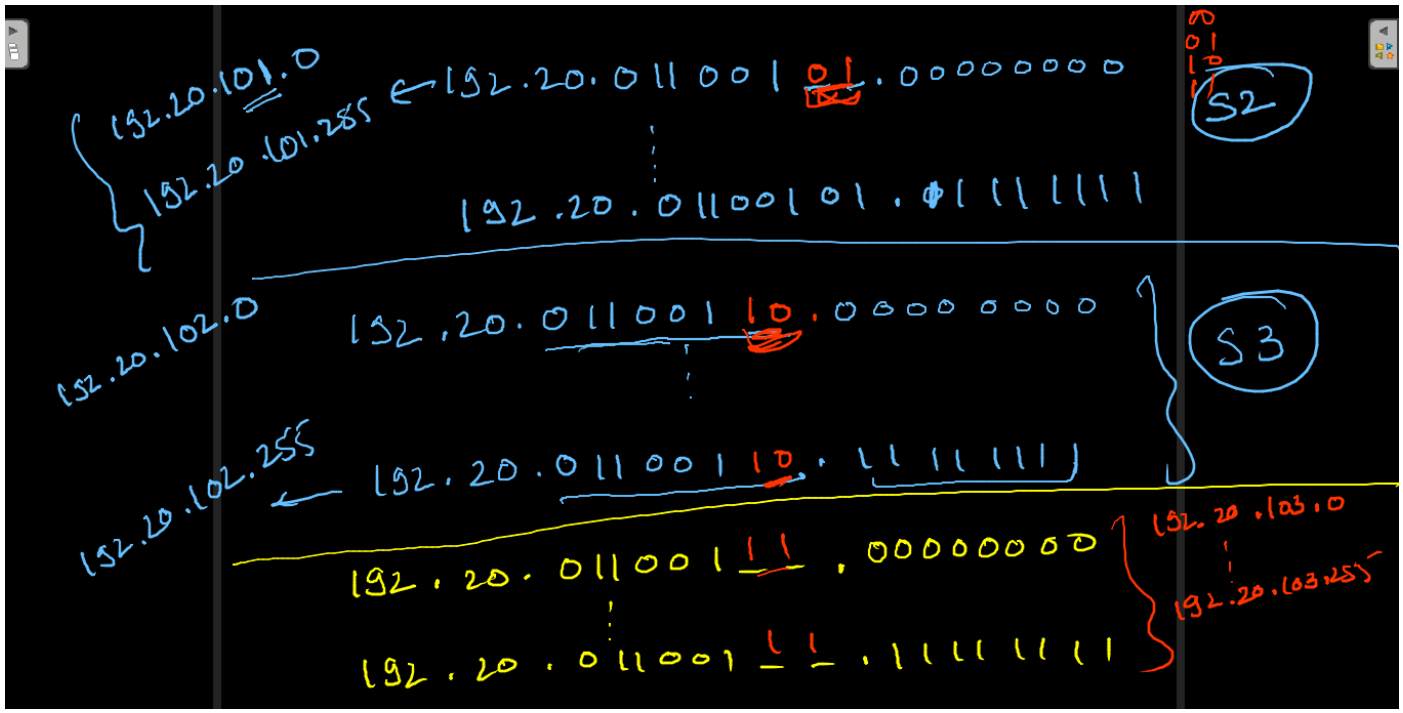
ie you need to see how many bits required to create 4 subnets from the host part. [Here 2 bits] $\Rightarrow 2^2 = 4$

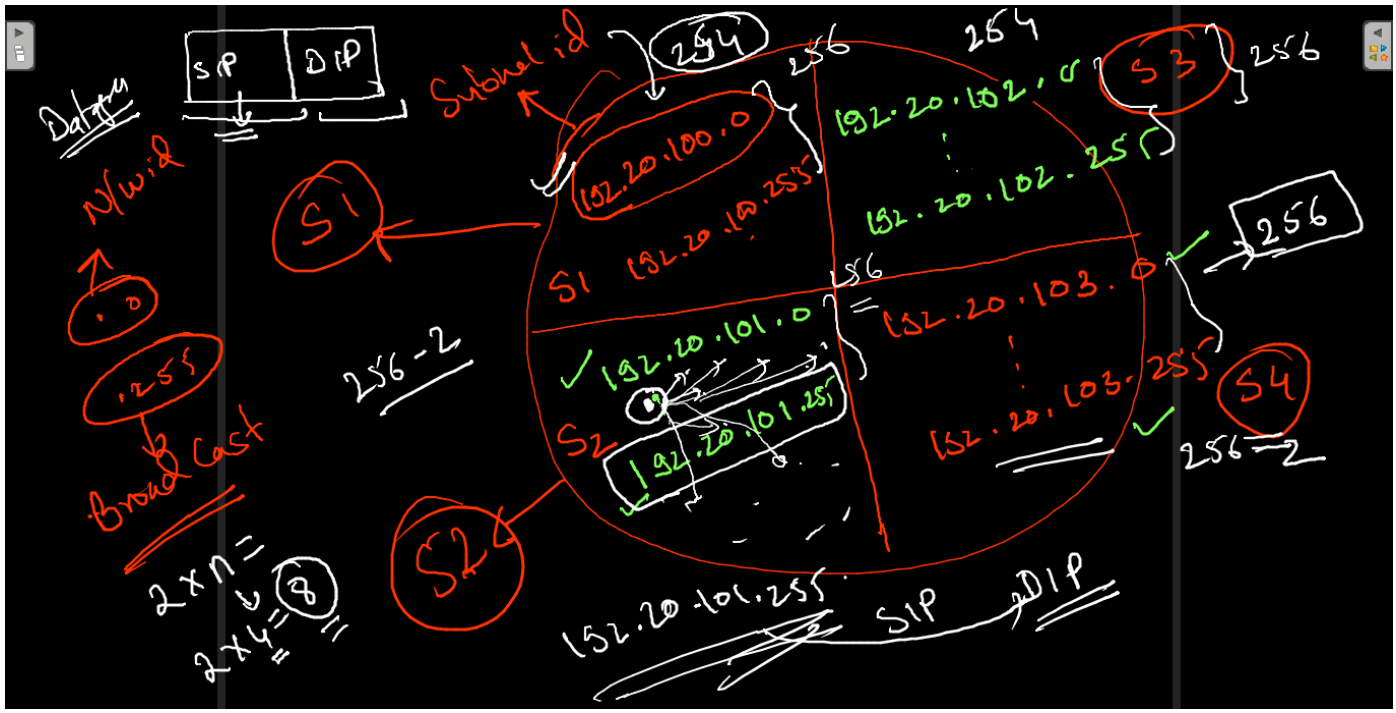

Classful IP Addressing Scheme
CIDR — Classless
Subnetting
A
B
C
D
E
Nodes
0
1111
Default Mask
32 bits
8
8
8
Notati
w.x.y.z/ Mask Valu
n = 20
N/W
8
8
4
Host
12

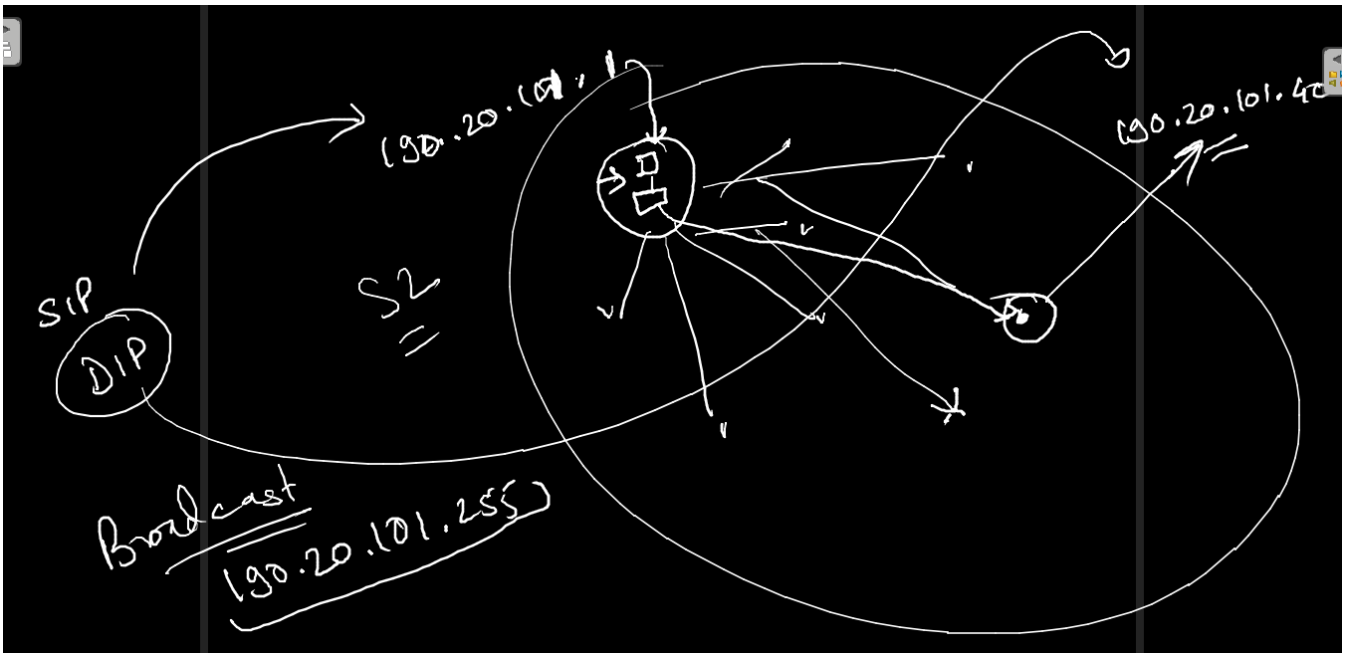

Subnetting with CIDR : → 32-bits → group of 4 octats
IPV4
0.0.0.0
255.255.255.255
IP address of Host
N/w id
8
8
8
8
190.20.100.10/22
N/w portion
Id
N/w id
Masking (AND)
Default Mask
(1)
32-22
10
S1
S3
S2
S4
Subnets
4
reserved → vary
190.20.01100100.00001010
22
Host part
10 bits
Host

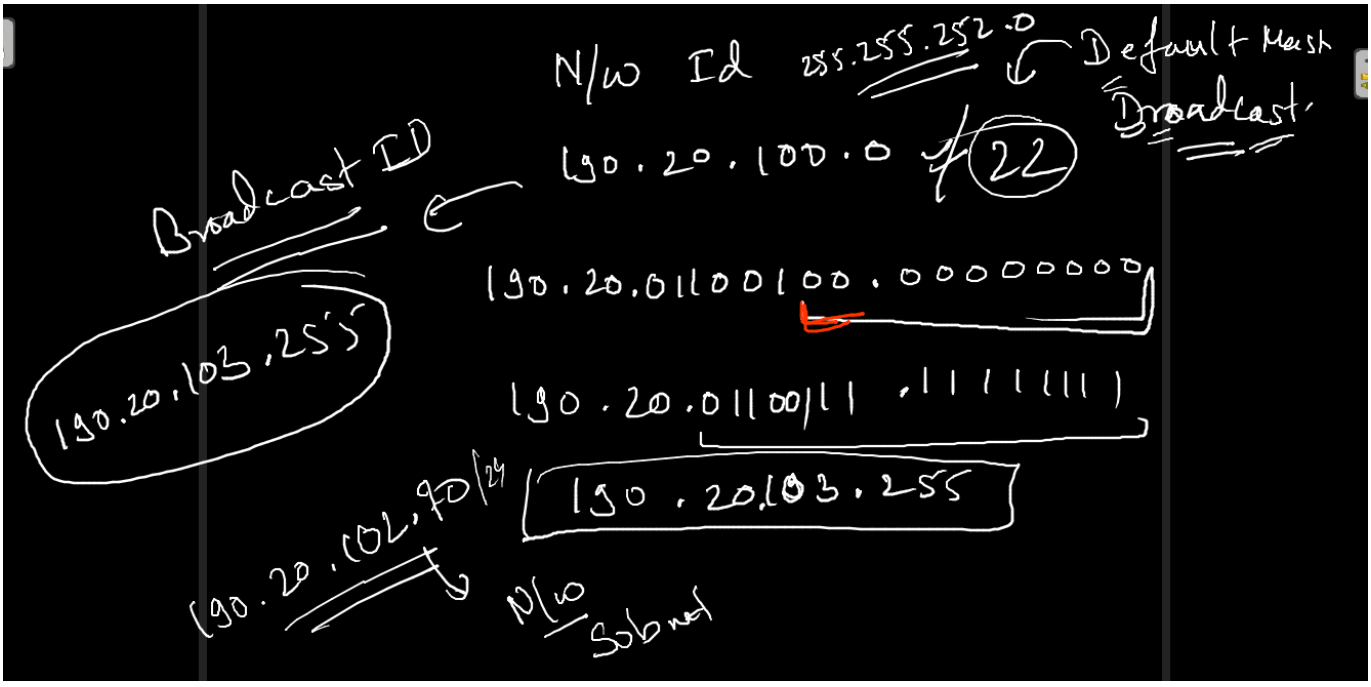

Class A = 255.0.0.0
B = 255.255.0.0
190. 20. 01100100. 00001010
11111111. 11111111. 11111100. 00000000
255 . 255 . 252 . 0
Default Mask
128
64
32
16
8
4
1 = 1
1 - 0 = 0
190.20.100.0
190.20.100.0
Subnet
150.20.100.10
IP

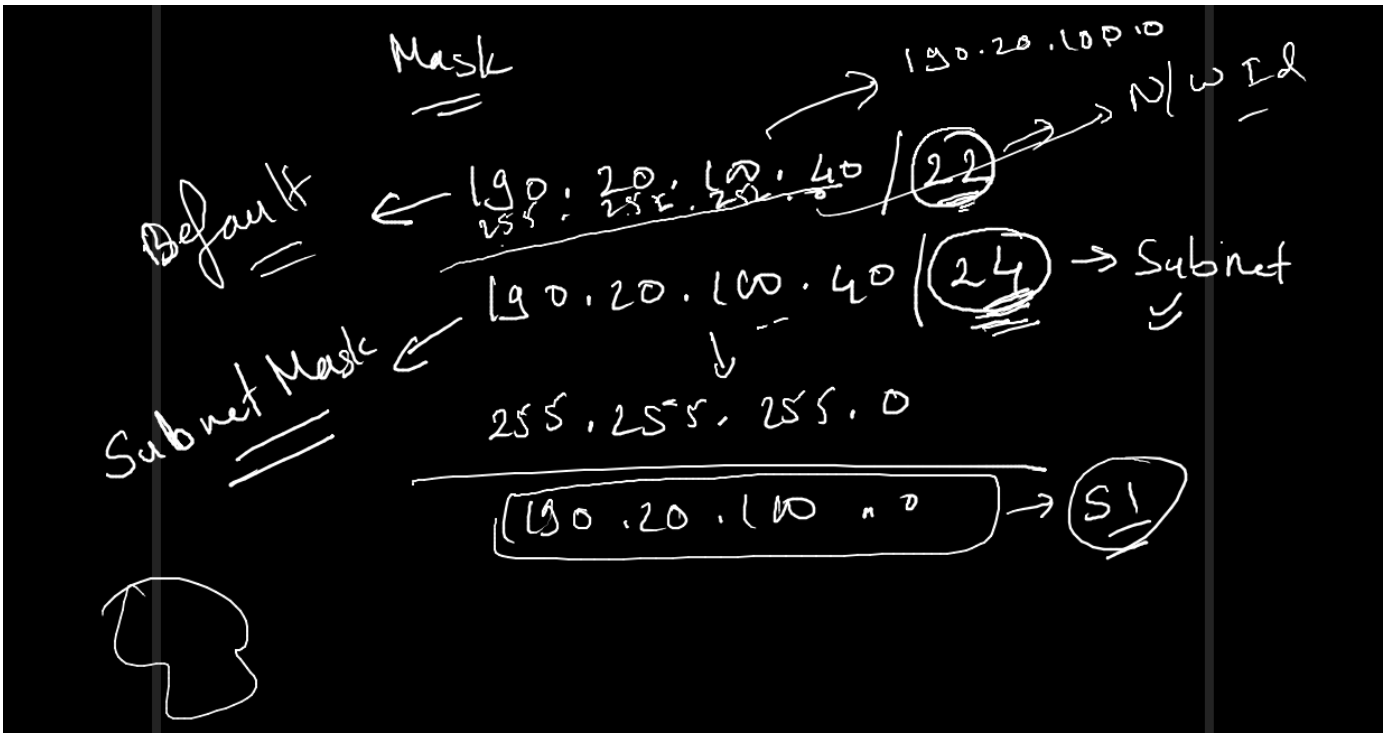

32-bit
192.20.100.0/22
192.20. ------ ---. - - - - - - - -
MSB → 00. 00000000
Host
Subnet bit
Sub
11. 11111111
4
00
01
10
11
3
192.20.01100100.00000000/24
192.20.01100100.11111111
S1
192.20.100.0
192.20.100.255

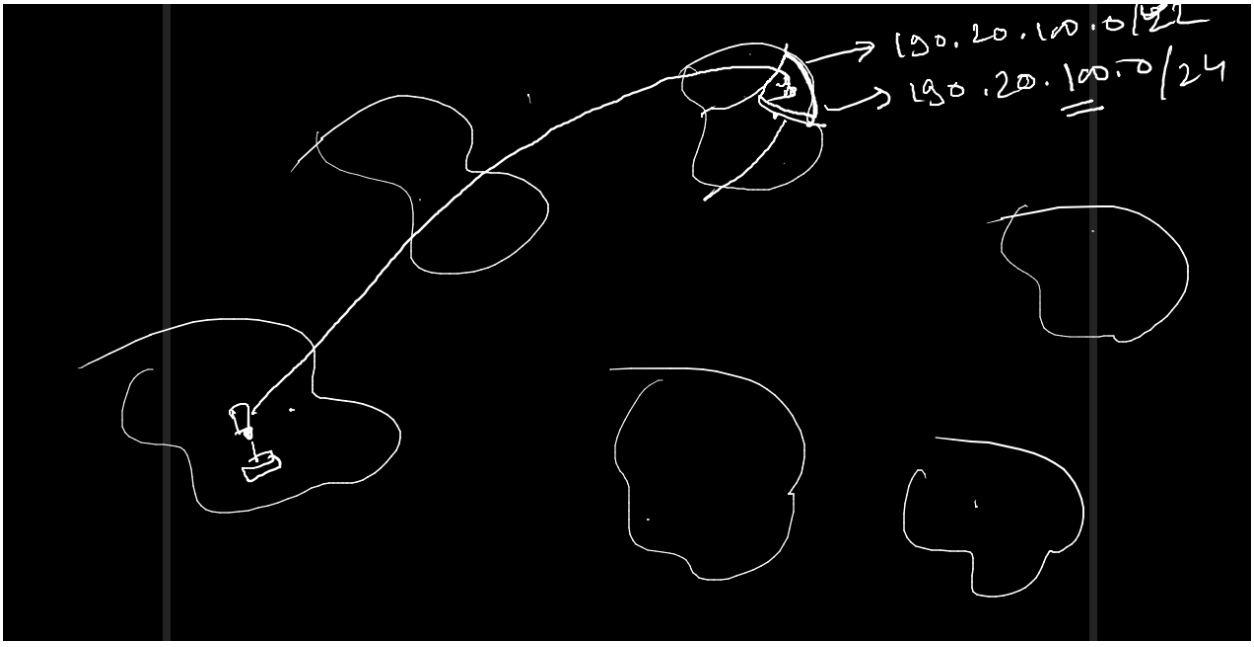

192.20.101.0
192.20.101.255
192.20.01100101.00000000
192.20.01100101.11111111
S2
192.20.102.0
192.20.01100110.00000000
192.20.102.255
192.20.01100110.11111111
S3
192.20.01100111.00000000
192.20.01100111.11111111
192.20.103.0
192.20.103.255


SIP
DIP
Data
N/w id
Subnet id
254
256
S1
S2
S3
S4
192.20.100.0
192.20.100.255
192.20.102.0
192.20.102.255
192.20.101.0
192.20.101.255
192.20.103.0
192.20.103.255
.0
.255
broadcast
256-2
2xn=
2x4=8
192.20.101.255
SIP
DIP


190.20.101.40
SIP
DIP
S2
Broadcast
190.20.101.255


N/w Id 255.255.252.0
Default Mask
Broadcast.
190.20.100.0 /22
Broadcast ID
190.20.103.255
190.20.01100100.00000000
190.20.01100111.11111111
190.20.103.255
190.20.102.90/24
N/w
Subnet


Mask
Default
190.20.100.40/22
190.20.100.0
N/w Id
Subnet
Subnet Mask
190.20.100.40/24
255.255.255.0
190.20.100.0
S1


190.20.100.0/22
190.20.100.0/24

VLSM : Variable length Subnet Masking.

IP = 190.20.10.30/16 → N/w ID or IP of host

7 Subnets

S1 = 2000 hosts
S2 = 1000 hosts
S3 = 500 hosts
S4 = 200 hosts
S5 = 125 hosts
S6 = 60 hosts
S7 = 50 hosts

Subnet ID
Subnet Mask

VLSM Default Mask

190.20.10.30/16
255.255.0.0
190.20.0.0/16 → N/w Id.

Default Mask

2000

$2^{10} = 1024_{10}$

2048

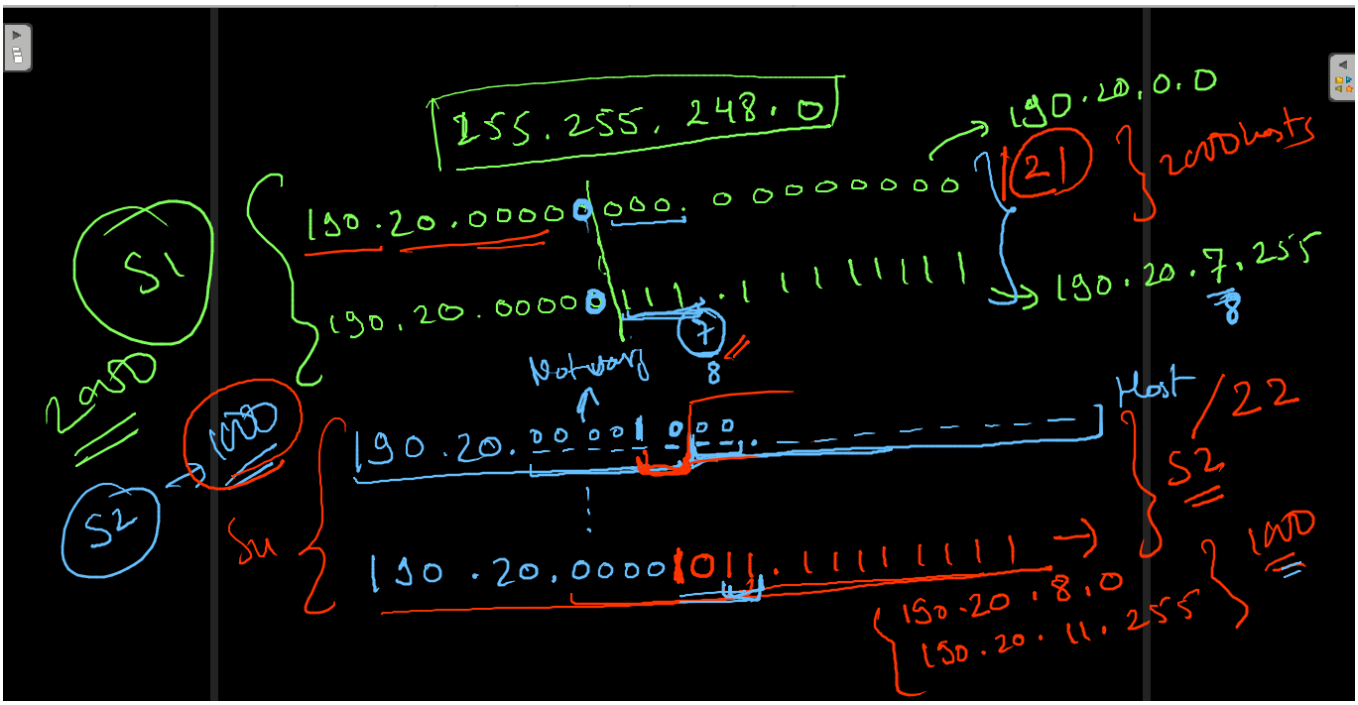
fix

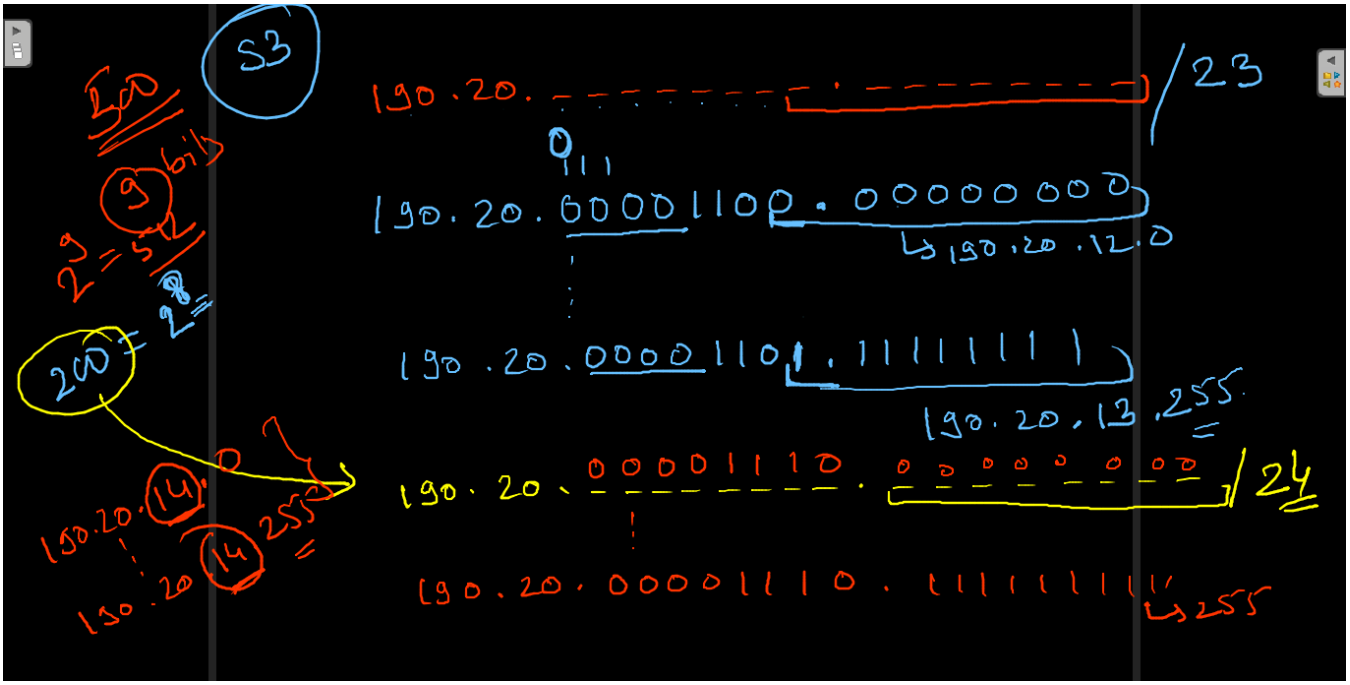
N/w | Host

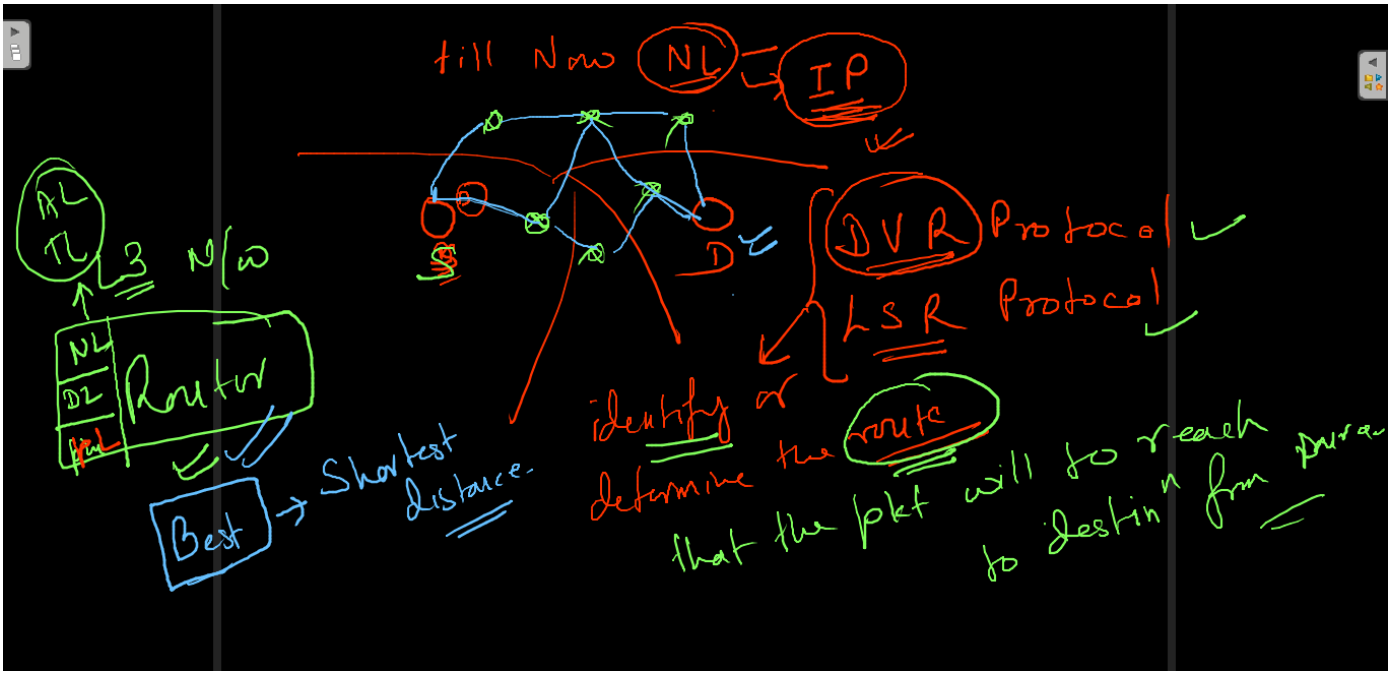
7 6 5 4 3 2 1 0
1 1 1 1 1

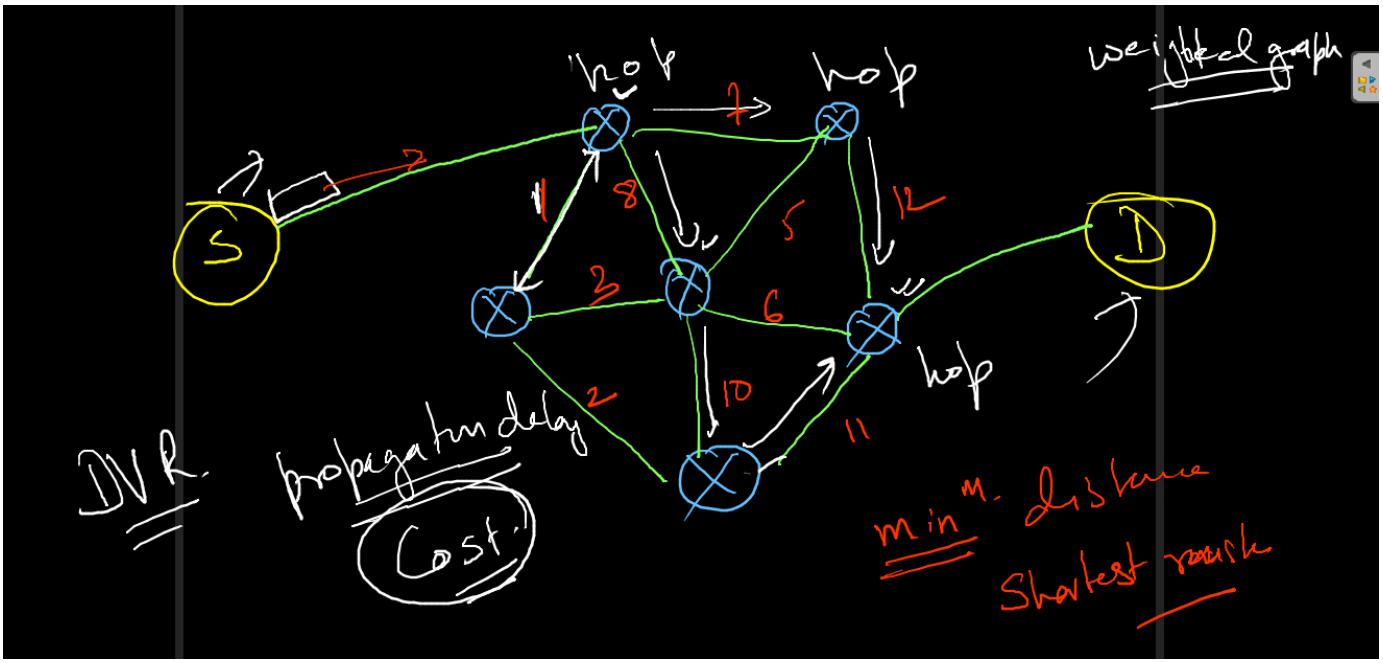
190.20.00000000.00000000
190.20.00000 | 111.11111111

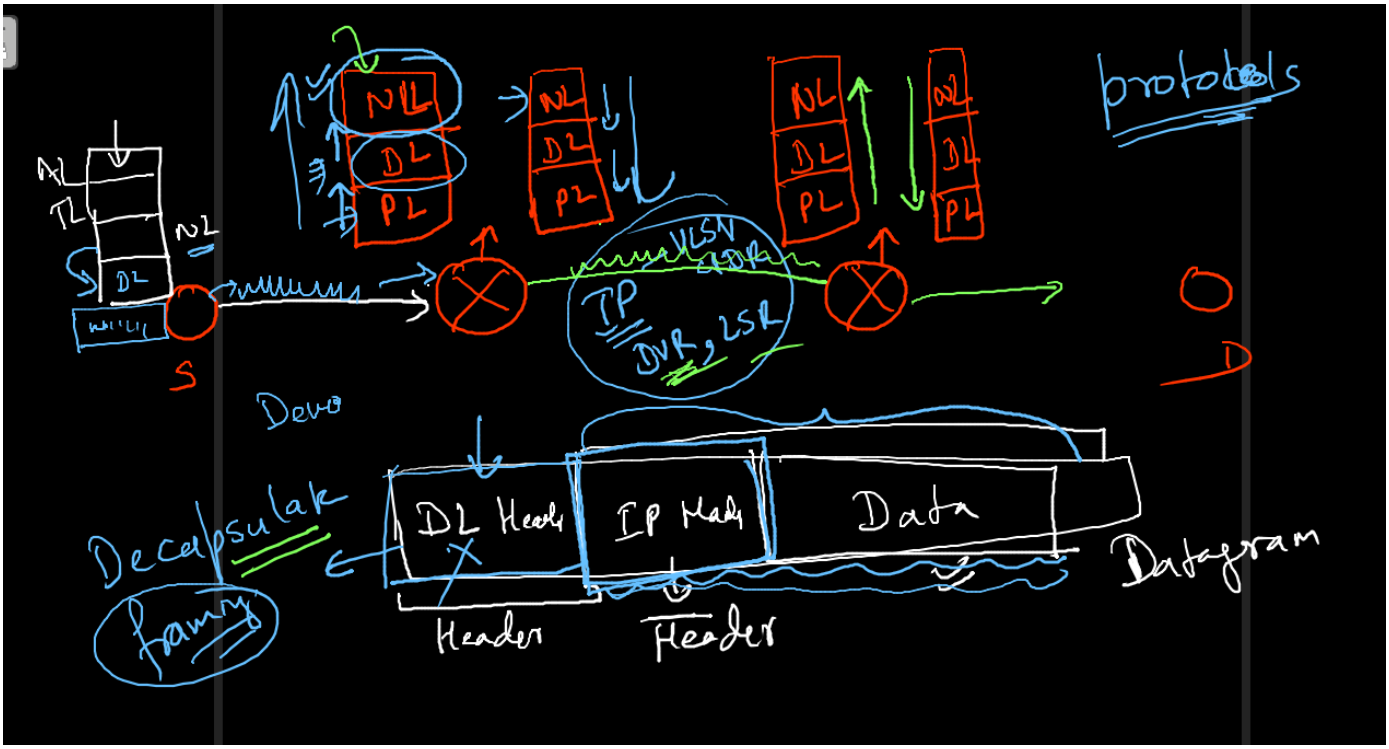
S1 = 2000


255.255.248.0
190.20.0.0
2046 hosts
S1
2048
190.20.0000 0000.0000 0000
190.20.0000 0111.1111 1111
190.20.7.255
Network
Host
/22
S2
1000
190.20.0000 1000.
190.20.0000 1011.1111 1111
190.20.8.0
190.20.11.255
1000


S3
500
9 bits
2^9 = 512
2^8 =
200
190.20.
/23
190.20.0000 1100.0000 0000
190.20.12.0
190.20.0000 1101.1111 1111
190.20.13.255
190.20.0000 1110.0000 0000
/24
190.20.0000 1110.1111 1111
255
190.20.14.0
190.20.14.255


till Now
NL
IP
AL
TL
3 N/w
NL
DL
PHL
Router
S
D
DVR Protocol
LSR Protocol
identify or
determine the route
that the pkt will go to reach
to destin" from sour
Best → Shortest distance.

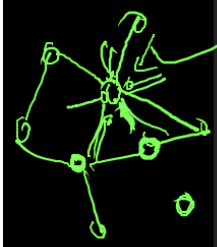

hop
hop
weighted graph
S
D
7
7
11
8
5
12
3
6
10
2
11
hop
DVR.
propagation delay
Cost
min" distance
Shortest route



DVR → Distance Vector Routing protocol.

route → shortest route.

(1) Every router maintains a Routing Table



(2)

Destination Node	Distance	Next Node

(3) Every router knows, how many total routers are there in the N/w

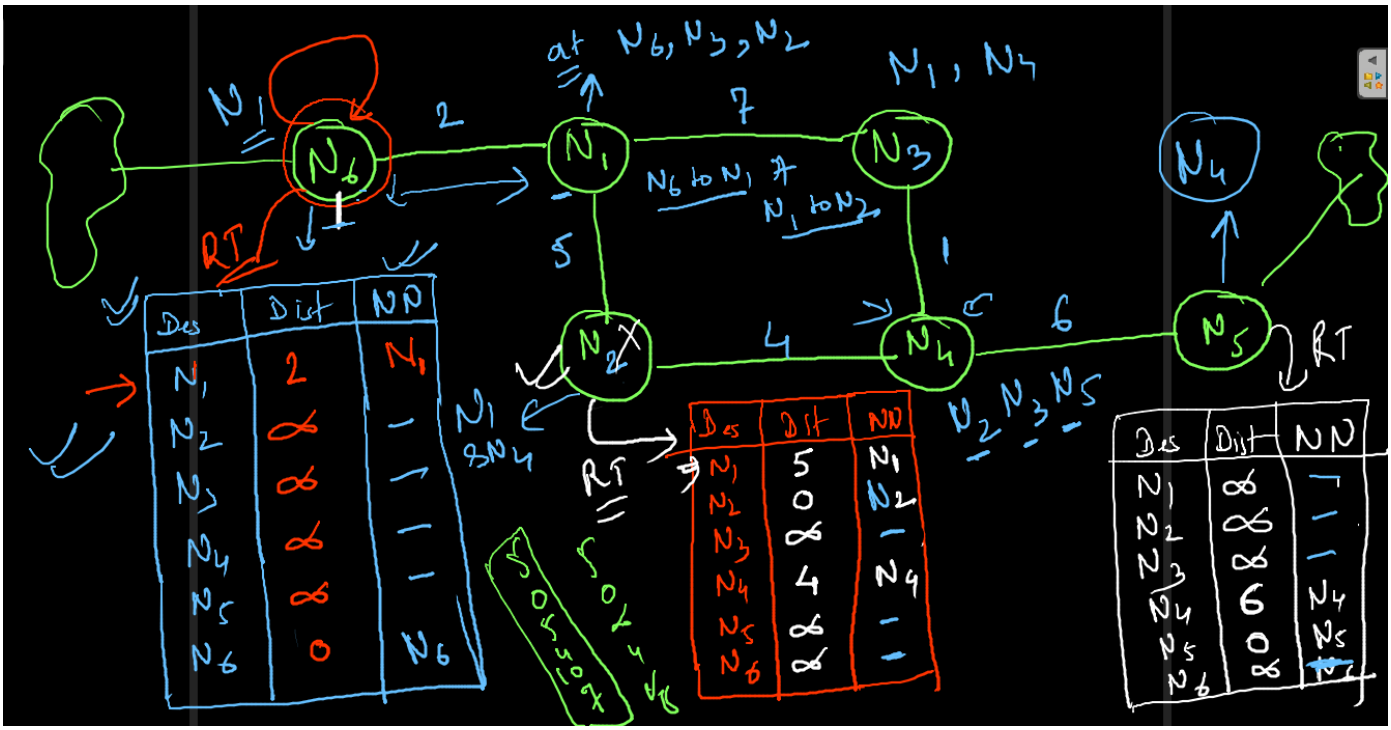
(4) Every router sends "hello" message to its neighbour router

two steps

(1) Every router maintain its local RT

(2) Every router shares infrm^n to its neighbour router.

→ RT → empty



RT N_1

	DV	
N_1	0	N_1
N_2	5	N_2
N_3	7	N_3
N_4	∞	—
N_5	∞	—
N_6	2	2

RT N_3

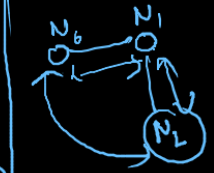
	DV	
N_1	7	N_1
N_2	∞	—
N_3	0	N_3
N_4	1	N_4
N_5	∞	—
N_6	2	—

RT N_4

	DV	
N_1	∞	—
N_2	4	N_2
N_3	1	N_3
N_4	0	N_4
N_5	6	—
N_6	∞	—

(2) $N_6 \rightarrow N_4 \Rightarrow$

	Di
N_1	0
N_2	5 = 7
N_3	7
→ N_4	∞
→ N_5	∞
N_6	2



N_6

D	DB	N
N_1	2	N_1
→ N_2	7 = 9	N_1, N_2
N_3	9 =	N_1, N_3
N_4	∞	—
N_5	∞	—
N_6	0	N_6 =

$N_6 \rightarrow N_1 \Rightarrow$ N_6 to N_1 & N_1 to N_1

$= 2 + 0$

$= (2)$

Di

2
∞
(∞)
∞
∞
0

N_6 to N_2 = N_6 to N_1 + N_1 to N_2

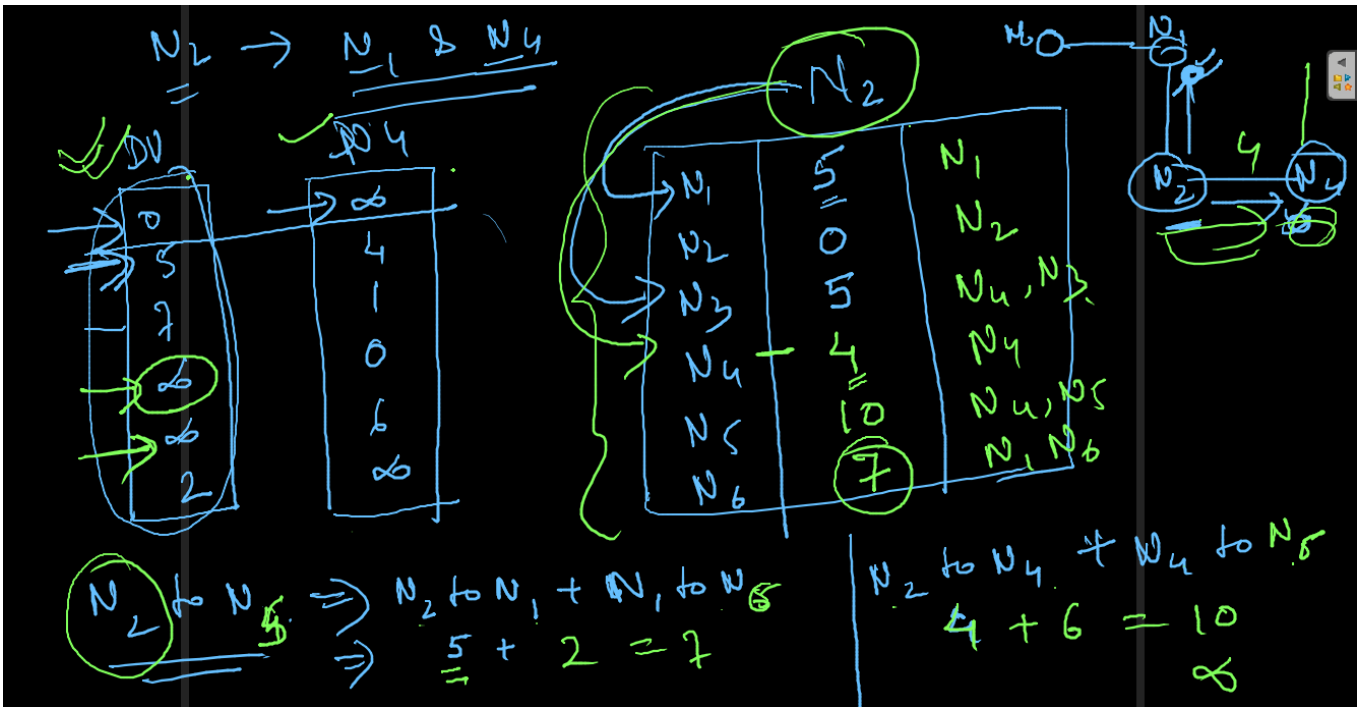
$2 + 5 = 7$

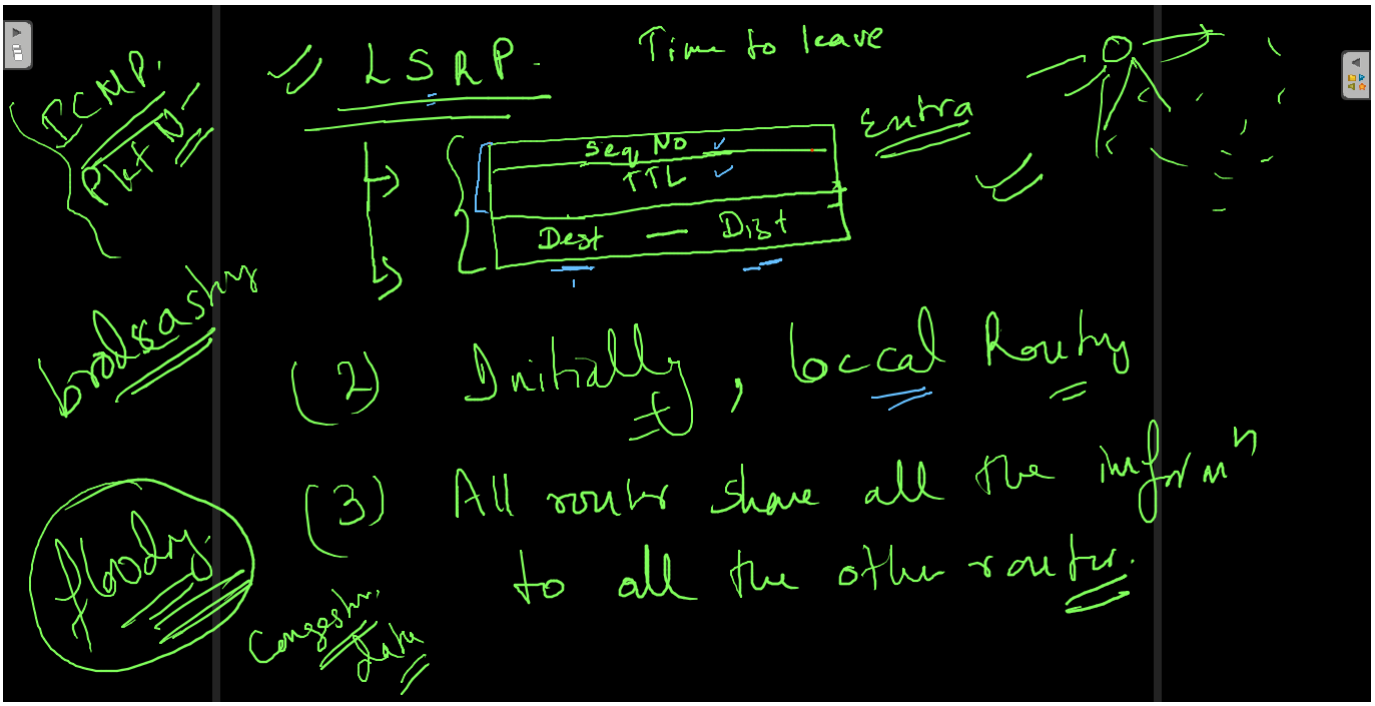
N_6 to N_3 = N_6 to N_1 + N_1 to N_3

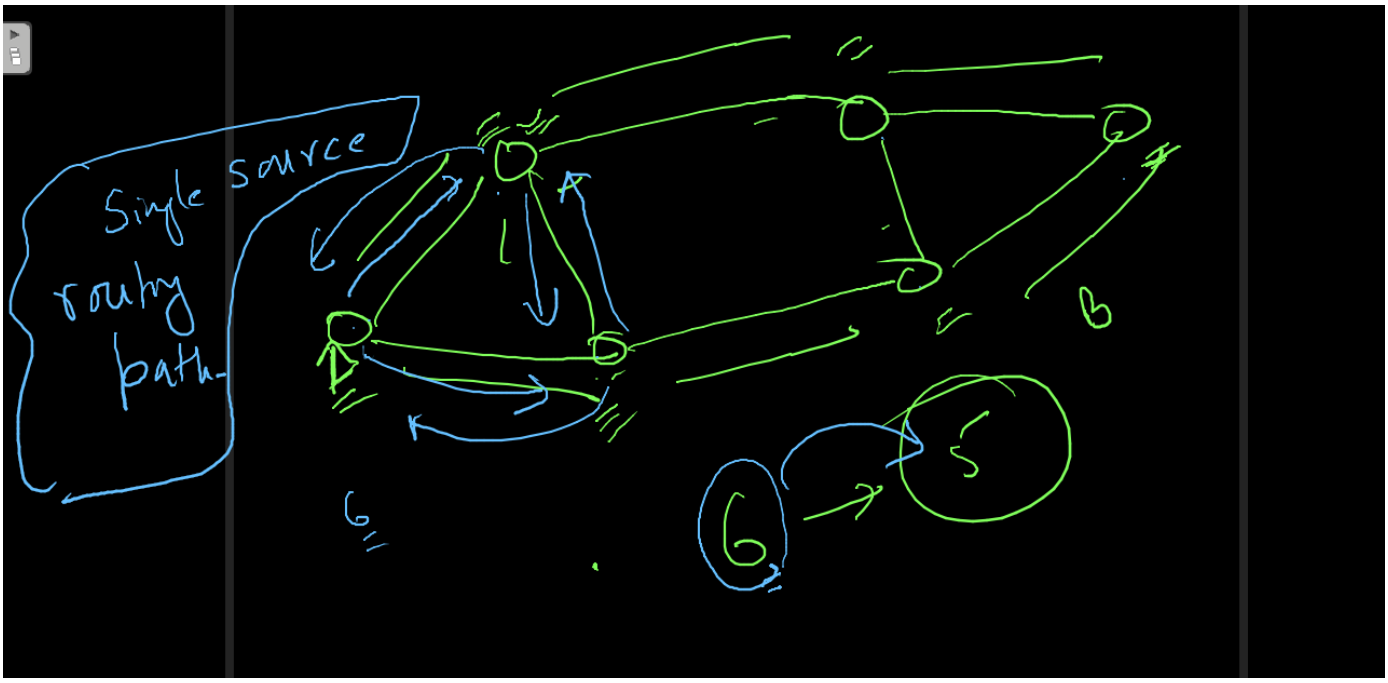
$= 2 + 7 = 9$

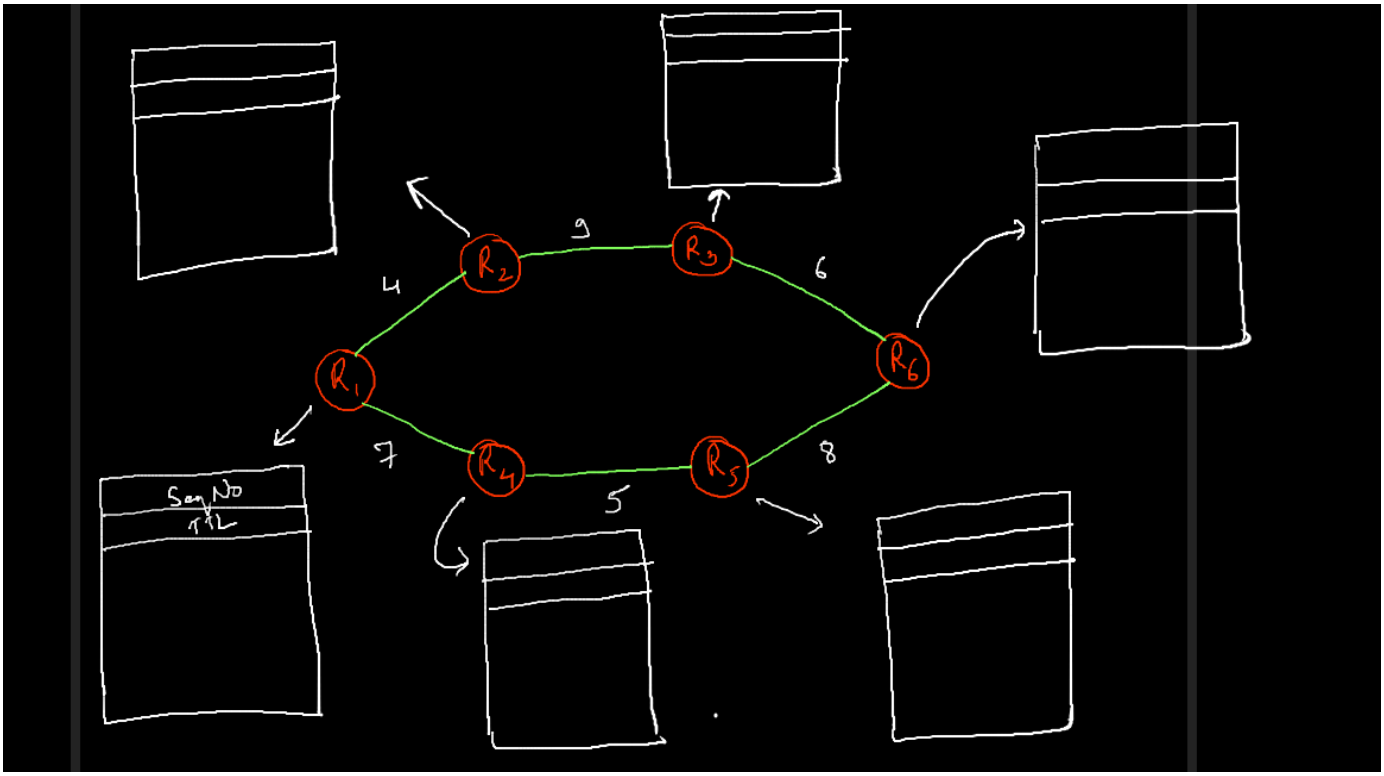
N_6 to N_4 = N_6 to N_1 + N_1 to N_4

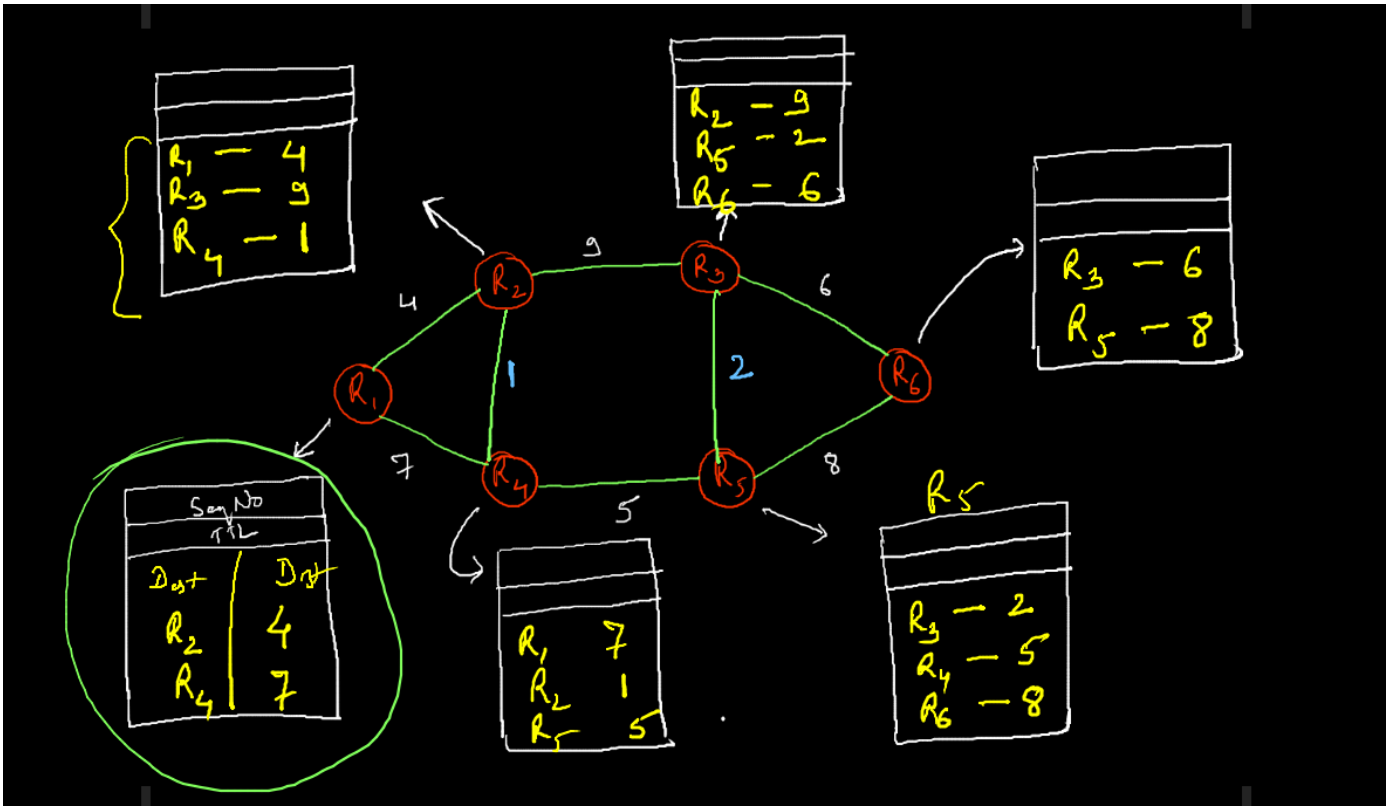
$= 2 + \infty = \infty$

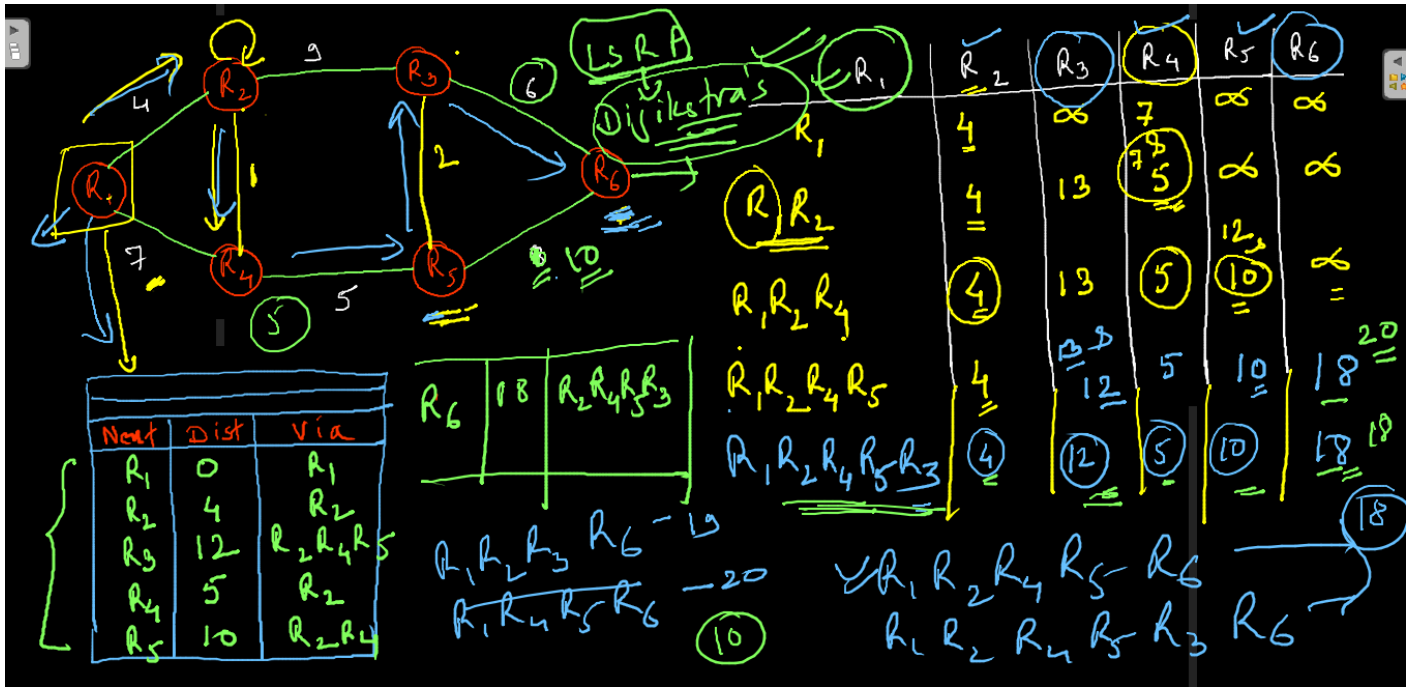


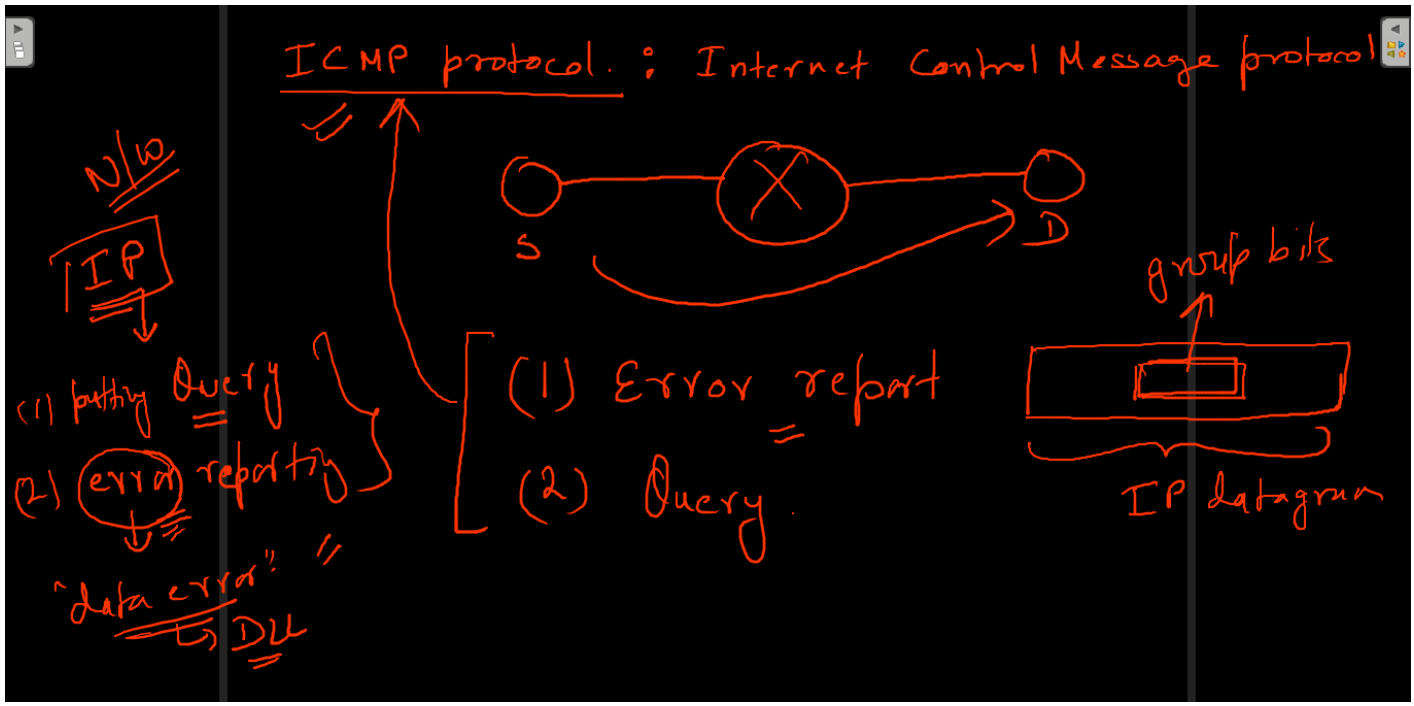

ICMP
Ping
LSRP
Time to leave
Seq No
TTL
Dest — Dist
Extra
broadcasting
(2) Initially, local Routing
(3) All router share all the information to all the other routers.
flooding
Congestion Table

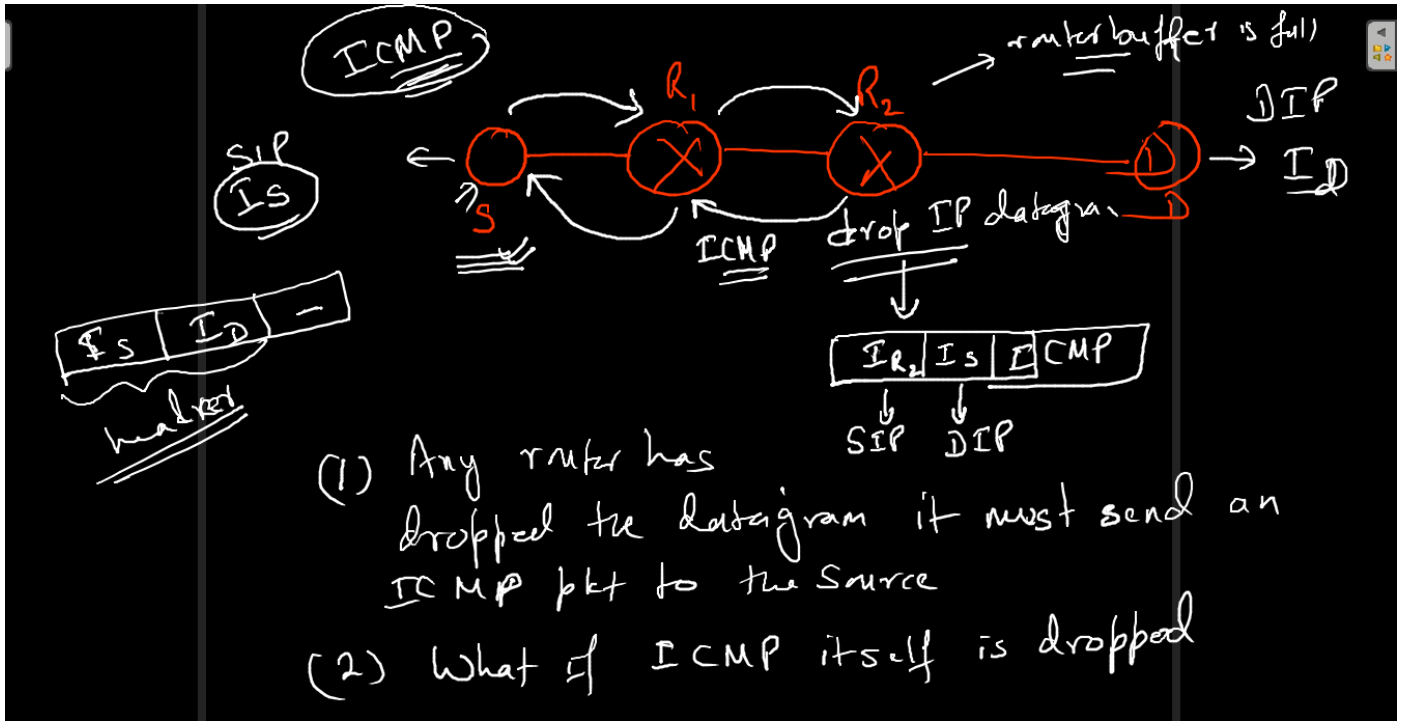

Single source routing path
A
B
K
6
5
6


R1
R2
R3
R4
R5
R6
9
4
6
7
8
5
Seq No
TTL


R1 — 4
R3 — 9
R4 — 1
R2 — 9
R5 — 2
R6 — 6
R3 — 6
R5 — 8
R1
R2
R3
R4
R5
R6
9
4
6
1
2
7
8
5
Seq No
TTL
Dest
Dist
R2 4
R4 7
R1 7
R2 1
R5 5
R5
R3 — 2
R4 — 5
R6 — 8

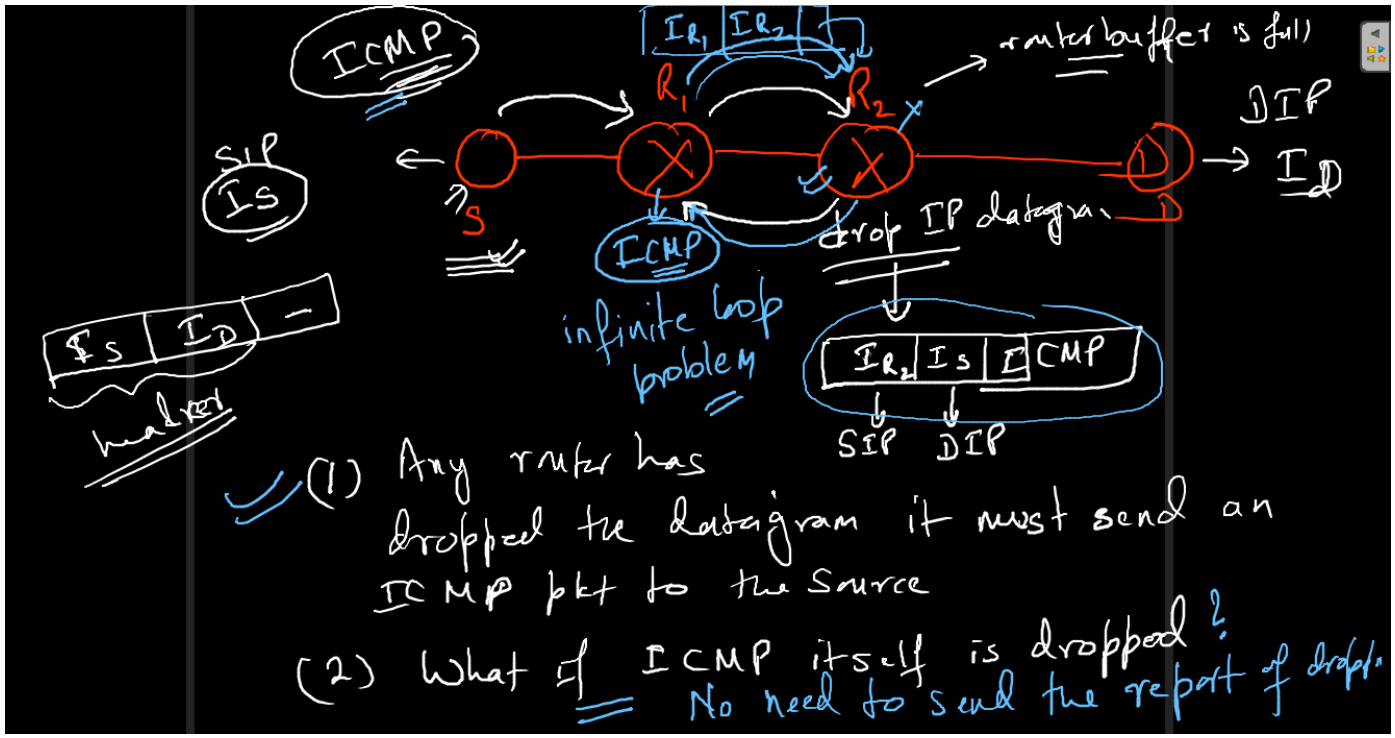

LS R A
Dijkstra's
R1
R2
R3
R4
R5
R6
4
9
6
2
1
7
5
10
Next | Dist | Via
R1 | 0 | R1
R2 | 4 | R2
R3 | 12 | R2R4R5
R4 | 5 | R2
R5 | 10 | R2R4
R6 | 18 | R2R4R5R3
R1R2R3R6 — 20
R1R4R5R6 — 10
R1R2R4R5R6
R1R2R4R5R3R6
18
20


ICMP protocol : Internet Control Message protocol
N/W
IP
(1) putting Query
(2) error reporting
"data error"
DLL
(1) Error report
(2) Query
S
D
group bits
IP datagram



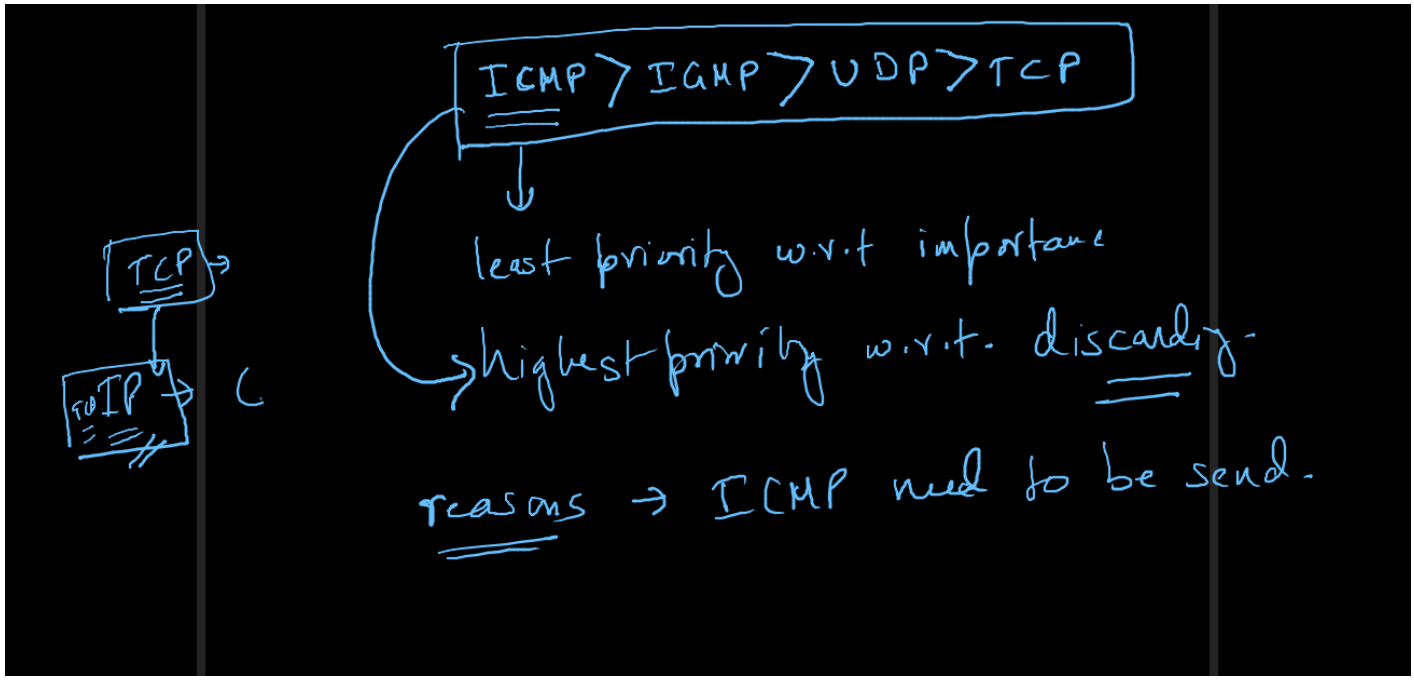
(1) Any router has dropped the datagram it must send an ICMP pkt to the Source

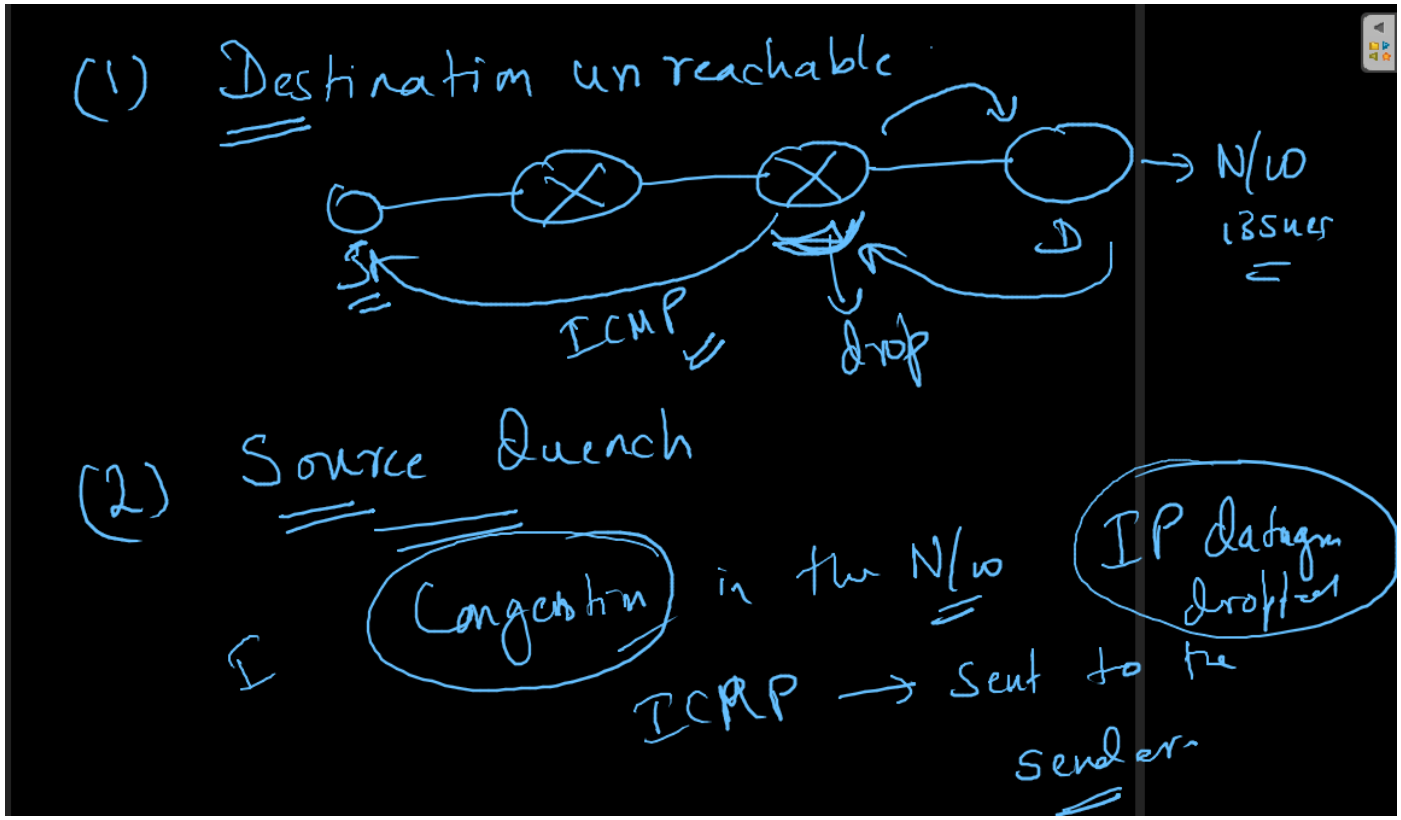
(2) What if ICMP itself is dropped



(1) Any router has dropped the datagram it must send an ICMP pkt to the Source

(2) What if ICMP itself is dropped ?
No need to send the report of drop!

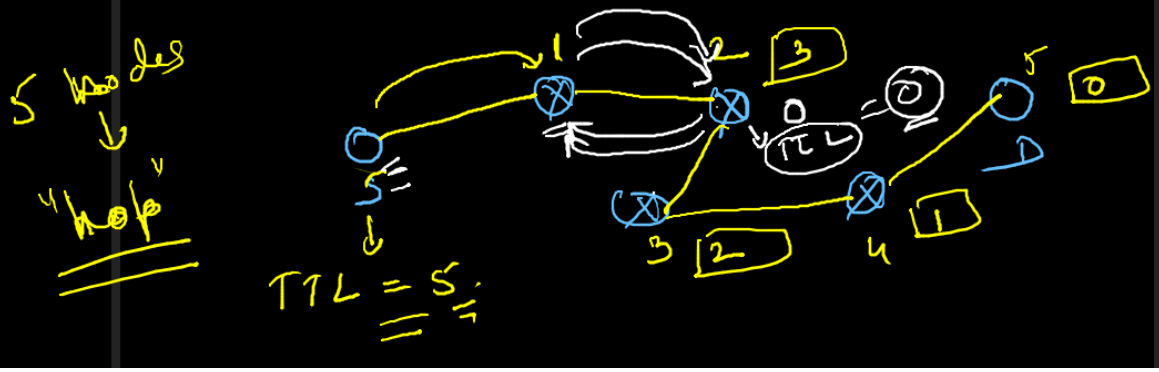

ICMP > IGMP > UDP > TCP
TCP
IP
least priority w.r.t importance
highest priority w.r.t. discarding
reasons → ICMP need to be send.

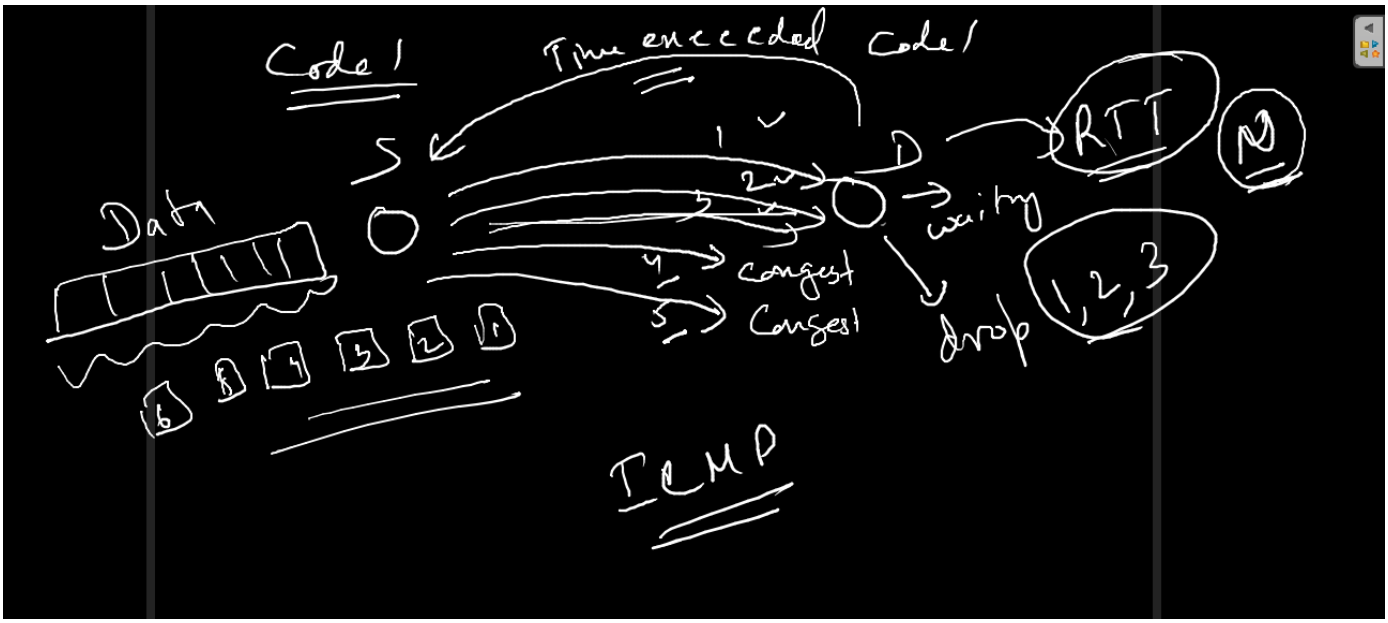

(1) Destination unreachable
S
D
N/W issues
ICMP
drop
(2) Source Quench
Congestion in the N/W
IP datagram dropped
ICMP → Sent to the sender.

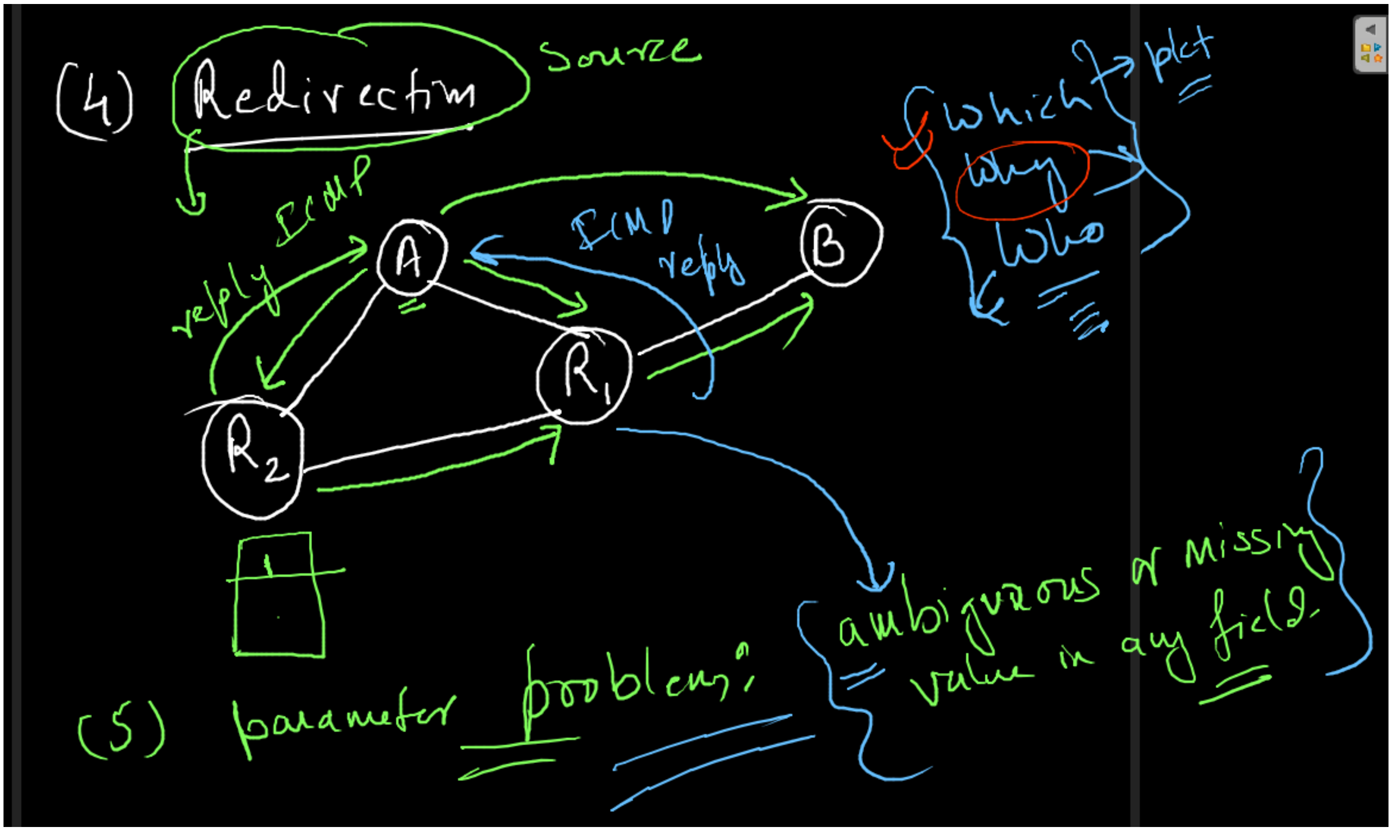
(3) Time Exceeded

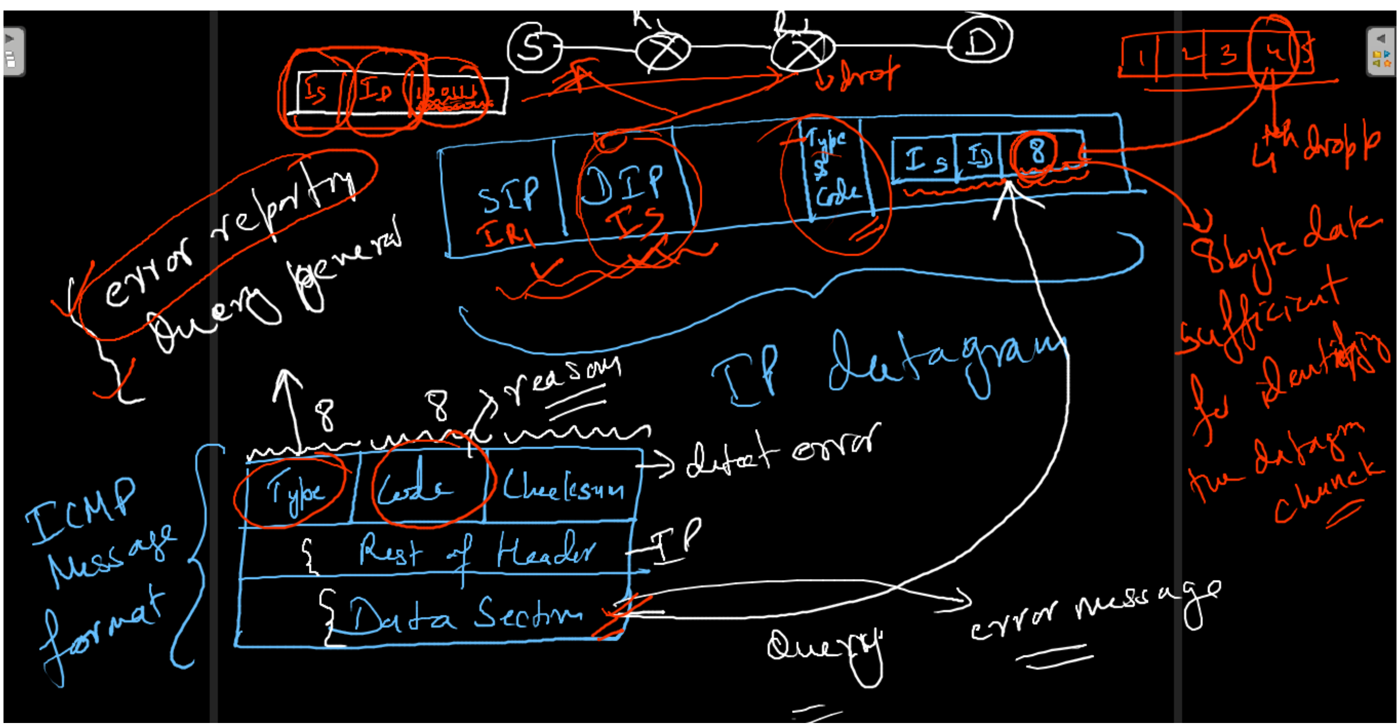
- Code '0' – TTL (Time To Leave)
- Code '1'

TTL → ↓

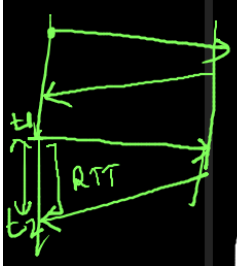





(4) Redirection
Source
ICMP
reply
A
B
R1
R2
ICMP reply
which
why
who
pkt
(5) parameter problem:
ambiguous or missing value in any field.


S
D
drop
SIP IR1
DIP IS
Type & Code
IS
ID
8
4th drop
8 byte data sufficient for identifying the datagram chunk
error reporting
Query general
8
8
reason
IP datagram
ICMP Message format
Type
Code
Checksum
detect error
Rest of Header
IP
Data Section
error message
Query

ICMP IP

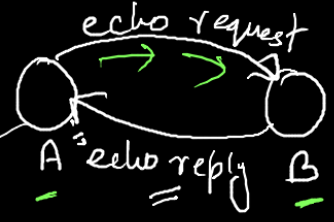


↳ Error messg
=
↳ Query Message

Echo request & Reply

Timestamp Request & Reply.

↳ Round-Trip-Time.



determining

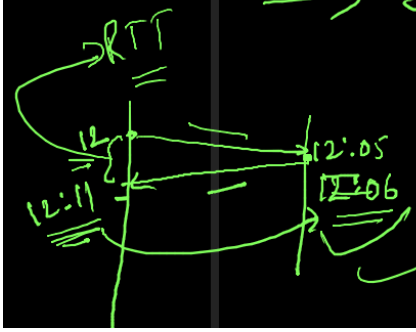
diagnose

RTT (Duration b^n Send & Receive.

→ Send Time : Received - Original

Receiving time : Returned - Transmit

Round Trip Time : Send time + RT



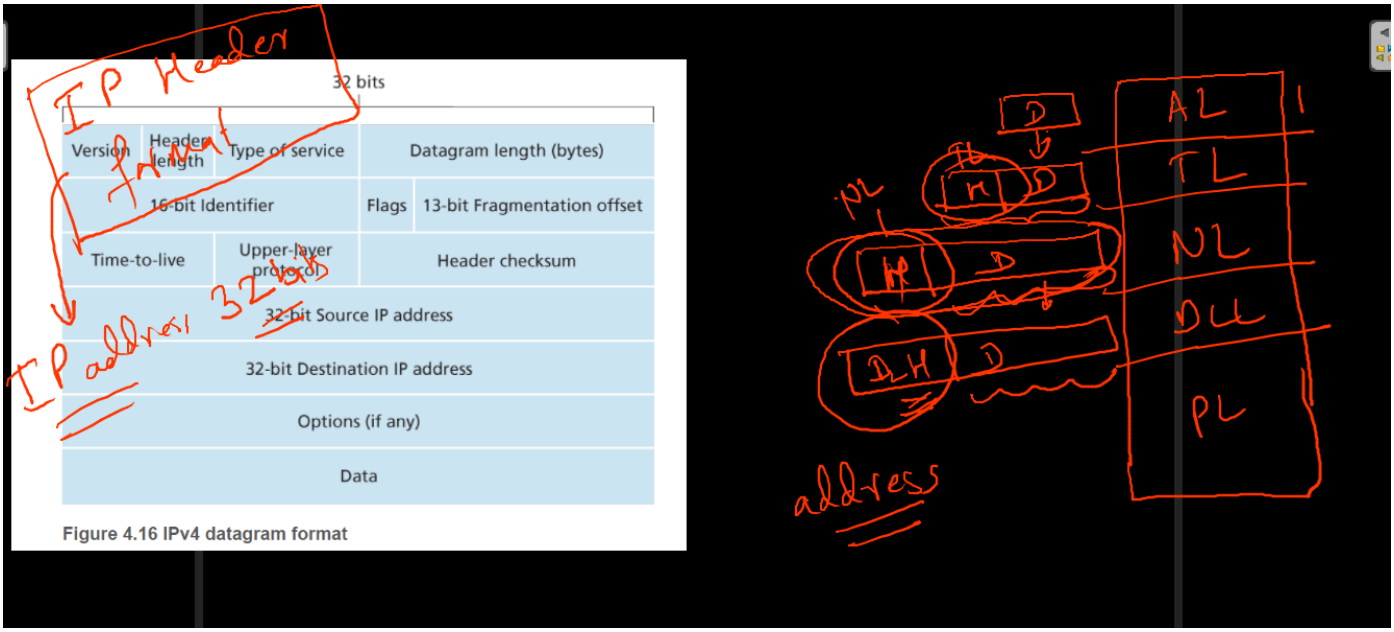


Figure 4.16 IPv4 datagram format

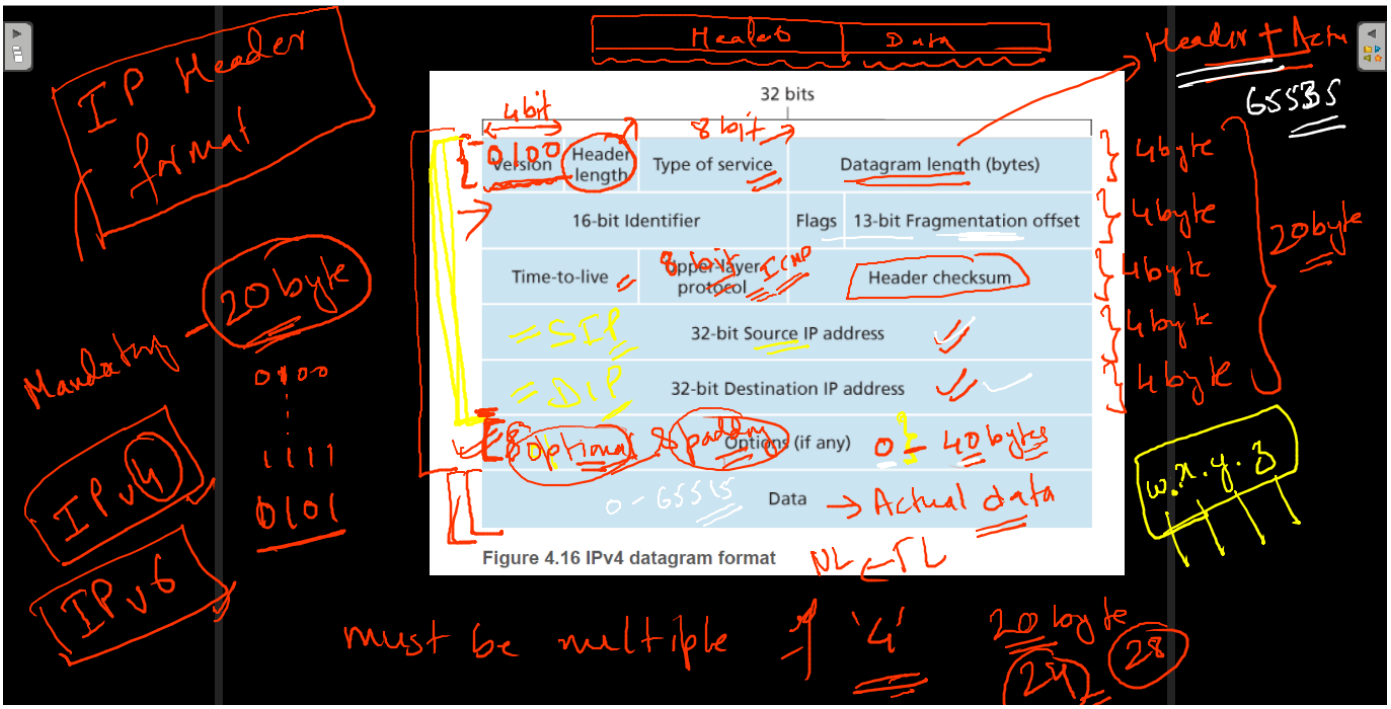
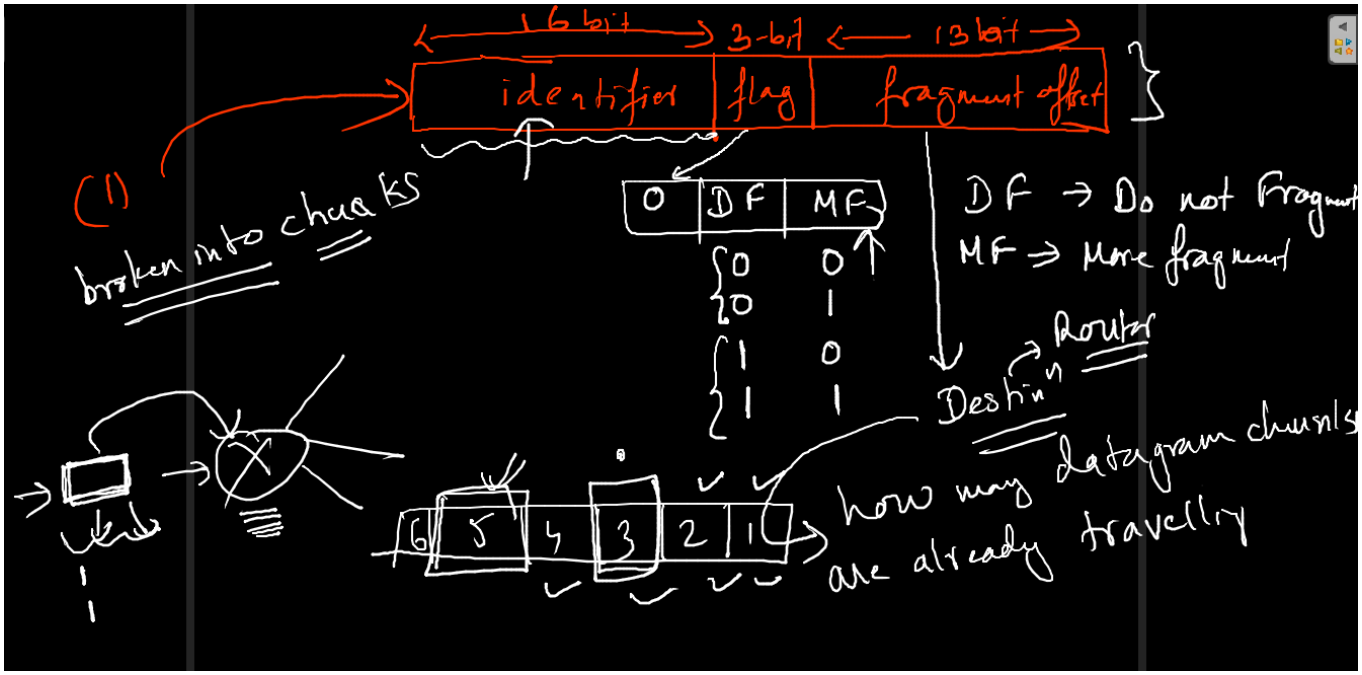
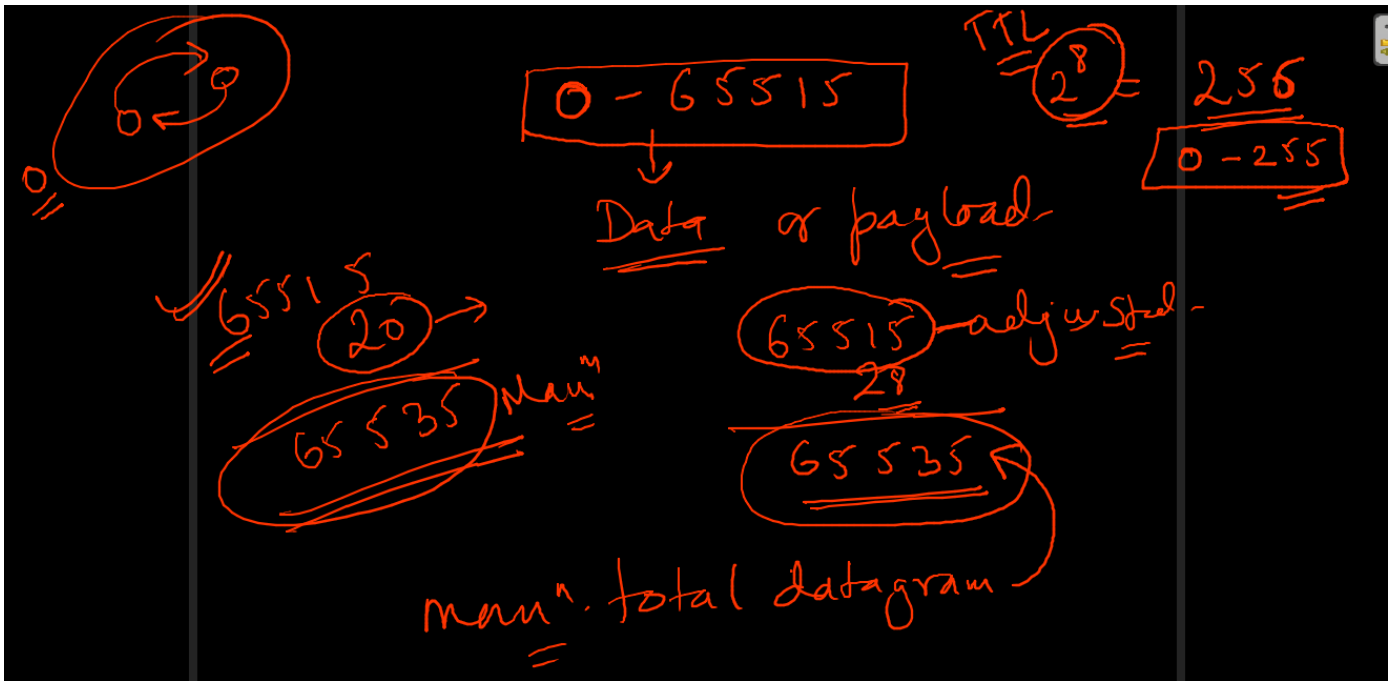
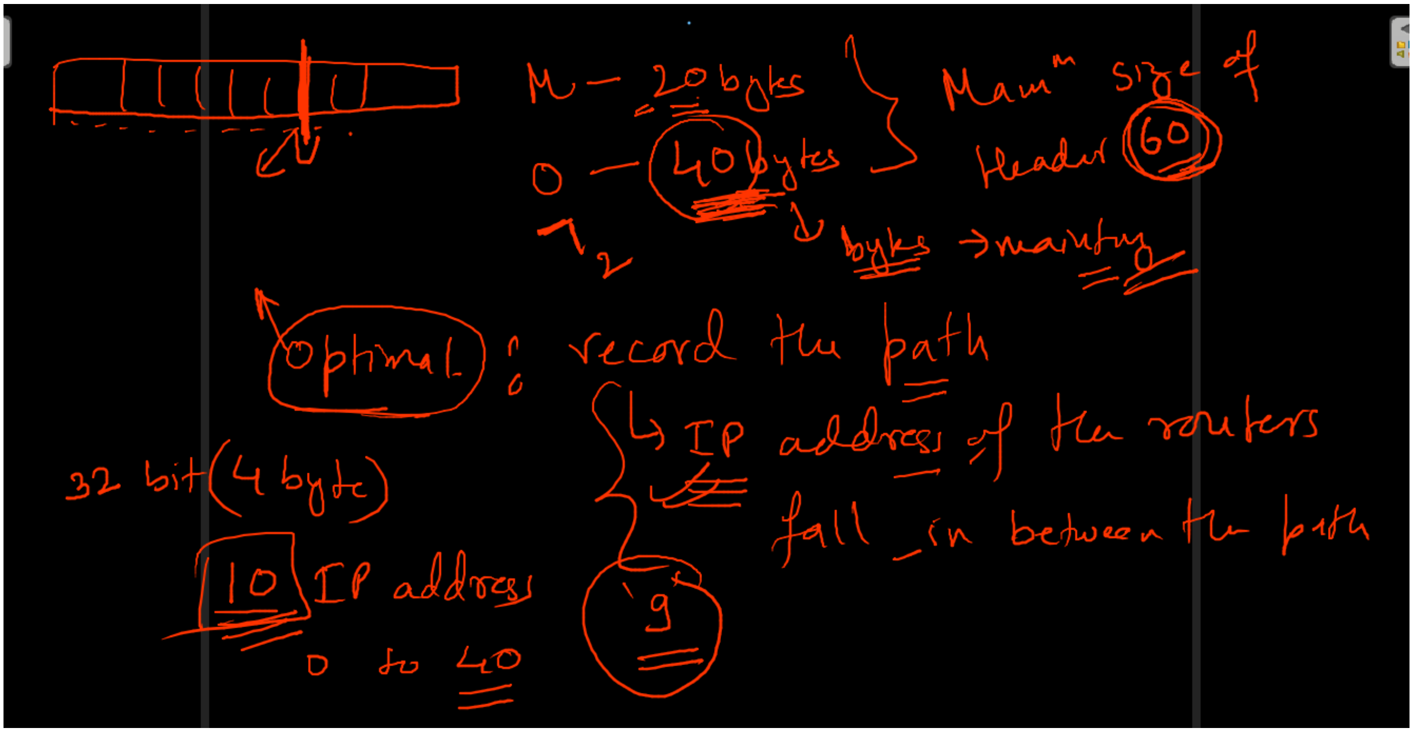


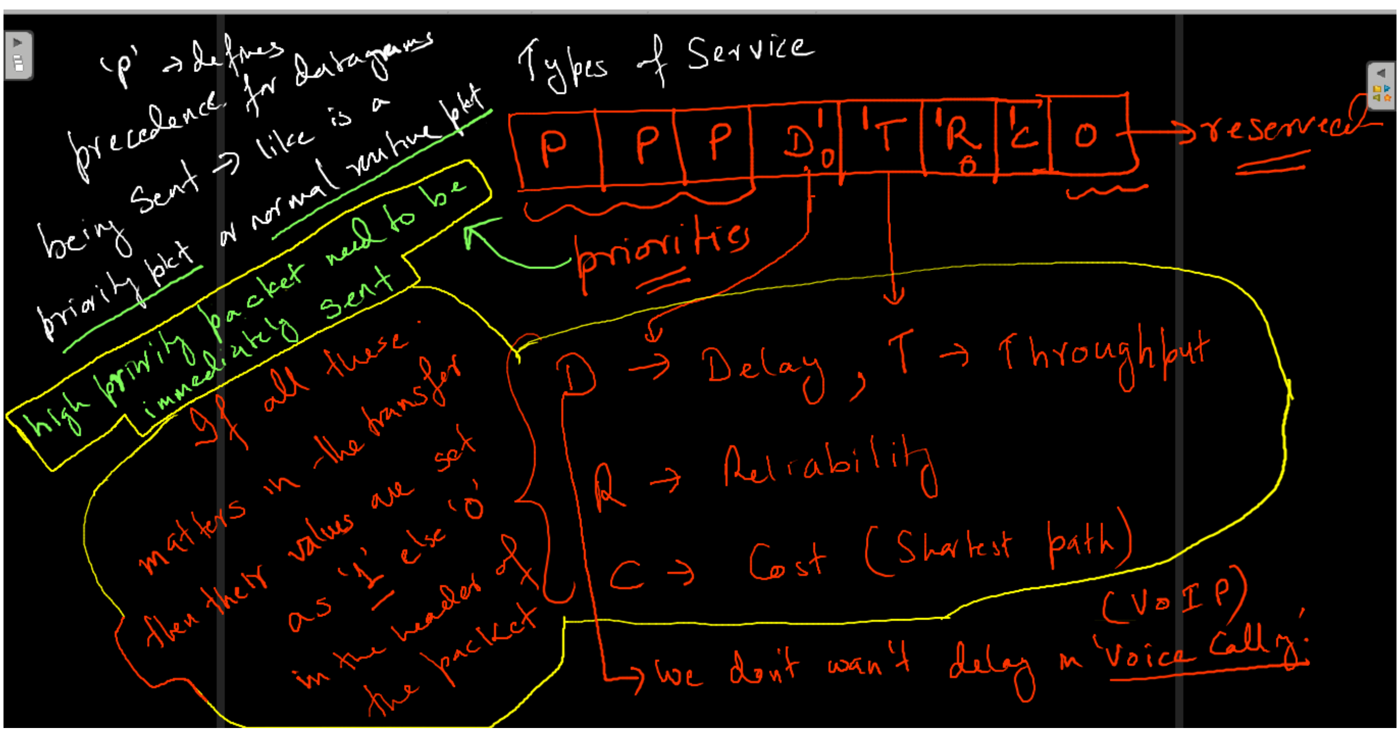
Figure 4.16 IPv4 datagram format


16 bit
3-bit
13 bit
identifier
flag
fragment offset
(1)
broken into chunks
0 DF MF
0 0
0 1
1 0
1 1
DF → Do not Fragment
MF → More fragment
Router
Destin
how many datagram chunks are already travelling

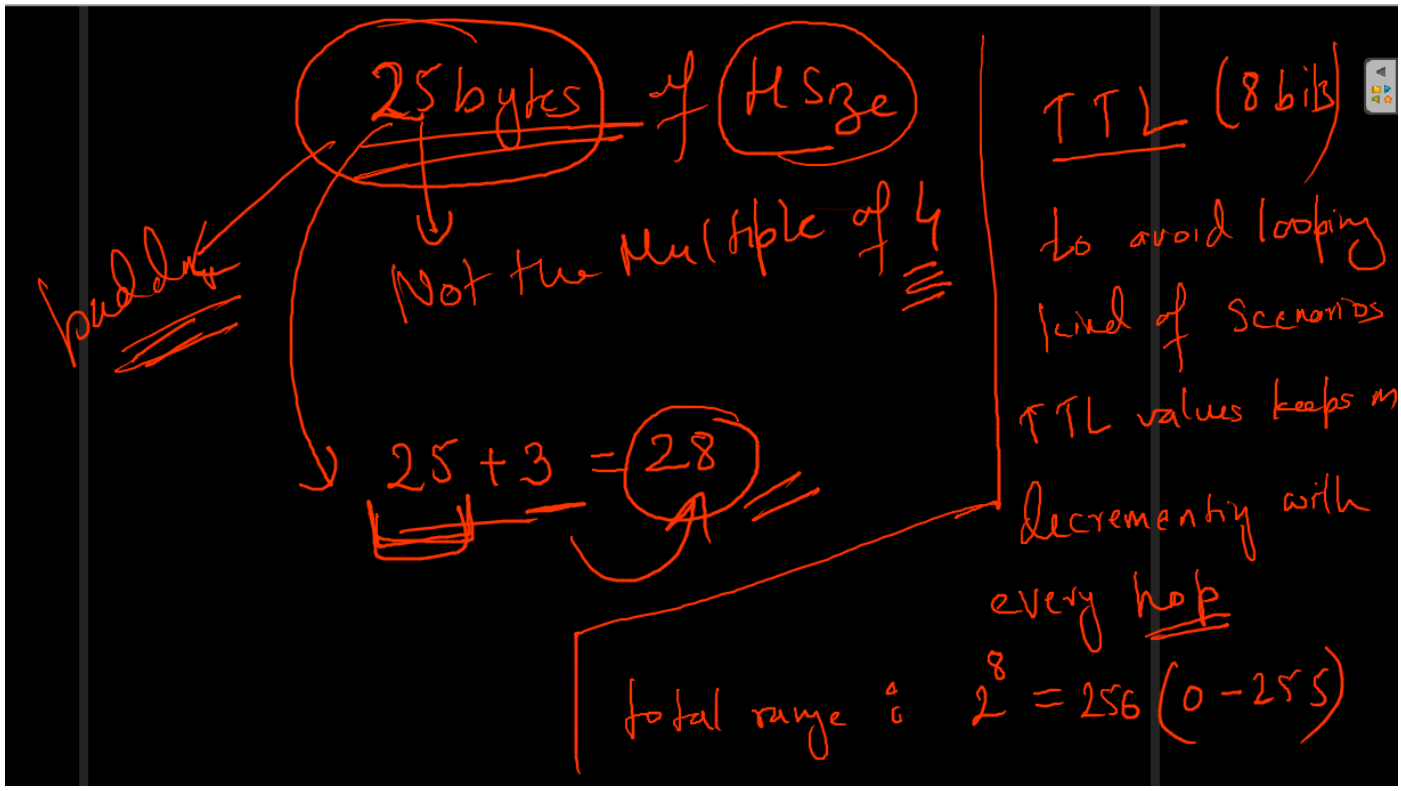

0
0-65515
Data or payload
TTL
2^8
256
0-255
65515
20
65535
Max^n
65515
adjusted
28
65535
max^n. total datagram



M – 20 bytes
O – 40 bytes
Maxm size of Header 60
bytes → maintaining
Optional : record the path
↳ IP address of the routers
fall in between the path
32 bit (4 byte)
10 IP address
0 to 40
'9'



'p' → defines precedence for datagrams being sent → like is a priority pkt or normal routine pkt
high priority packet need to be immediately sent
Types of Service
P P P D T R C 0
→ reserved
priorities
D → Delay , T → Throughput
R → Reliability
C → Cost (Shortest path)
If all these matters in the transfer then their values are set as '1' else '0' in the header of the packet
↳ we don't want delay in 'Voice Calling'
(VoIP)



- **Version number.** These 4 bits specify the IP protocol version of the datagram. By looking at the version number, the router can determine how to interpret the remainder of the IP datagram. Different versions of IP use different datagram formats. The datagram format for IPv4 is shown in Figure 4.16. The datagram format for the new version of IP (IPv6) is discussed in Section 4.3.5.
- **Header length.** Because an IPv4 datagram can contain a variable number of options (which are included in the IPv4 datagram header), these 4 bits are needed to determine where in the IP datagram the payload (e.g., the transport-layer segment being encapsulated in this datagram) actually begins. Most IP datagrams do not contain options, so the typical IP datagram has a 20-byte header.
- **Type of service.** The type of service (TOS) bits were included in the IPv4 header to allow different types of IP datagrams to be distinguished from each other. For example, it might be useful to distinguish real-time datagrams (such as those used by an IP telephony application) from non-real-time traffic (for example, FTP). The specific level of service to be provided is a policy issue determined and configured by the network administrator for that router. We also learned in Section 3.7.2 that two of the TOS bits are used for Explicit Congestion Notification.

- **Datagram length.** This is the total length of the IP datagram (header plus data), measured in bytes. Since this field is 16 bits long, the theoretical maximum size of the IP datagram is 65,535 bytes. However, datagrams are rarely larger than 1,500 bytes, which allows an IP datagram to fit in the payload field of a maximally sized Ethernet frame.
- **Identifier, flags, fragmentation offset.** These three fields have to do with so-called IP fragmentation, a topic we will consider shortly. Interestingly, the new version of IP, IPv6, does not allow for fragmentation.
- **Time-to-live.** The time-to-live (TTL) field is included to ensure that datagrams do not circulate forever (due to, for example, a long-lived routing loop) in the network. This field is decremented by one each time the datagram is processed by a router. If the TTL field reaches 0, a router must drop that datagram.
- **Protocol.** This field is typically used only when an IP datagram reaches its final destination. The value of this field indicates the specific transport-layer protocol to which the data portion of this IP datagram should be passed. For example, a value of 6 indicates that the data portion is passed to TCP, while a value of 17 indicates that the data is passed to UDP. For a list of all possible values, see **[IANA Protocol Numbers 2016]**. Note that the protocol number in the IP datagram has a role that is analogous to the role of the port number field in the transport-layer segment. The protocol number is the glue that binds the network and transport layers together, whereas the port number is the glue that binds the transport and application layers together. We'll see in **Chapter 6** that the link-layer frame also has a special field that binds the link layer to the network layer.

- **Header checksum.** The header checksum aids a router in detecting bit errors in a received IP datagram. The header checksum is computed by treating each 2 bytes in the header as a number and summing these numbers using 1s complement arithmetic. As discussed in **Section 3.3**, the 1s complement of this sum, known as the Internet checksum, is stored in the checksum field. A router computes the header checksum for each received IP datagram and detects an error condition if the checksum carried in the datagram header does not equal the computed checksum. Routers typically discard datagrams for which an error has been detected. Note that the checksum must be recomputed and stored again at each router, since the TTL field, and possibly the options field as well, will change. An interesting discussion of fast algorithms for computing the Internet checksum is **[RFC 1071]**. A question often asked at this point is, why does TCP/IP perform error checking at both the transport and network layers? There are several reasons for this repetition. First, note that only the IP header is checksummed at the IP layer, while the TCP/UDP checksum is computed over the entire TCP/UDP segment. Second, TCP/UDP and IP do not necessarily both have to belong to the same protocol stack. TCP can, in principle, run over a different network-layer protocol (for example, ATM) **[Black 1995]**) and IP can carry data that will not be passed to TCP/UDP.

- **Source and destination IP addresses.** When a source creates a datagram, it inserts its IP address into the source IP address field and inserts the address of the ultimate destination into the destination IP address field. Often the source host determines the destination address via a DNS lookup, as discussed in **Chapter 2**. We'll discuss IP addressing in detail in **Section 4.3.3**.
- **Options.** The options fields allow an IP header to be extended. Header options were meant to be used rarely—hence the decision to save overhead by not including the information in options fields in every datagram header. However, the mere existence of options does complicate matters—since datagram headers can be of variable length, one cannot determine a priori where the data field will start. Also, since some datagrams may require options processing and others may not, the amount of time needed to process an IP datagram at a router can vary greatly. These considerations become particularly important for IP processing in high-performance routers and hosts. For these reasons and others, IP options were not included in the IPv6 header, as discussed in **Section 4.3.5**.
- **Data (payload).** Finally, we come to the last and most important field—the *raison d'etre* for the datagram in the first place! In most circumstances, the data field of the IP datagram contains the transport-layer segment (TCP or UDP) to be delivered to the destination. However, the data field can carry other types of data, such as ICMP messages (discussed in **Section 5.6**).

Note that an IP datagram has a total of 20 bytes of header (assuming no options). If the datagram carries a TCP segment, then each (non-fragmented) datagram carries a total of 40 bytes of header (20 bytes of IP header plus 20 bytes of TCP header) along with the application-layer message.

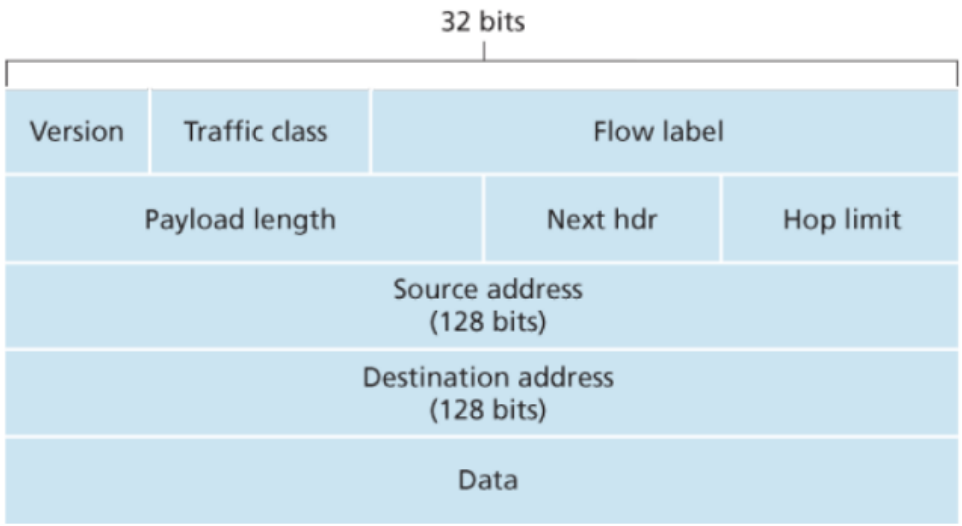


Figure 4.26 IPv6 datagram format

The format of the IPv6 datagram is shown in Figure 4.26. The most important changes introduced in IPv6 are evident in the datagram format:

- **Expanded addressing capabilities.** IPv6 increases the size of the IP address from 32 to 128 bits. This ensures that the world won't run out of IP addresses. Now, every grain of sand on the planet can be IP-addressable. In addition to unicast and multicast addresses, IPv6 has introduced a new type of address, called an **anycast address**, that allows a datagram to be delivered to any one of a group of hosts. (This feature could be used, for example, to send an HTTP GET to the nearest of a number of mirror sites that contain a given document.)
- **A streamlined 40-byte header.** As discussed below, a number of IPv4 fields have been dropped or made optional. The resulting 40-byte fixed-length header allows for faster processing of the IP datagram by a router. A new encoding of options allows for more flexible options processing.

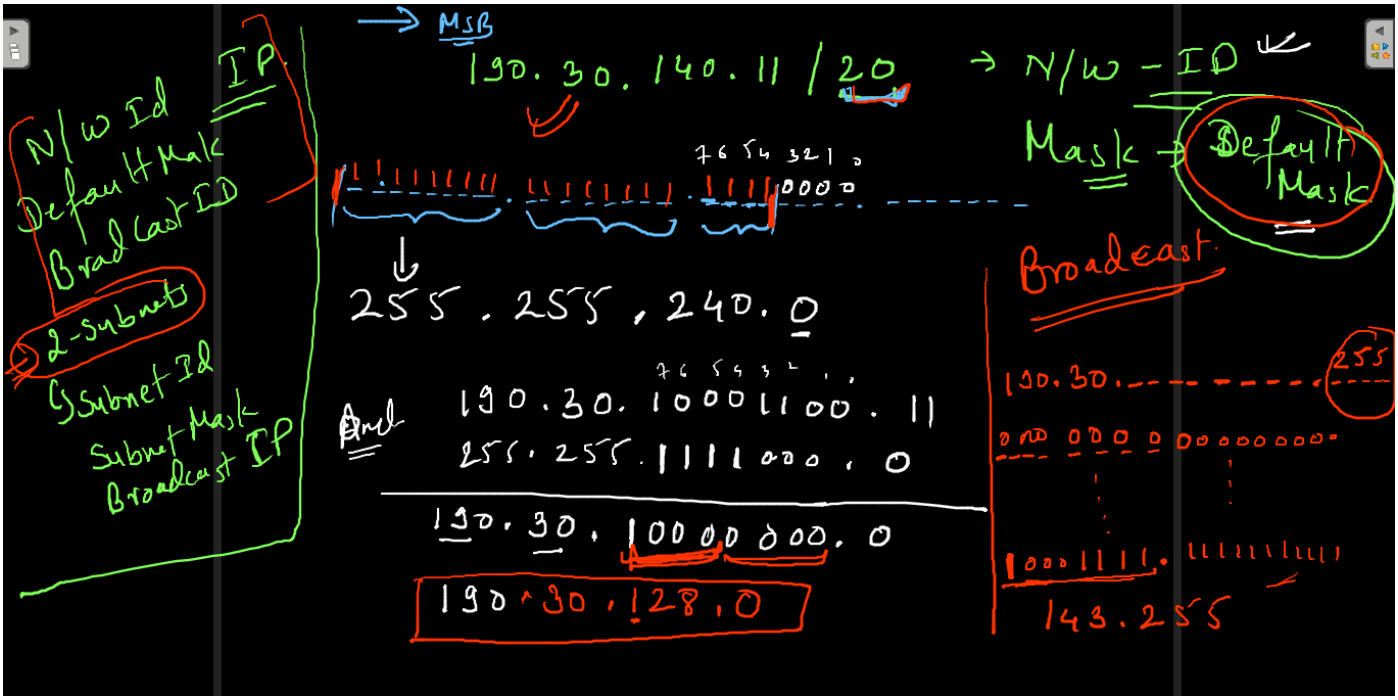
- **Flow labeling.** IPv6 has an elusive definition of a flow. RFC 2460 states that this allows "labeling of packets belonging to particular flows for which the sender

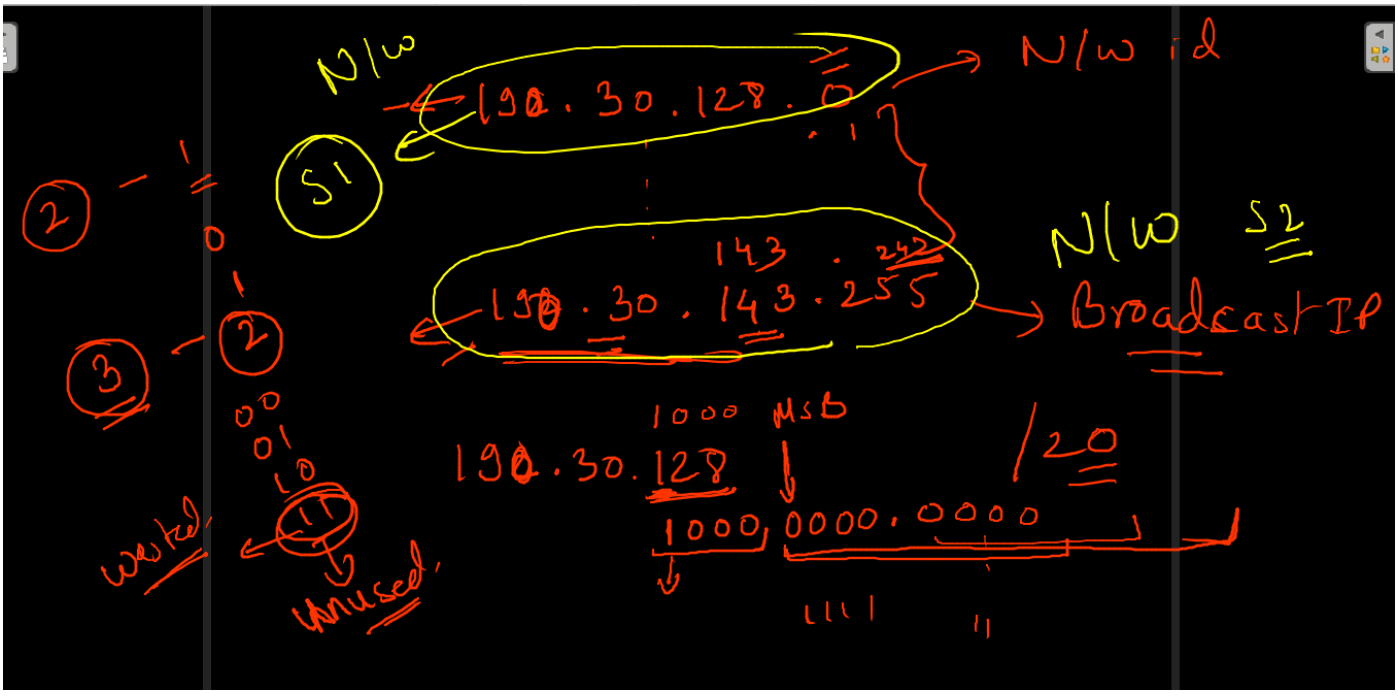
requests special handling, such as a non-default quality of service or real-time service." For example, audio and video transmission might likely be treated as a flow. On the other hand, the more traditional applications, such as file transfer and e-mail, might not be treated as flows. It is possible that the traffic carried by a high-priority user (for example, someone paying for better service for their traffic) might also be treated as a flow. What is clear, however, is that the designers of IPv6 foresaw the eventual need to be able to differentiate among the flows, even if the exact meaning of a flow had yet to be determined.

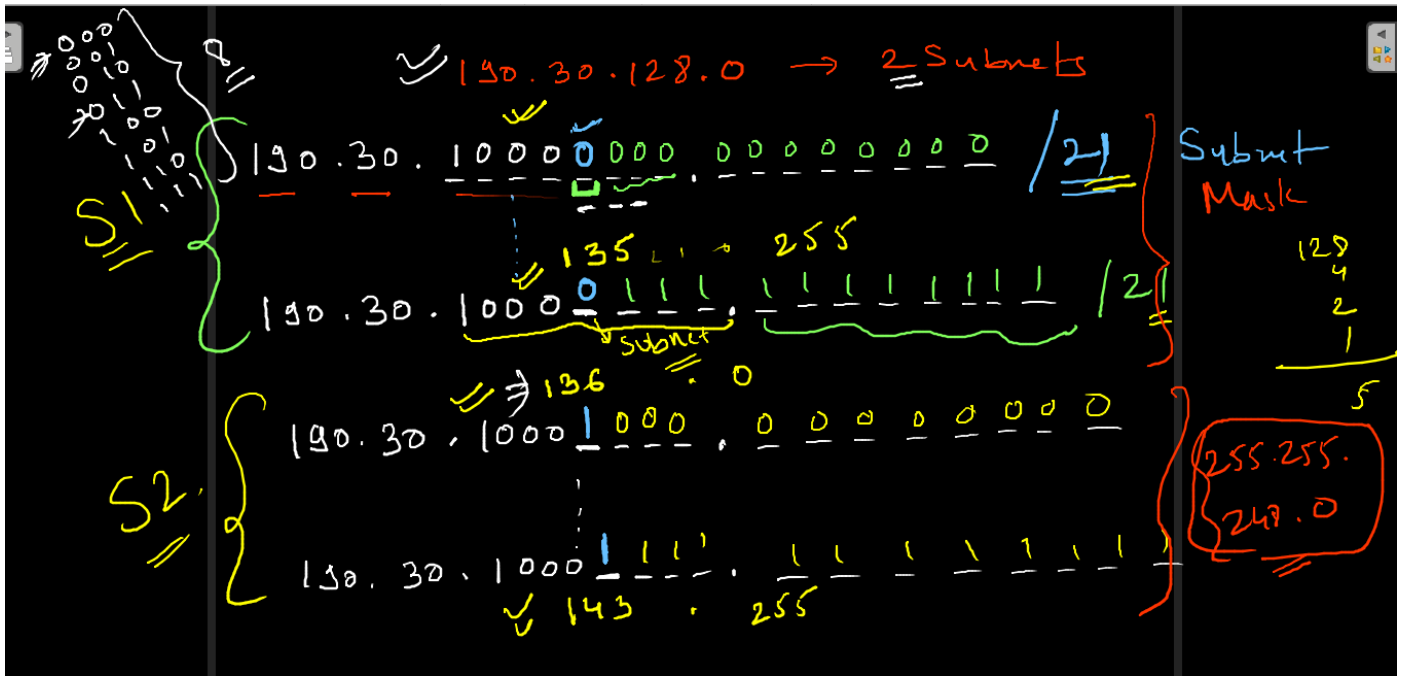
- **Version.** This 4-bit field identifies the IP version number. Not surprisingly, IPv6 carries a value of 6 in this field. Note that putting a 4 in this field does not create a valid IPv4 datagram. (If it did, life would be a lot simpler—see the discussion below regarding the transition from IPv4 to IPv6.)
- **Traffic class.** The 8-bit traffic class field, like the TOS field in IPv4, can be used to give priority to certain datagrams within a flow, or it can be used to give priority to datagrams from certain applications (for example, voice-over-IP) over datagrams from other applications (for example, SMTP e-mail).
- **Flow label.** As discussed above, this 20-bit field is used to identify a flow of datagrams.
- **Payload length.** This 16-bit value is treated as an unsigned integer giving the number of bytes in the IPv6 datagram following the fixed-length, 40-byte datagram header.
- **Next header.** This field identifies the protocol to which the contents (data field) of this datagram will be delivered (for example, to TCP or UDP). The field uses the same values as the protocol field in the IPv4 header.
- **Hop limit.** The contents of this field are decremented by one by each router that forwards the datagram. If the hop limit count reaches zero, the datagram is discarded.

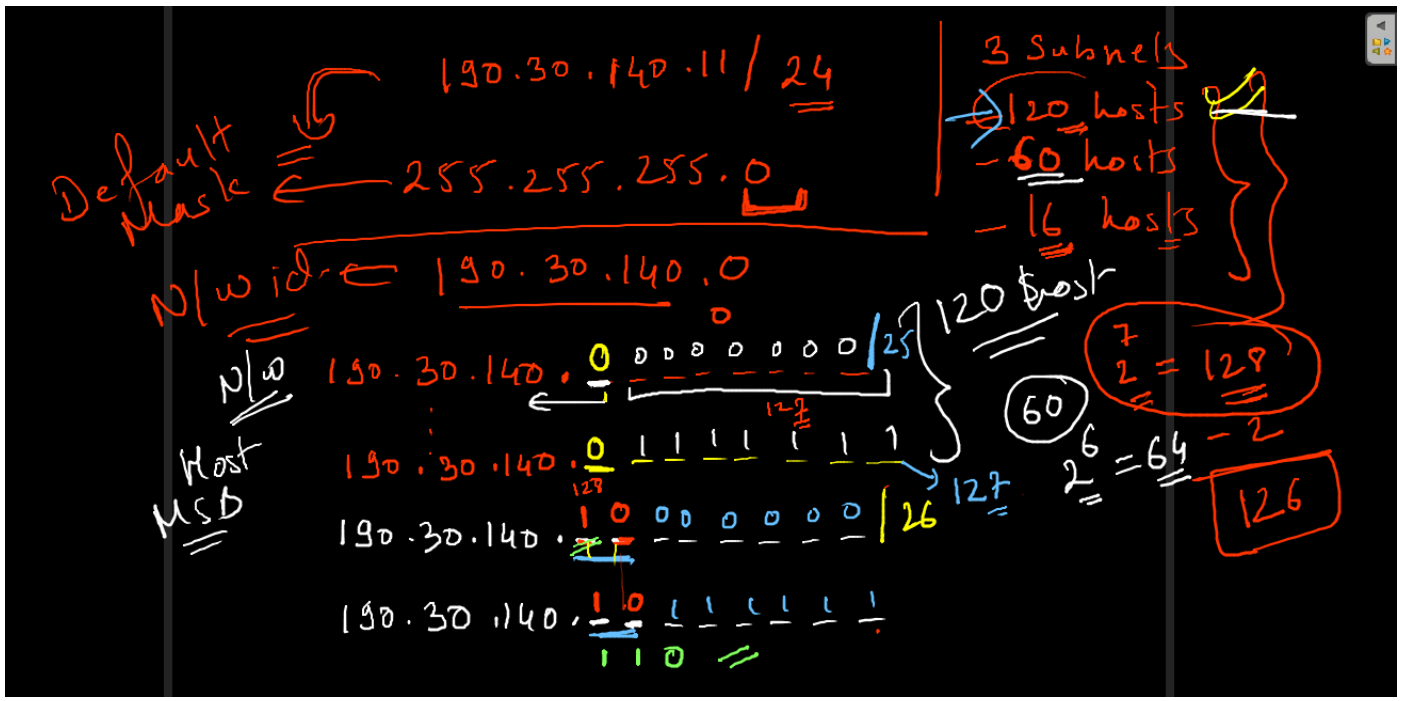
- **Source and destination addresses.** The various formats of the IPv6 128-bit address are described in RFC 4291.
- **Data.** This is the payload portion of the IPv6 datagram. When the datagram reaches its destination, the payload will be removed from the IP datagram and passed on to the protocol specified in the next header field.

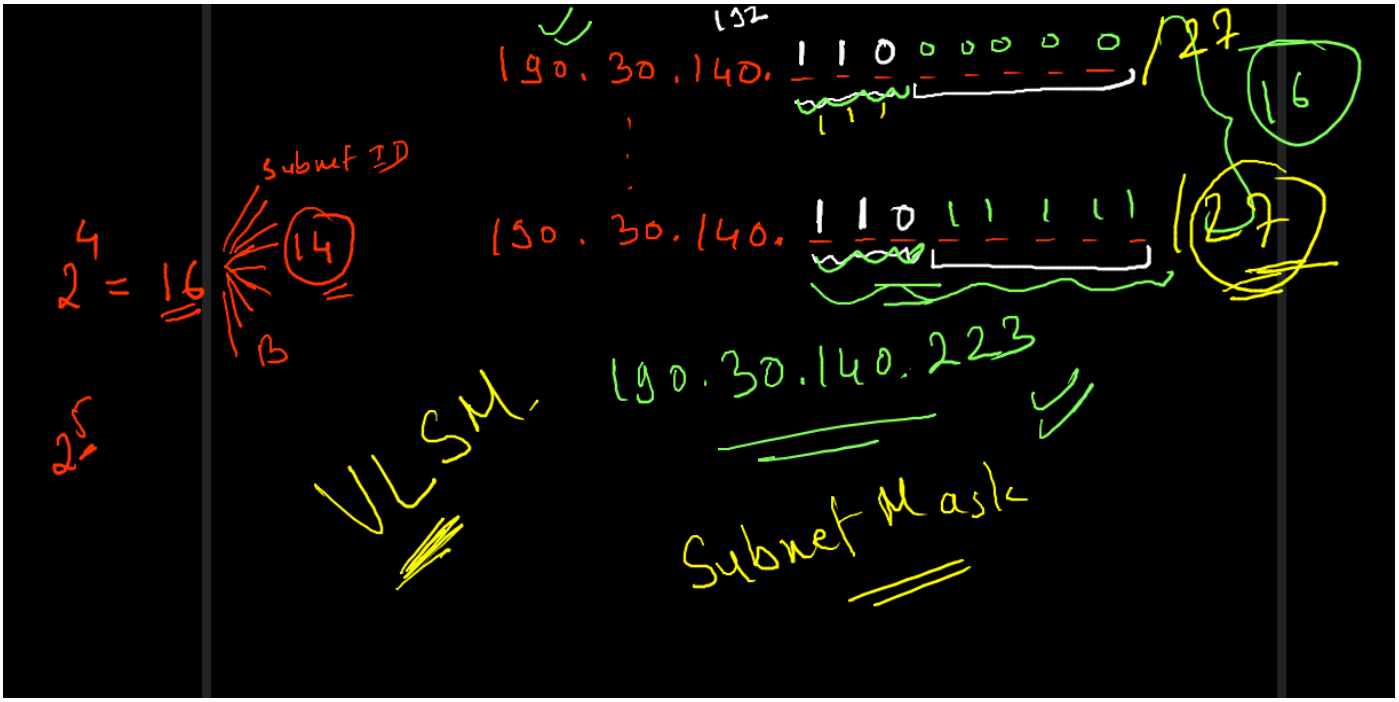
Practice


MSB
190.30.140.11 /20
→ N/w - ID
IP
N/w Id
Default Mask
Brad cast ID
2-Subnets
Subnet Id
Subnet Mask
Broadcast IP
Mask → Default Mask
255.255.240.0
And
190.30.10001100.11
255.255.1111000.0
190.30.10000000.0
190.30.128.0
Broadcast
190.30.
255
143.255


N/w
S1
190.30.128.0
N/w id
143
190.30.143.255
N/w S2
Broadcast IP
2 - 1 = 1
3 - 2 = 1
00
01
10
Wasted
Unused
1000 MSB
190.30.128
/20
1000 0000.0000


190.30.128.0 → 2 Subnets
190.30.10000000.00000000 /21
Subnet Mask
128
4
2
1
5
255.255.248.0
S1
S2
135
255
Subnet
0
136
190.30.1000 1000.00000000
190.30.1000 1111.11111111
143
255


190.30.140.11 / 24
Default Mask ← 255.255.255.0
N/w id. ← 190.30.140.0
N/w
Host
MSD
190.30.140.0 00000000 /25
190.30.140.0 1111111
127
190.30.140.10 000000 /26
190.30.140.110 111111
110
3 Subnets
120 hosts
60 hosts
16 hosts
120 hosts
60
2^6 = 64 - 2
126
2^7 = 128



A datagram of 3000B (20 byte of IP header + 2980 byte IP payload) reached at Router and must be forward to link with MTU of 500Byte. How many fragments will be generated and also write MF, offset, Total length value for all.

A datagram of 3000B (20 byte of IP header + 2980 byte IP payload) reached at Router and must be forward to link with MTU of 500Byte. How many fragments will be generated and also write MF, offset, Total length value for all.

MTU
Maximum Transmission Unit

3000B

0	DF	MF

1 → 20+480 byte

Identific^n bit	flag	fragment-offset

A datagram of 3000B (20 byte of IP header + 2980 byte IP payload) reached at Router and must be forward to link with MTU of 500 Byte. How many fragments will be generated and also write MF, offset, Total length value for all.

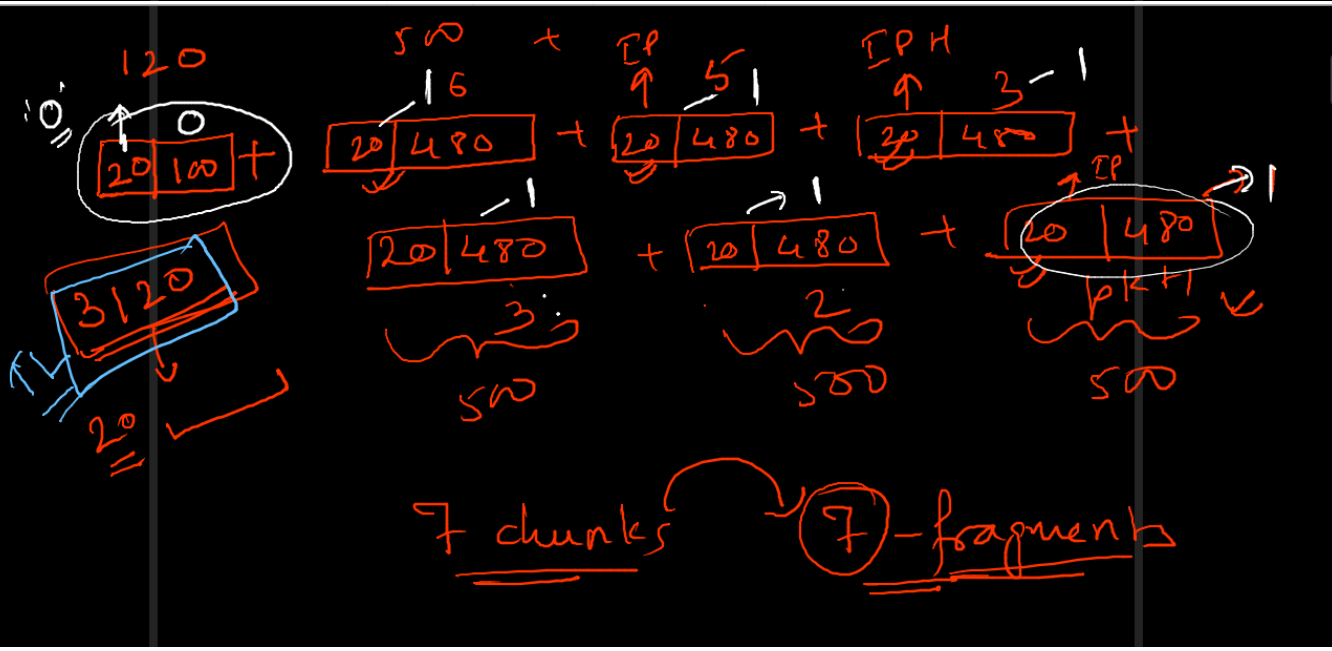
3000 Byte

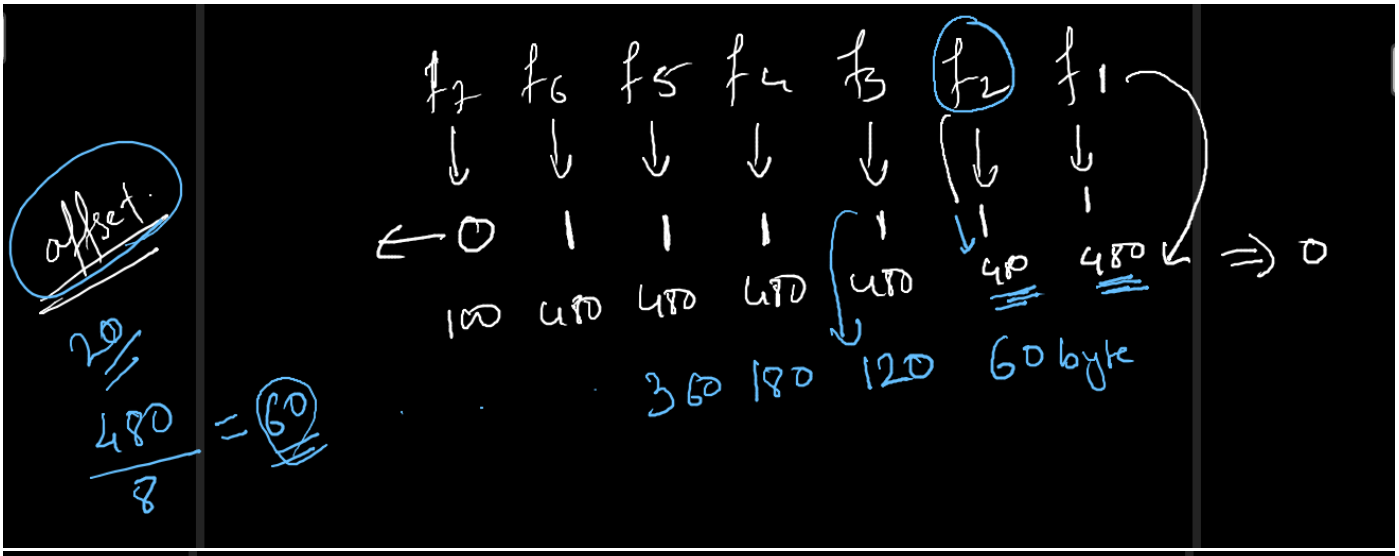
20 Byte = 480 byte Header + Data

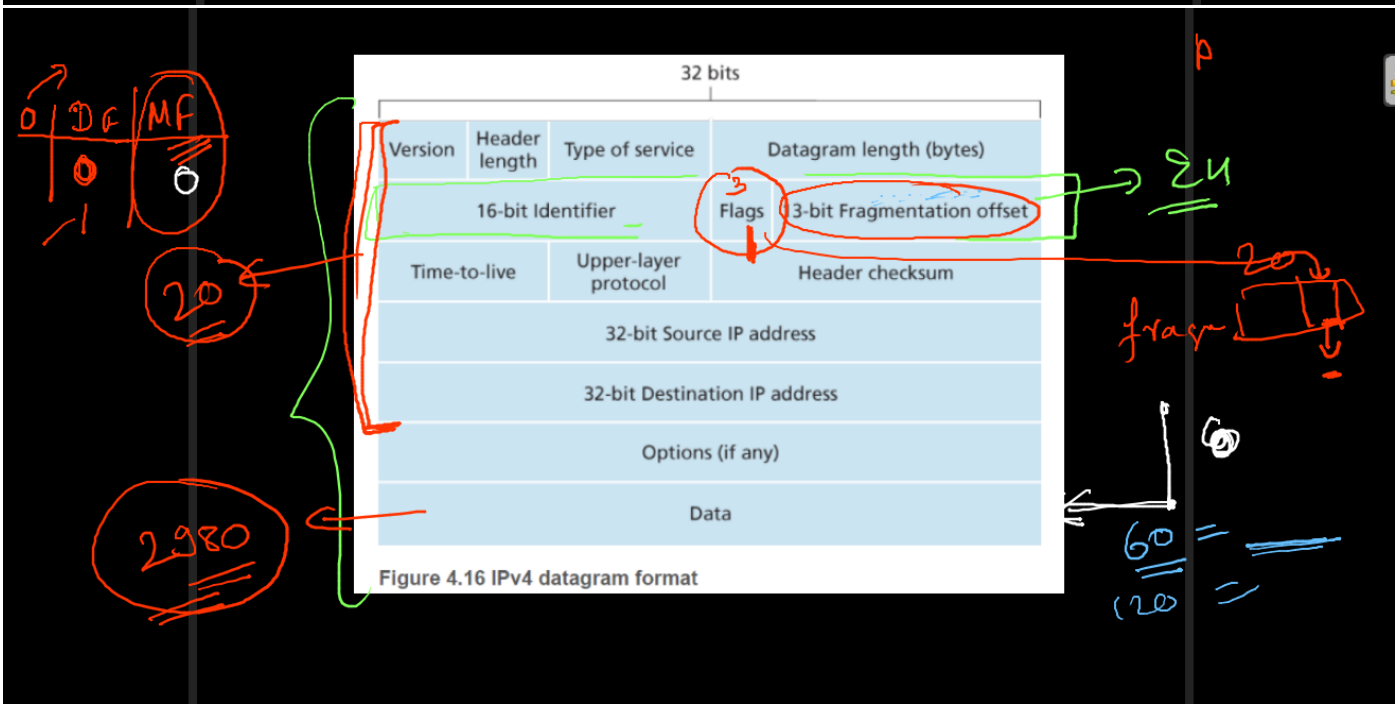
$\frac{2980}{480} = 6.2083$

$100 + 480 + 480 + 480 + 480 + 480 + 480$

$= 2880$ (2980)




offset
20
480/8 = 60
f7 f6 f5 f4 f3 f2 f1
0 1 1 1 1 1 1
100 480 480 480 480 40 480
360 180 120 60 byte
⇒ 0


0 DF MF
0 0
20
2980
32 bits
Version
Header length
Type of service
Datagram length (bytes)
16-bit Identifier
Flags
3-bit Fragmentation offset
Time-to-live
Upper-layer protocol
Header checksum
32-bit Source IP address
32-bit Destination IP address
Options (if any)
Data
Figure 4.16 IPv4 datagram format
24
20
60
60
120