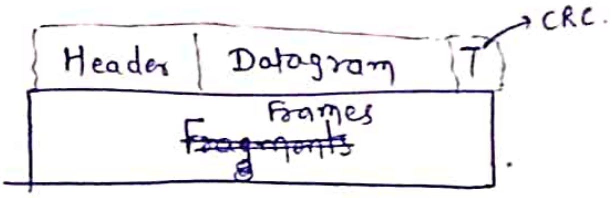
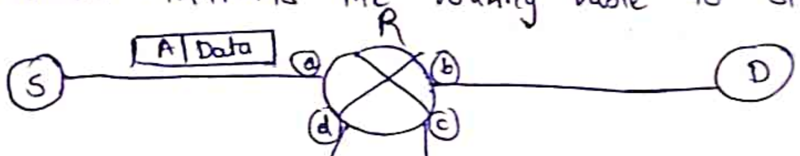


→ each router maintains the routing table to store all the information.



Header Value	Interface
A	b
D	c
⋮	⋮

↑ routing table.

Forwarding ⇒ taking a packet from one interface to another interface.

Routing ⇒ shortest path from source to destination

* Service Provided by Network layer

→ transfer segment from sender to receiving host.

→ On sending side, it incapsulates segments into datagram.

→ On receiving side, decapsulates segments and delivered it to transport layer

* Forwarding

→ Move packets from router's input to appropriate router's output.

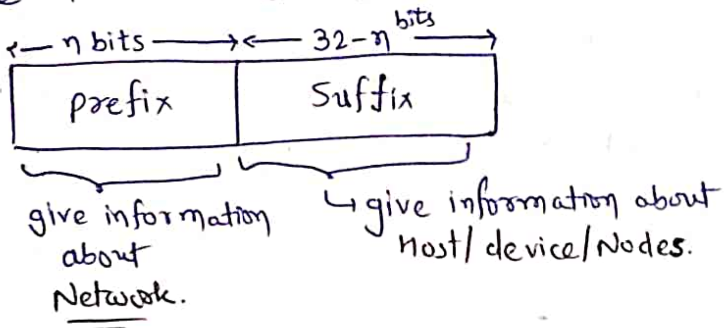
* Routing

→ determine the route to be taken by packet from source to destination

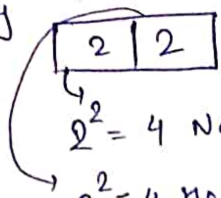
→ forwarding & routing is best services provided by network layer.

+ IPv4 Address (32 bit)

↳ An identifier used in the network layer to identify the connection of each device to the internet.



eg



$2^2 = 4$ Network. possible.

$2^2 = 4$ Hosts/IP/devices

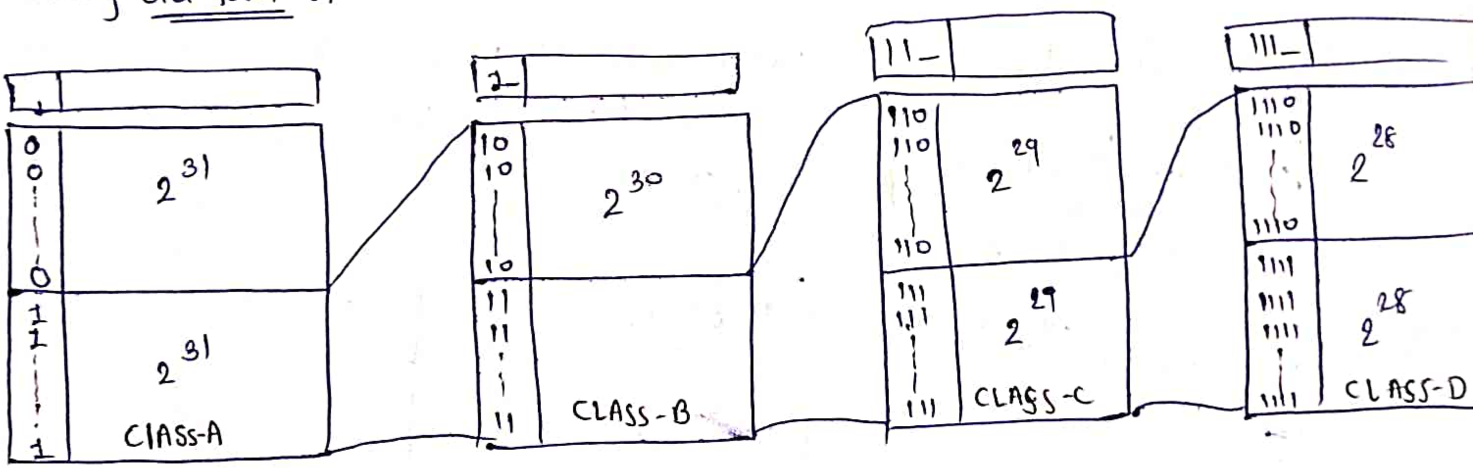
↳ each network have 4 hosts.

→ IP address

128.162.21.18

* Classful IP Address

↳ very old form of IP Address.

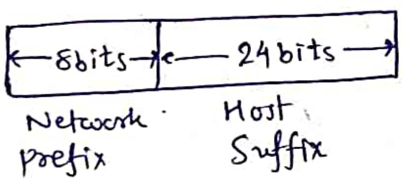


1) CLASS-A IP-ADDRESS

→ default mask = 255.0.0.0

→ ~~To get~~

$X_4 . X_3 . X_2 . X_1$ ← x_1 = first octade.



→ In this 2^{24} bits of hosts are there.

$X_4 . X_3 . X_2 . X_1$

Max value of $x_4 = 127$.

0 ¦ 0 0 0 0 0 0 0 $. X_3 . X_2 . X_1$ ⇒ $127 . X_3 . X_2 . X_1$

0 ¦ 0 0 0 0 0 0 1

0 ¦ 0 0 0 0 0 1 0

0 ¦ 1 1 1 1 1 1 1

7 bits can be vary.

This -host IP Address
↳ 0.0.0.0 (Minimum Address)
It's all known as
NULL Address

↳ We used Null Add. in DHCP.

→ reserved for special purpose. →
0.0.0.0
0.0.0.1
|
0.255.255.255

127.x_3.x_2.x_1

min	127.0.0.0
max	127.255.255.255

→ this is reserved for loop-backed purpose. and for diagonstic purpose.

Range of Class-A ⇒ Now left one :-

1.x_3.x_2.x_1
|
126.x_3.x_2.x_1

→ આ range માંના કોઈપણ IP Address ને class-A IP Address કહેવાય.

eg 122.68.91.8 → Class-A IP Address

2) CLASS-B IP Address

→ class-B IP Add. have 2^{16} bit Address that means they can have 2^{16} hosts/devices.

←16 bit→	←16 bit→

→ ~~2^{16}~~

.x_4.x_3.x_2.x_1

128 ← 10 000000 .x_3.x_2.x_1
10 000001
|
191 ← 10 111111

128.x_3.x_2.x_1
|
191.x_3.x_2.x_1

→ range of Class-B IP Address

↳ class-B માં first 2 fix હોવાથી $2^{16-2} = 2^{14}$ Networks. હશે. (Actual num of network = 2^{14})

→ 155.62 → Network Part.

eg 155.62.8.16 → Class-B IP Address
↳ 2^{14} networks.

8.16 ↓ Hosts

→ Out of 2^{14} Networks.

vary from 0 to 255, 0 to 255

16 bit of Network ⇒
128.1.x_2.x_1
128.2.x_2.x_1
|
128.255.x_2.x_1
129.0.x_2.x_1
129.1.x_2.x_1
|
191.255

3) CLASS-C IP-Address :-

Prefix	Suffix
← 24 bits →	← 8 bits →

→ 2^{21} Network possible.

→ 2^{8} hosts/device/Network.

$X_4 . X_3 . X_2 . X_1$

→ The number of class is to big. but

192 110 | 00000 . $x_3 . x_2 . x_1$
110 | 00001
⋮
223 110 | 11111 . $x_3 . x_2 . x_1$

→ 192 . $x_3 . x_2 . x_1$
⋮
223 . $x_3 . x_2 . x_1$ → range for class-C IP Address.

eg ⇒ 200 . 10 . 5 | 6 → out of 2^{21} Network ~~Nap~~

200.10.5 ↓ Network.

6 ↑ Number of hosts.

192.0.0.0
192.0.0.1
192.0.0.255
|
200.0.0.0
200.10.0.0
$\boxed{200.10.5.6}$ → Network is 200.10.5

Host ⇒ 6

4) CLASS-D IP-Address :- → No Network & Host define

~~Prefix~~	~~Suffix~~
~~← →~~	~~← →~~

$X_4 . X_3 . X_2 . X_1$

Range.

224 . $X_3 . X_2 . X_1$
⋮
239 . $X_3 . X_2 . X_1$

→ Class-D IP-Add. used for multicasting purpose.

5) CLASS-E IP-Address.

Range

240 . $X_3 . X_2 . X_1$
⋮
255 . $X_3 . X_2 . X_1$

↳ used for privacy purpose or millatory purpose.

→ difference b/w multicasting, unicasting, broadcasting

→ ~~Waste~~

* Disadvantage

① ↳ Wastage of IP-Address. → eg જો 500 IP Add. ની જરૂર હોય અને class-B use કરીશું તો $2^{16}-500$ જેટલા IP wastage આવશે.

② ↳ IP Address exhustion happen/ No scalibility and flexibility.

* Classless IP Address :-

a.b.c.d/n → n define prefix.

eg 23.17.26.28/26

23.17.000 11010.00 | 000000
00 | 011100
00 | 111111

→ this is for n.
range from 0 to 63

26 bits | 6 bits

23.17.26.0
23.17.26.1
|
23.17.26.63

→ 23.17.26.~~28~~ 0 /26

		Network Prefix	Host
Class-A	⇒ 1 - 126	(0)	2^{24}
Class-B	⇒ 128 - 191	(10)	2^{16}
Class-C	→ 192-223	(110)	2^{8}
Class-D	⇒ 224-239	(1110)	} Not define Host.
Class-E	⇒ 240-255	(1111)	

eg 40.40.40.40 → belongs to class-A

↳ to determine $\boxed{\text{network id}}$ make suffix bit zero → $\boxed{\text{IMP NOTE}}$

prefix → Network

suffix → Host Num.

Network id ⇒ 40.0.0.0.

eg 140.140.140.140 suffix ⇒ 16 bit. → make it 0 to get N/w id.

N/w id ⇒ 140.140.0.0

first Network of class-B. ⇒ 128.0.0.0

Second N/w of class-B → 128.1.0.0

⋮

128.255.0.0

129.0.0.0.

* Classless

→ classless IP also known as CIDR (Classless Inter Domain Routing).

• Rules Under CIDR

1) → All IP Address should be continuous.

→ It should not be scattered here and there.

2) → The demand of IP Addresses should be in the form of 2^n where 2^n is the size of the block.

3) → 1^{st} IP Address in the block should be devisible by the size of the block. This can be check by looking the last and least significant bits to be zero.

eg 74.10.7.32 – 74.10.7.47

R_1 ⇒ ✓

R_2 ⇒ ✓ $n=4$ → because we have 16 IPs

R_3 ⇒ 74.10.7.32 size of the block. =16

74.10.7.00100000

↳ True ✓

All three Rules follows.

Q Determine the range of IP-Address in CIDR if one of the IP-Add. represented as,

171.43.16.37/27
↳ Prefix.

→ 5 → So total $2^5 = 32$ IP possible.

171.43.16.001|00101
00000 → min

→ 171.43.16.32/27 → Network Id
⋮
171.43.16.63/27

11111 → max.

Network.

→ In CIDR, representation can be done by choosing any IP-Address from the range.

Network Id ⇒ 171.43.16.32/27 ⇒ Host = 0

Q In the given IP address determine the Network Id which can accomodate 1000 Host such that the given IP Add. is the part of that

21.8.1.7

1000 Host ⟹ 2^{10}

21.8.1.7/10

21.8.000000|01.00000111
00.00000000
⋮
11.11111111

$\boxed{\text{Range} \rightarrow 21.8.0.0/22 \text{ to } 21.8.3.255/22}$

→ when there is big network and you want to devide that into small parts it called subneting.

* Subneting :- deviding a bigger network into smaller network is known as Subnet.

Advantage
↳ Network Security can be improved.
↳ maintanance is easy & flexible.

eg
→ 199.211.5.6 class - c → Network Id ⇒ 199.211.5.0
No. of bit reserve for Host → 8

199.211.5.0 | 0 0 0 0 0 0 0 0 ... 0 | 1 1 1 1 1 1 1 or 199.211.5.1 | 0 0 0 0 0 0 0 ... 1 | 1 1 1 1 1 1 1

Network ↲ bit fixed

range ⇒ 199.211.5.0 → Subnet ID
to
199.211.5.127

Subnet-1

↑ first addr of the subnet.

range ⇒ 199.211.5.128 → Subnet ID
to
199.211.5.255

Subnet-2

→ This IP Address known as Directed Broadcast Address. (DBA)
⇓
આ broadcast કરશે msg બધા જ Host ને.

* limited Broadcast ⇒ 255.255.255.255

Broadcast < limited → fixed
Directed → vary

Q 218.13.42.0 /24

① 218.13.42.00|000000
00|000000
⋮
00|111111

② 218.13.42.01|000000
⋮
01|111111

③ 218.13.42.10|000000
⋮
10|111111

④ 218.13.42.11|000000
⋮
11|111111

11 → Subnet id bits.

Subnet [1]	218.13.42.0/24	to	218.13.42.63/24
Subnet [2]	218.13.42.64/24	to	218.13.42.127/24
Subnet [3]	218.13.42.128/24	to	218.13.42.191/24
Subnet [4]	218.13.42.192/24	to	218.13.42.255/24
	Subnet ID		Broadcast Address

→ ~~for each IP address~~

Disadvantage of Subnet → first last IP Address get reserved.

① More time complexity

② Wastage of IP Address.

⇒ Combination of different network to a bigger network is called ~~subnetting~~ Supernetting

* **Subnet Mask :-**

→ It's 32 bit number that seprates an IP Address into 2 parts. That's Network ID & Host ID.

→ It's used to determine the subnet where a perticular host belongs to.

→ subnet mask is made up of 1's and 0's.

1's ⇒ Network id bits + Subnet id bits → the bits which remain fixed to make subnet from some Network IP address.

0's ⇒ Host id bits (we will not include subnet id bits)

Q let's suppose Packet needs to be send to user with IP 218.13.42.132. (Question is linked with previous)

SM :- (Subnet Mask) 24 + 2 = 26 (1's)

6 (0's)

11111111.11111111.11111111.11000000 } Bitwise AND

11011010.00001101.00101010.10000100 } 218.13.42.132

11011010.00001101.00101010.1000000

$\boxed{218.13.42.128}$ ⇒ Subnet ID of Subnet where router have to send packet.

→ Subnet mask used by router.

Class-B IP Address

Q 161.72.49.16 Network ID where this IP address belongs. & devide that Network in to 4 Subnet ID.

161.72.0.0

161.72.00	000000.0	
00	111111.255	
01	00000.0	
01	1111 11.255	
10	000000.0	

S1 161.72.0.0 to 161.72.63.255

S-2 161.72.64.0 to 161.72.127.255

S-3 161.72.128.0 to 161.72.191.255

S-4 161.72.192.0 to 161.72.255.255

→ Classfull IP address followed fixed length Subnet mask which has equal num. of subnets and equal number of host in each subnet.

* VLSM (Variable Length Subnet Mask)

218.13.42.0 | 0 0 0 0 0 0 0
0 | 1 1 1 1 1 1 1
0 - 127

218.13.42.10 | 0 0 0 0 0 0
10 | 1 1 1 1 1 1
128 - 191

218.13.42.110 | 0 0 0 0 0 0
110 | 1 1 1 1 1 1
192 - 255

→ It allows to network into subnet of different sizes.

→ It's considering as subnetting a subnet.

→ classless addressing supports both FLSM and VLSM - where classful add. only support FLSM.

Q 200.1.2.0/24 devide this network into 3 subnet such that 200.1.2.120 should fall in biggest ~~IP Add. among~~ subnet among all subnet?

S1 : 200.1.2.0 - 200.1.2.127 ⇒ 200.1.2.0/25 → biggest will be 1st.

S2 : 200.1.2.128 - 200.1.2.191 → 200.1.2.128/26

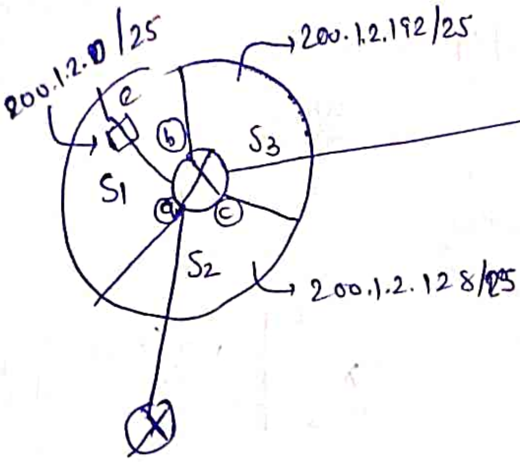
S3 : 200.1.2.192 - 200.1.2.255 → 200.1.2.192/26

→ Subnet Mask

~~S1 : 200.1.2.~~

S1 : 255.255.255.128

S2, S3 → 255.255.255.192



Routing Table.

Design	SM	Interface
200.1.2.0	255.255.255.128	a
200.1.2.128	255.255.255.192	c
200.1.2.192	255.255.255.192	b
0.0.0.0	0.0.0.0	e

default entry in routing table.

0.0.0.0 → dishost entry.

let us,

→ destination IP Add. 200.1.2.194

200.1.2.192/26

11111111 . 11111111 . 11111111 . 01111111
11001000 . 00000001 . 00000010 . 11000000

11001000 . 00000001 . 00000010 . 01000000

$\boxed{200.1.2.128}$ → Not true.

200.1.2.194   255.255.255.192

⇓

AND operation

$\boxed{200.1.2.192}$

Imp

⇒ If there is only one match/one subnet mask, router forward the data packet to the corresponding interface.

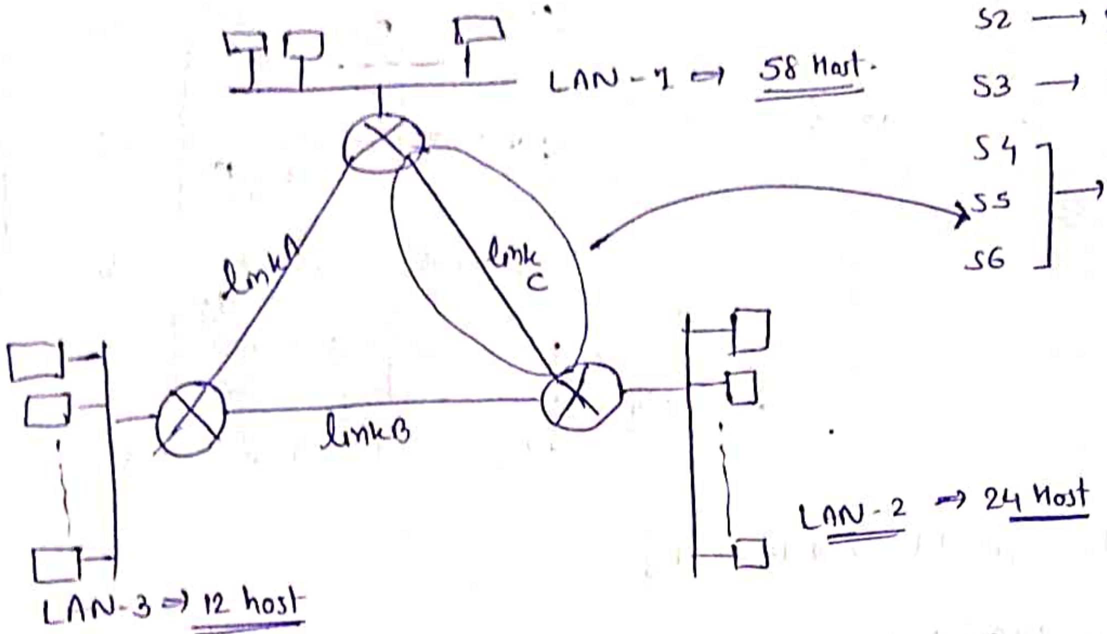
⇒ If there are more than one match/one subnet mask, router forwards the data packet corresponding to the largest subnet mask. (Probability of getting ~~reg~~ Interface increasing)

⇒ If there is no match, router forwards the data packet ~~correspond~~ to the interface corresponding to the default entry/default gateway.

Q. With an IP Address ~~ranging~~ range 174.168.10.0/24

(Imp)

Subnet	Host
S1 →	58
S2 →	24
S3 →	12
S4, S5, S6 →	2 (connecting 2 routers)



→ devide the Network efficiently so that minimum wastage of IP Address?

→ Total number of subnet = 6 (Smallest subnet is link router-router)

* Steps ~~Trade~~

↳ select the network from largest to smallest in size.

1) LAN-1 ⇒ (58)
2) LAN-2 ⇒ (24)
3) LAN-3 ⇒ (12)
4) Link-A
5) Link-B
6) Link-C

Link-A, Link-B, Link-C ⇒ (2)

Subnet	1	2	4	8	16	32	64	128	256
Host	256	128	64	32	16	8	4	2	1
SM	/24	/25	/26	/27	/28	/29	/30	/31	/32

last we can not assign because they already assigned.

N/w id	SM	Possible Hosts.	N/w Allocated
174.168.10.0	/26	62	LAN-1
174.168.10.64	/26	62	
174.168.10.128	/26	62	
174.168.10.192	/26	62	

→ devide into 2 subnets. (SM માં 3 bit fix રહે.)

2 તો fixed + 1 ને add થશે.

255.255.255.192

(64 - 2)

$\boxed{\text{first \& last}}$

	SM	Possible Host	
174.168.10.64	/27	30	LAN-2
174.168.10.96	/27	30	
~~174.168.10.~~	~~/27~~		

→ devide into 2 subnet. (4 bit fixed)

(32-2)

		Possible host	
174.168.10.92	/28	14	LAN-3
174.168.10.112	/28	14	

→ devide into 4 Subnets.

		Host	
174.168.10.112	/30	2	link1
174.168.10.116	/30	2	link2
174.168.10.120	/30	2	link3
174.168.10.124	/30	2	

Q If you want to store 500 Host then How will you find Subnet mask

500 Host → 2^9 bits required.

255. 255. 11111110 .00000000

$\boxed{255.255.254.0}$ → SM

⇒ Suppose, we have host-A with IP Address IP_A and subnet mask of subnet where $A \in S_A$. If host-A want to send packet to host-B whose IP add. is IP_B. then A will do bitwise AND operation with S_A. And then IP_B with S_A in order to know wheather host-B belongs to same subnet or not.

$\boxed{IP_A \text{ bitwise AND } S_A = IP_B \text{ bitwise AND } S_A}$

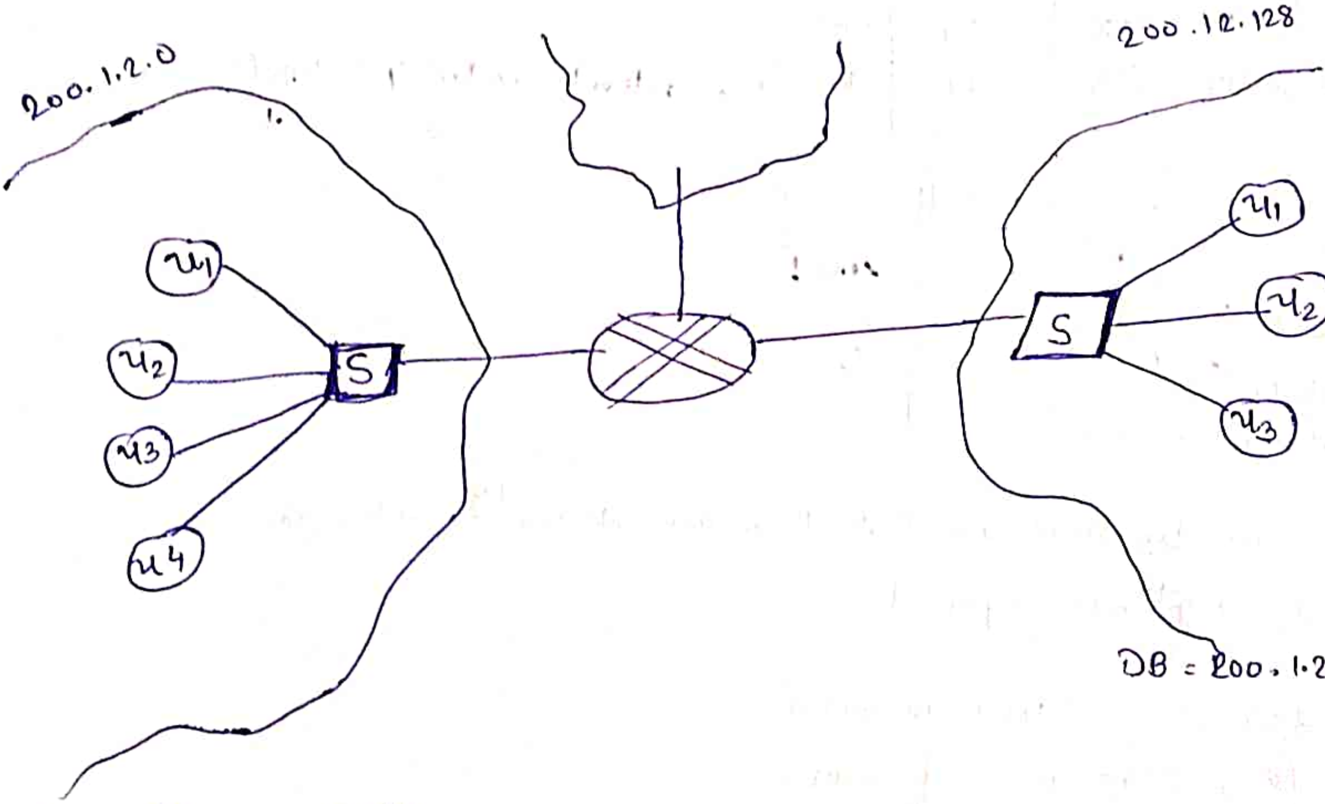
↑ This should be satisfied.

Q IP_A is 200.1.2.134, IP_B 200.1.2.155. Subnet mask where A belongs. S_A = 255.255.255.192 check wheather both are belong to same network or not? → bitwise AND

255. 255 .255. 192	11000000		255.255.255.192	11000000
ff 200. 1. 2. 134	~~01111111~~ 10000110	ff	200.1.2.155	10011011
	$\boxed{200.1.2.128}$		$\boxed{200.1.2.128}$	1

~~255. 255. 255~~
~~200. 1. 2. 134~~

→ both get same so they are from same Network.



Destination Broadcast : 200.1.2.127

SA → 200.1.2.1

Limited Broadcast Add = 255.255.255.255

Source Mac = ___ . ___ . ___ . ___

Dest. MAC = FF. FF. FF. FF. FA.

2 type of broadcasting → limited → within the subnet
directed → ~~within~~ to transfer from one Subnet to another.

→ to find broadcasting address so fix all the bits to "1."

+ IPv4 Header → Connectionless, → datagram Service (travel through any routes)

→ Header min length 20 byte.

Header ⇒ 20-60 byte (range)

Datagram range ⇒ 0 - 65,535 byte
(Header + Segment)

length of data = Total length - (Head. leng. ×4)

→ define total length (header + data) of IP (Payload)

0 VER	4 HLEN	8 Service Type	16 Total length 31
4 bit	4 bit	8 bit	16 bit

IP version (eg IPv4, like 0100)
eg IPv6
IP 0110

length of the header.
eg if HLen 24 bytes.
→ min. length of header should be 20.

→ hlen will be multiple of 4.

eg if hlen is presented $\boxed{0001}$ then multiply with 4. to get the Header length and should be greater than 20.

→ અહીં 5 (0101) થી bit start થશે. કેમ કે તે પહેલાંના બધાને 4 થી multiply કરતા તે invalid ગણી શકે.

define type of service
↳ how datagram should be handled.

P	P	P	D	T	R	C	0

Precedence, delay, throughput, Reliability, cost, Reserved entity (always zero)

0 Normal
1 Maximise

→ Identification, flags, fragmentations are 16, 3, 13 bits each.

+ TTL (8-bit) Time to live.
↳ datagram may be circulating one node to another.
→ create extra traffic.
→ each router decrement the number by 1.
→ when the values reaches 0, router discard the datagram.

+ Protocol (8 bit)
→ A packet from upper layer belongs to different protocol (TCP/UDP).
→ Packet can be from IP layer.
→ dest. knows which protocol packet belongs to.
→ multiplexing at source, demultiplexing at destination.

TCP - 06
UDP - 17

+ Header checksum (16 bit)
→ error check of payload is done by transport layer.
→ Any ~~header~~ error in header is disastrous.
→ needs to recalculated & check at each router.

+ Option (40 byte)

+ Source & destination IP Address → 32 bit each

↳ used for network testing, managment, & debugging purpose.
↳ optional field
↳ generally used by network administrator.

+ Padding
→ IP hader should be multiple of 32-bits (4 bytes).
→ 0 bits are padded.

+ Fragmentation
+ Maximum Transfer Unit (MTU)
→ max size of IP datagram that can be encapsulated in a frame.
size of payload <= MTU
→

Identification	Flags R D M	Flagged offset

+ Fragmentation
→ payload of IP datagram is fragmented.
→ when payload of datagram is fragmented, each fragment has its own headers.
→ datagram may be fragmented serval times before it reaches to destination.
→ Fragmented datagram can be further fragmented if it encounters a network with smaller MTU.

+ Reassembly
→ done at the destination.
→ each fragment is an independent identity.

• Identification (16 bit)
↳ provide uniqueness to each datagram
↳ possitive number ~~is~~ called counter is used.
↳ fragmented ~~dt~~ datagram uses the same identification field.
↳ It helps the destination for reassembly of datagram.

* **Flags :-** (3 bits)
  - left most is reserved.
  - 2nd bit (D bit) → do not fragment bit (0 : fragment, 1 : do not fragment)
  - 3rd bit (M bit) → more fragment bit (0 : No more fragments, 1 : more frag to come)
    - ↳ 0 → last packet

* **Fragmentation offset** (13 bits)
  - → relative position of fragment wrt to datagram.
  - → Measured in unit of 8 bytes. (Multiple of 8)

Q A datagram of length 5000 bytes with 20 bytes of header in it reached a router. The router has to forward the datagram on the link whose MTU is 700 bytes. How many fragments the router has to do? Determine the total length of each fragmented packet, the M bit and the fragmentation offset.

$20 + 4980 = 5000$ byte ⟹ datagram

↓ Header, ↳ Payload.

$20 + 680 = 700$ byte ⇒ Accomodate the data.

↑ should be multiple of 8 ✓.

No of fragments $= \left\lceil \frac{4980}{680} \right\rceil = \boxed{8}$ → 8 fragment will be there.

	FP1	FP2	FP3	FP4	FP5	FP6	FP7	FP8
	(20+680)	(20+680)	(20+680)	(20+680)	(20+680)	(20+680)	(20+680)	(20+220)
M bit ⇒	1	1	1	1	1	1	1	0
Fragmentation offset →	0	85	170	255	340	425	510	595

(M bit) (more fragment = 0 ⇒ Last packet)

આની પહેલા 680 byte જતા રહ્યા છે.

(85 × 8 = 680)

* If MTU = 695 bytes.

$20 + 675 = 695$ bytes.

$$\text{fragments} = \left\lceil \frac{4980}{672} \right\rceil = \underline{8 \text{ fragments}}$$

	FP1	FP2	FP3	FP4	FP5	FP6	FP7	FP8
	(20+672)	(20+672)	(20+672)	(20+672)	(20+672)	(20+672)	(20+672)	(20+87
M →	1	1	1	1	1	1	1	0
FO →	0	84	168	252	336	420	504	588

* Routing :-

↳ forwarding transfer interface from one point to another.

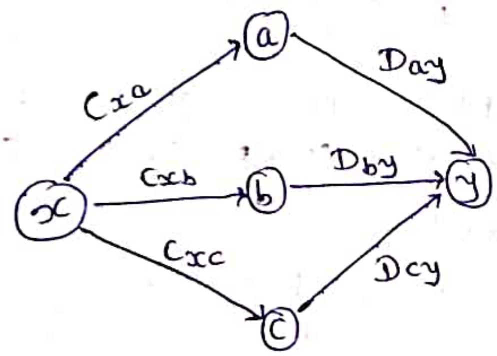
↳ routing table made through routers.

• Distance Vector Routing :-

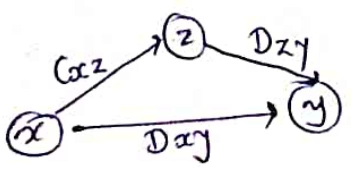
→ Made up of 2 parts.

1) Bellman - Ford Equation

→ It's used to find the least cost b/w source node x and destination node y. through some intermediate node. through a, b, c.



$$D_{xy} = \min \{ (C_{xa} + D_{ay}), (C_{xb} + D_{by}), (C_{xc} + D_{cy}) \}$$



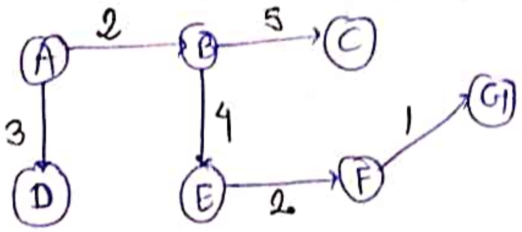
$$D_{xy} = \min \{ D_{xy}, C_{xz} + D_{zy} \}$$

⇒ Generliged Version / Egⁿ

$$\boxed{d_x(y) = \min_v \{ c(x,v) + d_v(y) \}}$$

x — source, y — destination, v — intermediate node, $c(x,v)$ — cost from source to v, $d_v(y)$ — v to y.

2) Distance Vector
→ It's one dimensional array to represent the tree.



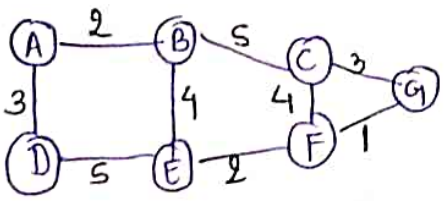
→ There should not be any loop in tree.

→ distance vector at "a"

	A
A	0
B	2
C	7
D	3
E	6
F	8
G	9

→ distance vector at $\boxed{F}$ → distance b/w all the node from given point
↳ we can store it in 1-D array.

⇒ distance vector provide least cost to the other nodes in the network.
⇒ node send some greeting messages outof it's interfaces and discovers the identity of the imediate neighbour and distance b/w itself and each neighbour.



⇒ Network → માં loop હોઈ શકે પણ treeમાં loop નહીં હોય.

intially, at node A.

	A		B
A	0	A	2
B	2	B	0
C	∞	C	5
D	3	D	∞
E	∞	E	4
F	∞	F	∞
G	∞	G	∞

→ આમ જ રીતે બધા જ ના node ના distance vector મેળવી શકીશું. only neighbour ને જ consider કરીશું.

→ After time t select any node and copy the neighbour of another node and make a new Vector.

	B new
A	2
B	0
C	~~8~~5
D	5
E	4
F	∞
G	∞

$d_B(A) = C(B,A) + d_A(A) = 2+0 = 2$

$d_B(B) = C(B,B) + d_B(B) = 0$

$d_B(C) = C(B,C) + d_C(C) = 5 + \cancel{8}\,0 = 5$

$d_B(D) = C(B,D) + d_D(D) = 5+\infty = \infty$

$= C(B,E) + d_E(D) = 4+5 = 9$

$= C(B,A) + d_A(D) = 2+3 = 5$

min = $\boxed{5}$

* distance vector Property

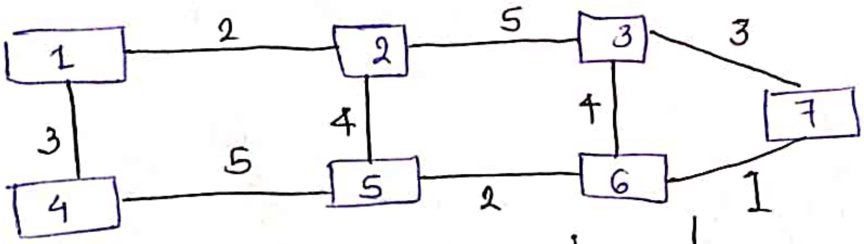
→ distributed

→ Itterative

↳ Self terminating Process.

→ Asynchronous.

→ All source single destination algorithm is same as distance vector.



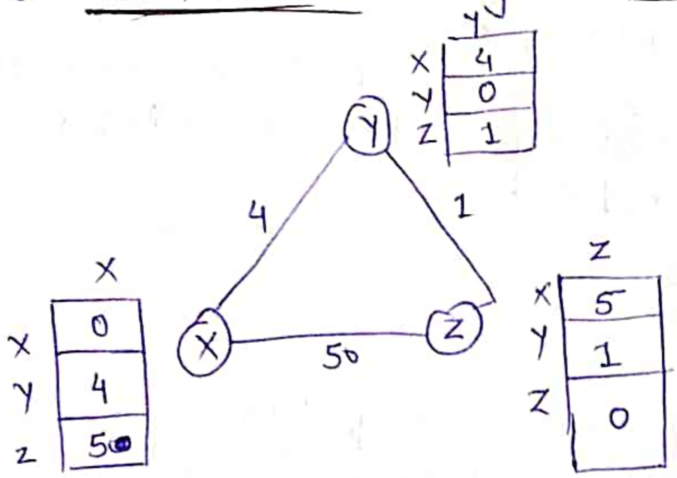
destination = N7

Cost (Next Hope)

N1	N2	N3	N4	N5	N6
∞ (-1)	∞ (-1)	3 (7)	∞ (-1)	∞ (-1)	1 (7)
∞ (-1)	8 (3)	3 (7)	∞ (-1)	3 (6)	1 (7)
10 (2)	7 (5)	3 (7)	8 (5)	3 (6)	1 (7)
9 (2)	7 (5)	3 (7)	8 (5)	3 (6)	1 (7)
9 (2)	7 (5)	3 (7)	8 (5)	3 (6)	1 (7)

↳ If are got same other time then it will self terminated.

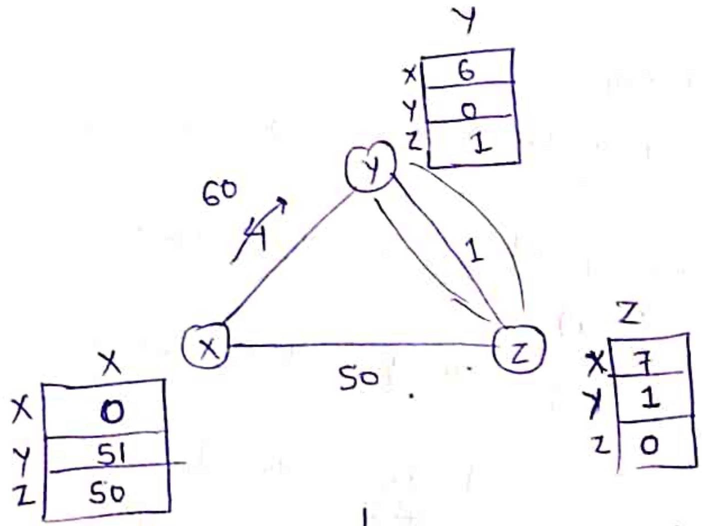
→ The procedure we just discussed known as distributed bellman Ford Algorithm or distance vector routing or all sources single destination routing.



→ Suppose at time t_0 the cost of X-Y link reduces to 1 from 4.

→ Y ditects link node change, update it's DV and send to it's neighbour.

→ At time t_0 cost of the link X-Y changes to 60 from 4.



→ This is routing loop problem or count to infinite problem.

↳ જ્યાં સુધી Z 50 around નહીં આવે ત્યાં સુધી Y-Z માં loop આવતી રહેશે અને X - update નહીં થાય એટલે લગભગ 50 વખત loop iterate થશે.

$$\Rightarrow D_Y(x) = \min\left\{\begin{array}{l} c(Y,X) + d_X(X), \\ c(Y,Z) + d_Z(X) \end{array}\right\}$$

$$= \min\{(60+0), (1+5)\}$$

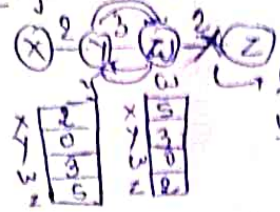
$$\boxed{D_Y(x) = 6}$$

$$\Rightarrow D_Z(x) = \min\left\{\begin{array}{l} c(Z,X) + d_X(x), \\ c(Z,Y) + d_Y(x) \end{array}\right\}$$

$$= \min\{(50+0), (1+6)\}$$

$$\boxed{D_Z(x) = 7}$$

→ This is keep on increasing and it known as <u>routing loop problem</u>.
It also know as <u>count to infinity Problem</u>.
↳ ઠાઈ ગુકાવાં



↳ loop will be created after W-Z link will broken.

* 1st solution

1) Split Horizon :- (refere a figure given above behind)
→ If Z thinks that it's best route to X is via Y then Z does not send the cost it has. That's no updation is send to Y from Z.

2) Split Horizon with Poisoned Reverse
→ If Node Z thinks that it's best route to X is via Y. then Z advertises its cost to X as ∞. ~~then Z advertises its cost~~ therefore Y will not route to X via Y.

+ Link-State Routing Protocol/Algorithm

↳ Each network node has complete map of the network known as link state Database. (LSD).
↳ LSDB is achieved using <u>Link State packet</u>. It has the identity of the node and cost of the link. (LSP)
↳ All nodes have same information using flooding called linked state broadcast. (LSB)
→ ~~At the end~~ Broadcast થયા પછી તે flood થશે information.
then at the end we have ~~broadcast~~ link state Database.
→ LSDB will stay to all the node. DB will be like
→ this will be store or access by all the nodes present in the Network.

	A (LSP(A))	B (LSP(B))	C	D	E	F	G
A	0	2					
B	2	0					
C	∞	2					
D	3	∞					
E	∞	1					
F	∞	∞					
G	∞	∞					

→ Single Source All destination Routing.

eg

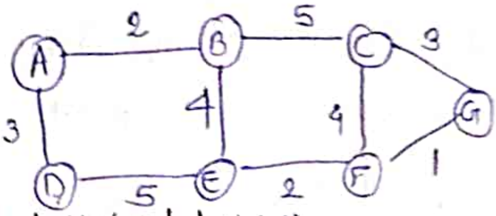
$C(x,y) \Rightarrow$ link cost from x to y

$d(V) \Rightarrow$ Current value of the cost of ~~per~~ the path from source to destination V.

$P(V) \Rightarrow$ Predecessor node along the path from source to V.

$N \Rightarrow$ Set of Nodes whose least cost path are known.

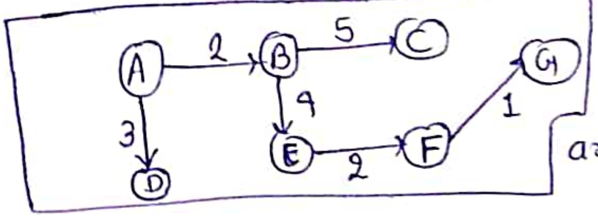
If A is the source.



lock the minimum one.

N	d(B) p(B)	d(C) p(C)	d(d) p(d)	d(E) p(E)	d(F) p(F)	d(G) p(G)
A	2, A locked	∞	3, A	∞	∞	∞
AB	2, A	7, B	3, A lock	6, B	∞	∞
ABD	2, A	7, B	3, A	6, B locked	∞	∞
ABDE	2, A	7, B locked	3, A	6, B	8, E	∞
ABDEC	2, A	7, B	3, A	6, B	8, E lo	10, E
ABDECF	2, A	7, B	3, A	6, B	8, E	9, F locked.
ABDECFG	2, A	7, B	3, A	6, B	8, E	9, F

$\Rightarrow$ This tree is formed assuming A is Node source & all the other node are destination



→ This is also known as dijkstras Algorithm / link state router / shortest Path first Algo / single source All destination Algo.

→ All automation system follows RIP / OSPF protocol.

↳ This is called interior Gateway Protocol.

→ Inside the Autonomous Sys. ⇒ IGP (All these read from Notes)

+ Address Resolution Protocol (ARP) → NL ના data ને end માં રાઉટર સુધી લાવશે ત્યારે ત્યાંથી frame ના રૂપમાં તેને આગળ મોકલશે.

→ IP = 32 bit   MAC = 48 bit

→ DLL deals with Hop by Hop transmission.

→ to use MAC add of router/host

→ ARP req. packet is sent that contains MAC & IP add of sender and IP add of destination.