CS & IT ENGINEERING

COMPUTER NETWORKS

IPv4 Header & Fragmentation

Lecture No-05

By- Ankit Doyla Sir

Fragmentation in IPv4

Difference B/W Fragmentation & Segmentation

IPv4 Header

VER	HL	Services	Total Length (16 bit)
Identification No.		Flags	Fragment offset
Time to Live		Protocol	Header checksum
Source IP Address			
Destination IP Address			
Option			

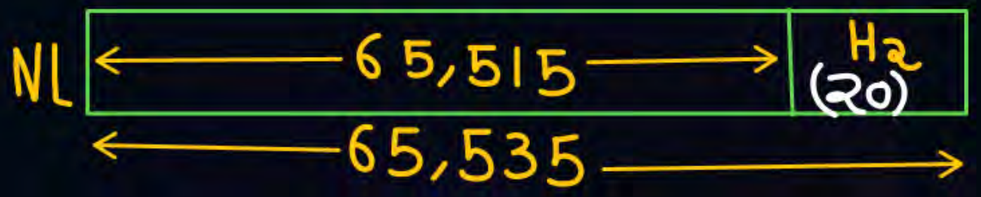
Total Length = 16 bit

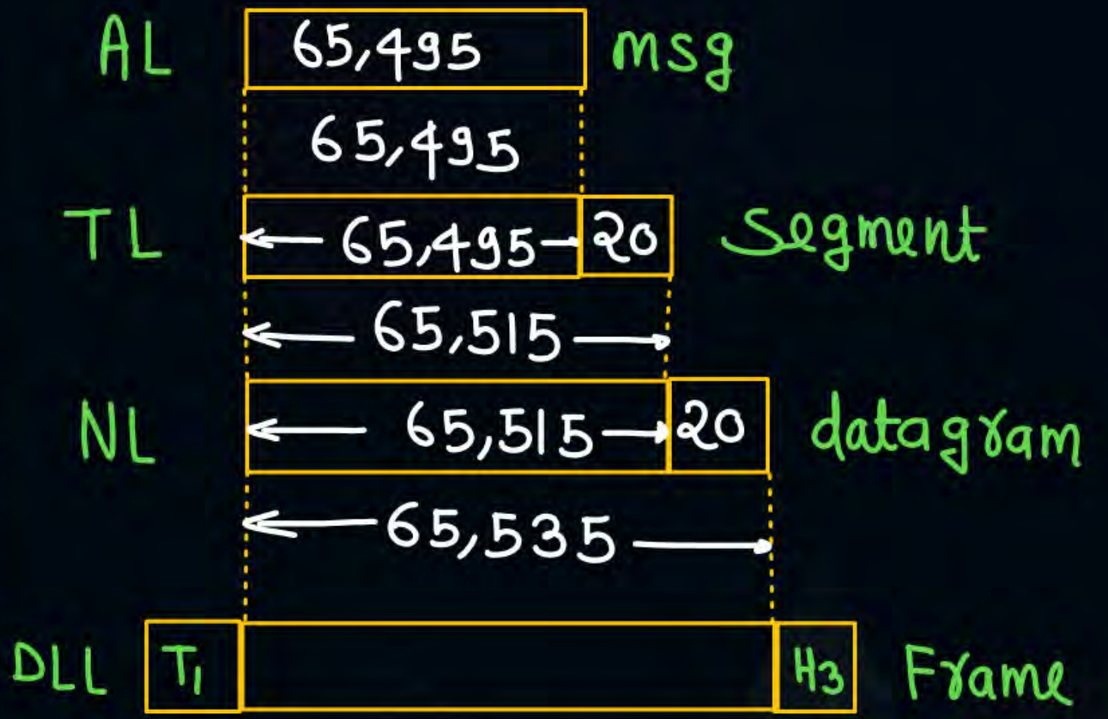
↓

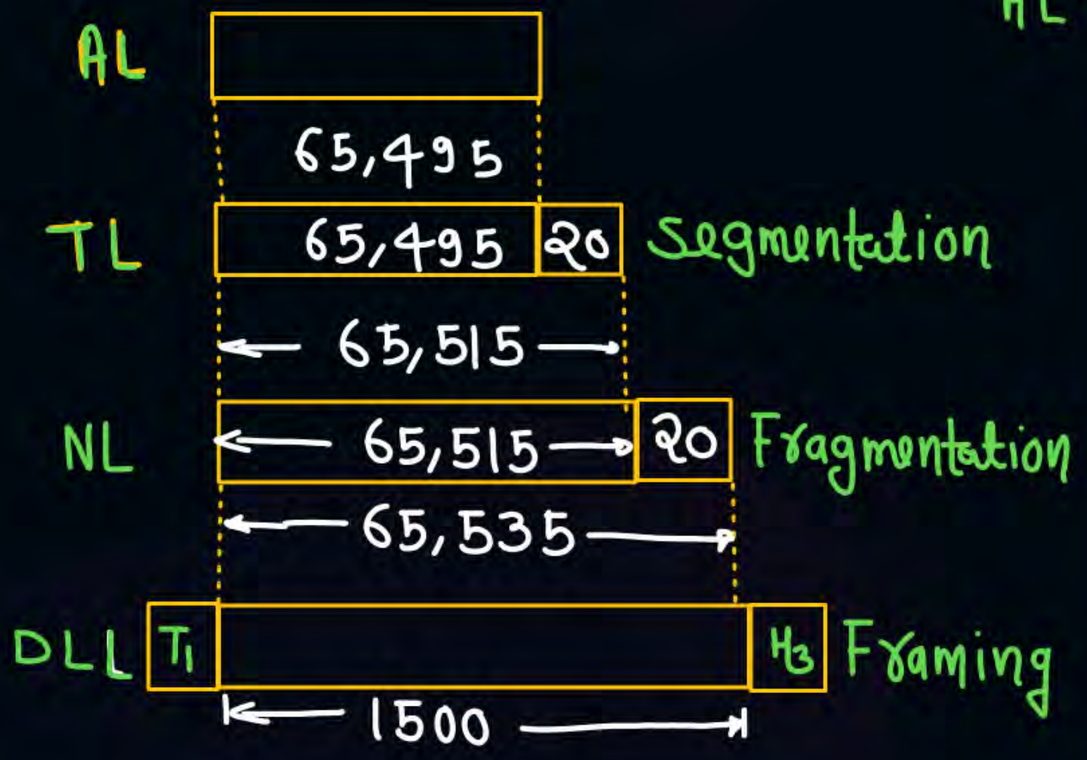
Maximum No. possible $= 2^{16} - 1$

$= 65,535$

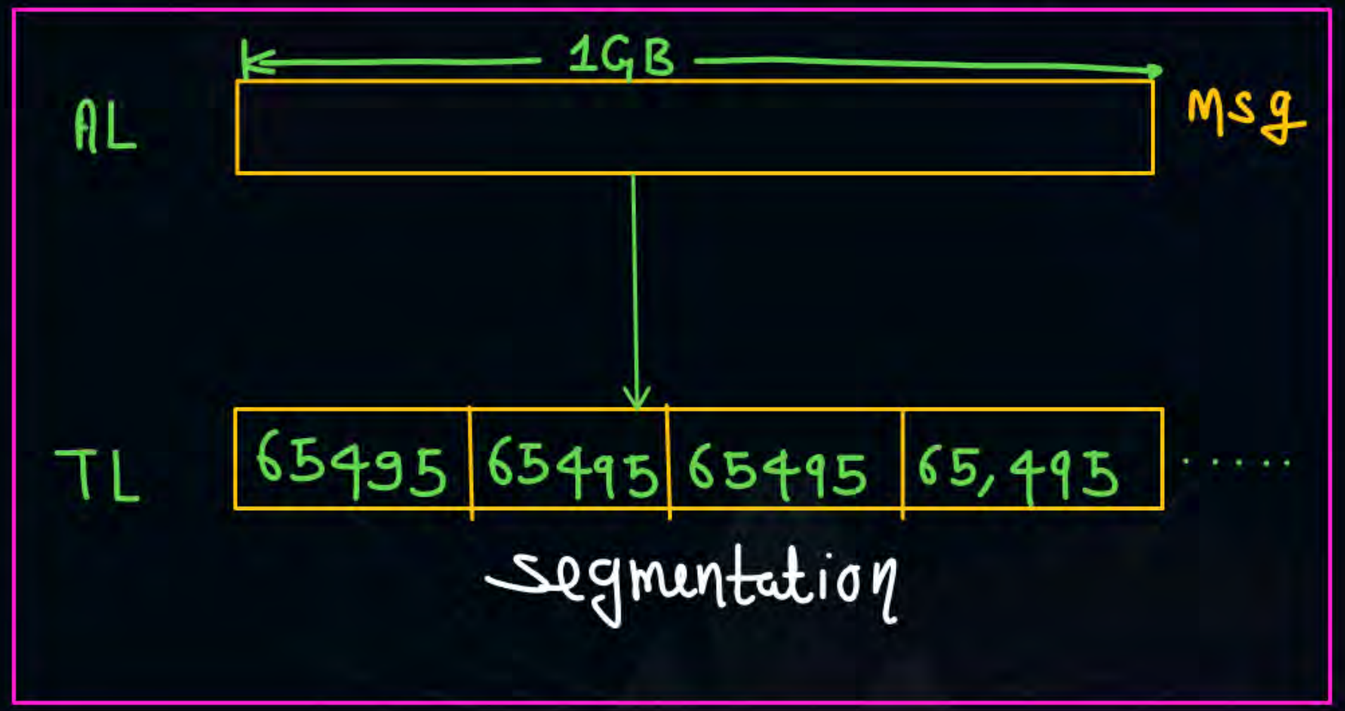
Total length = Data + Header

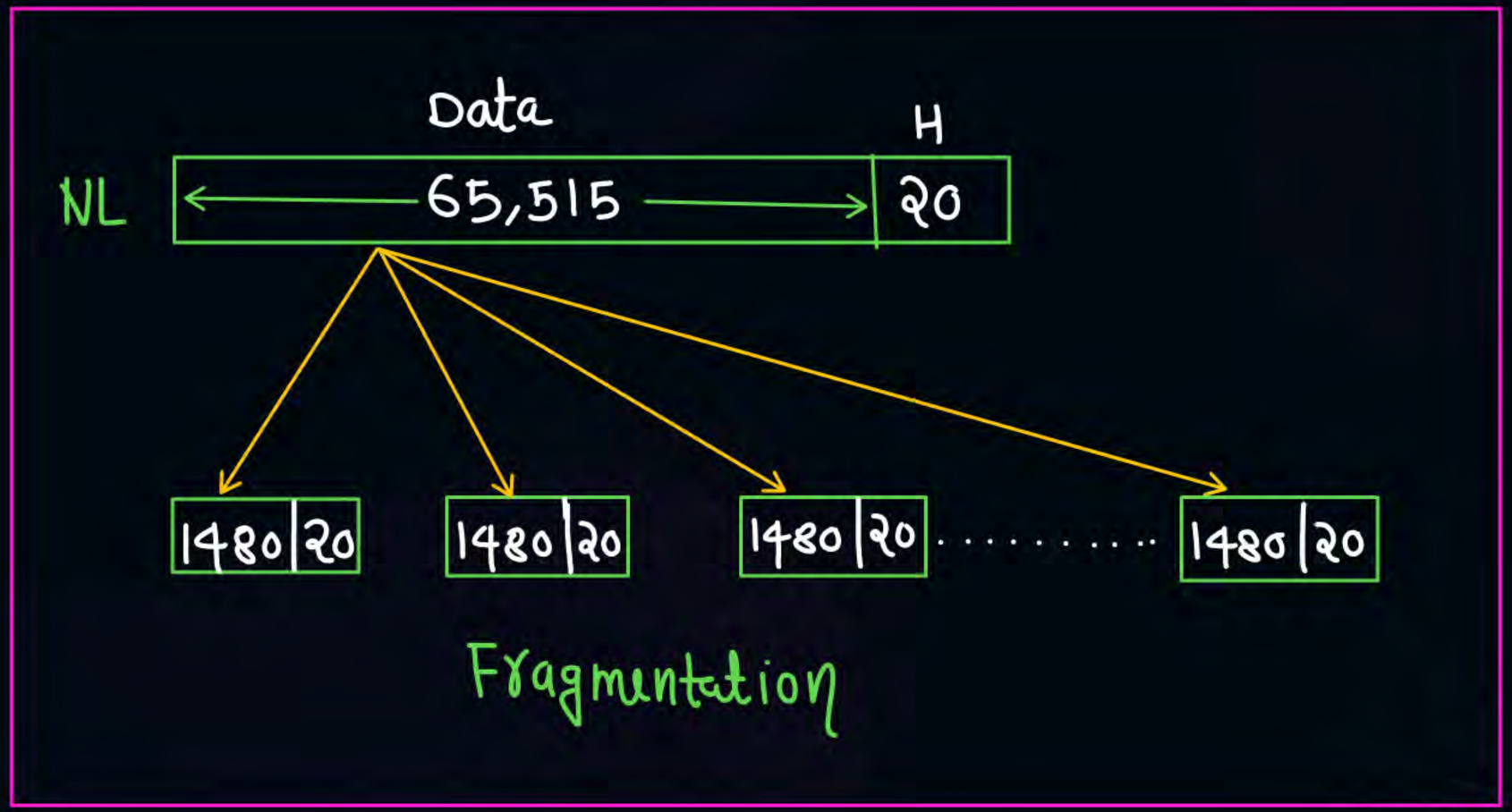


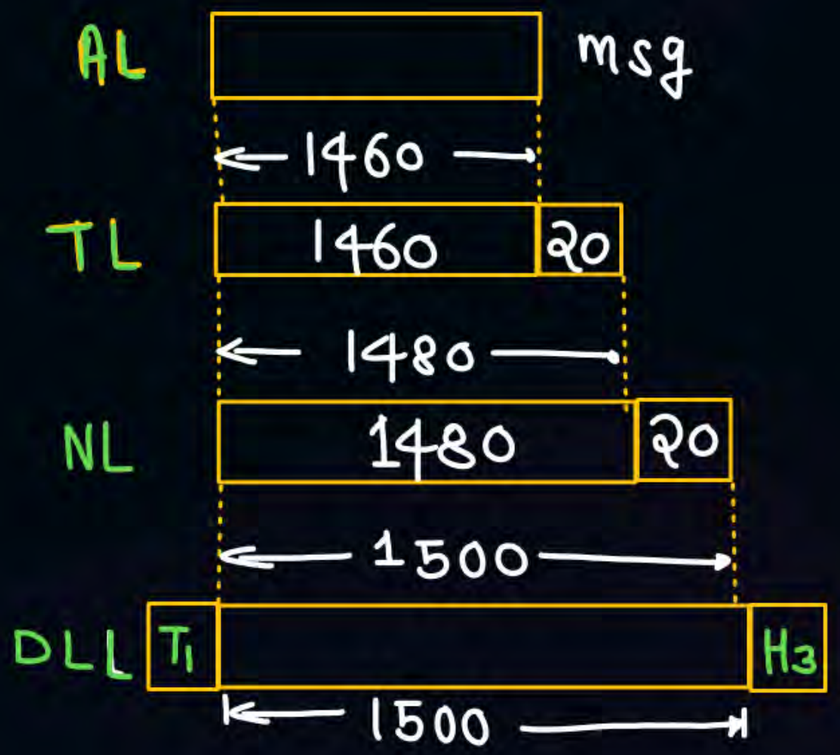




AL can pass any size of data to transport Layer




Data
H
NL
65,515
20
1480
20
1480
20
1480
20
1480
20
Fragmentation

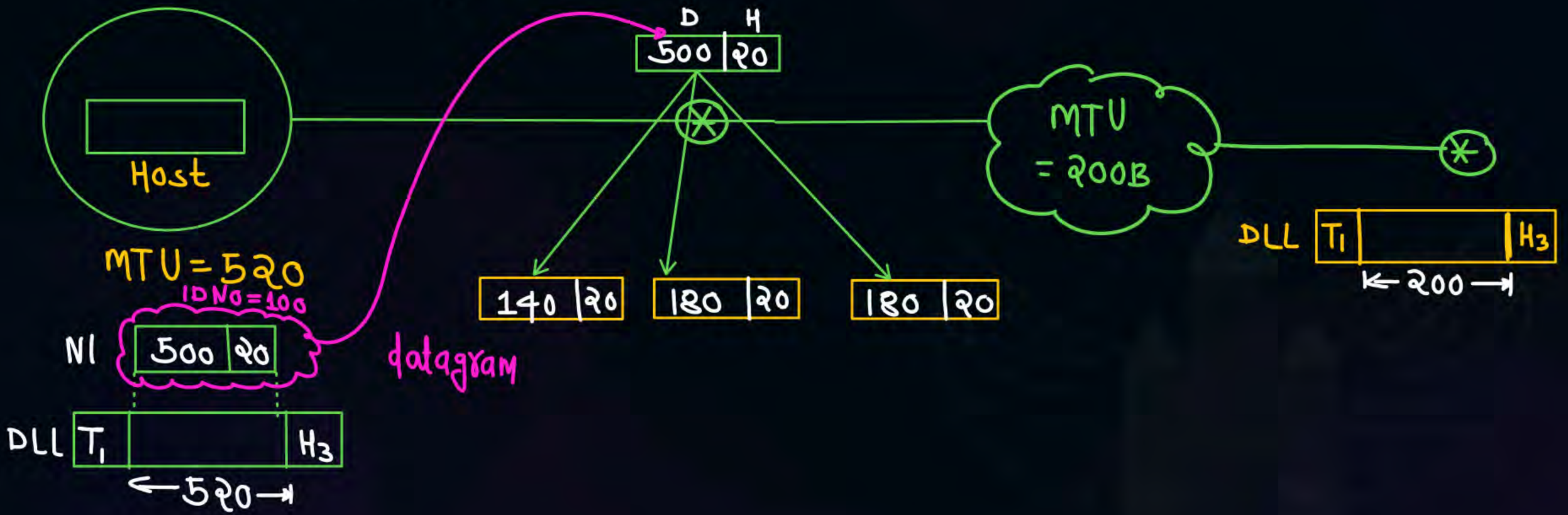

AL
msg
1460
TL
1460
20
1480
NL
1480
20
1500
DLL
T1
H3
1500

Source Host

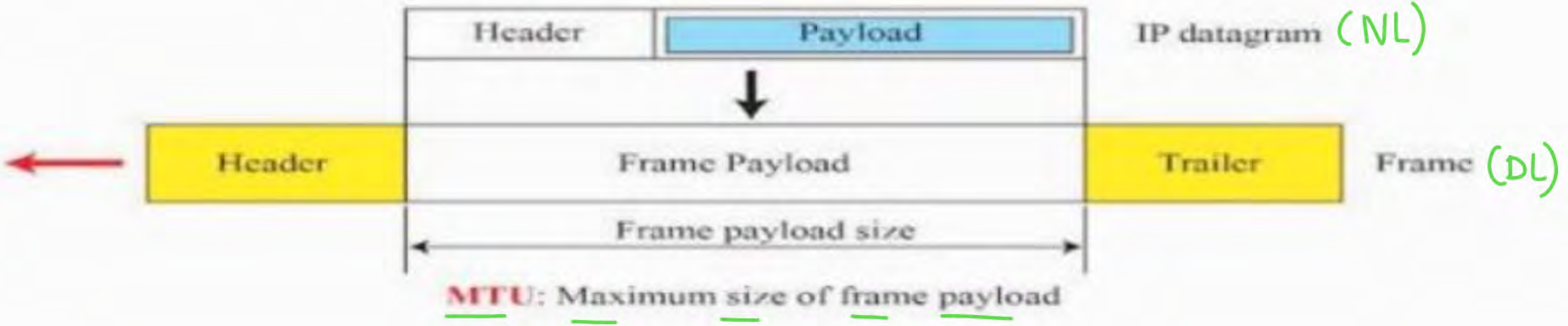

1GB
AL
TL
1460
1460
1460
1460
1460

Fragmentation at Router

MTU → maximum Amount of data that can be stored any data Link Layer Frame

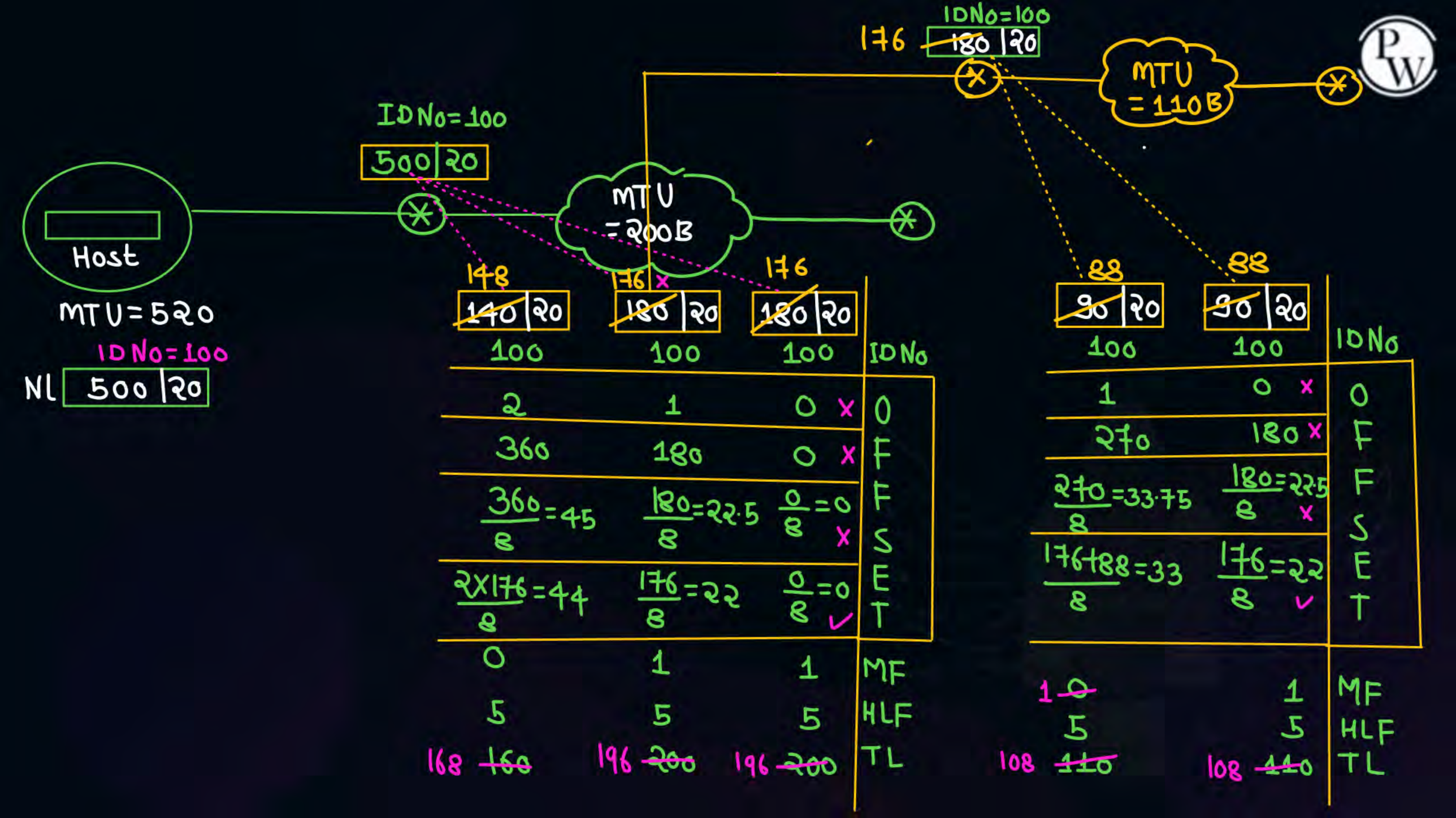


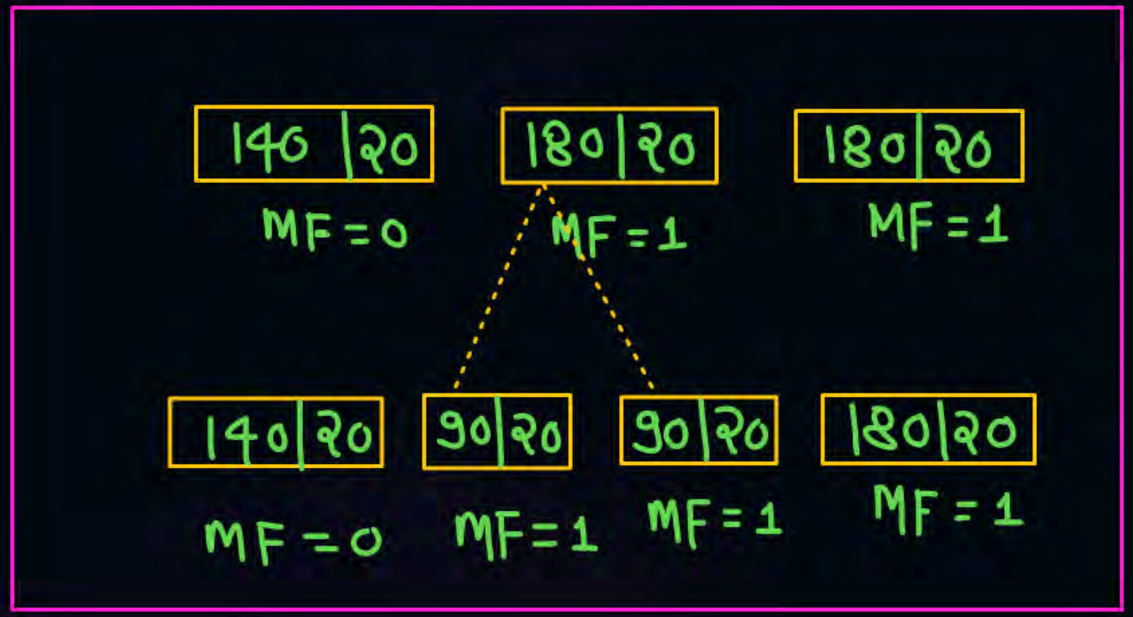
Maximum transfer unit (MTU)

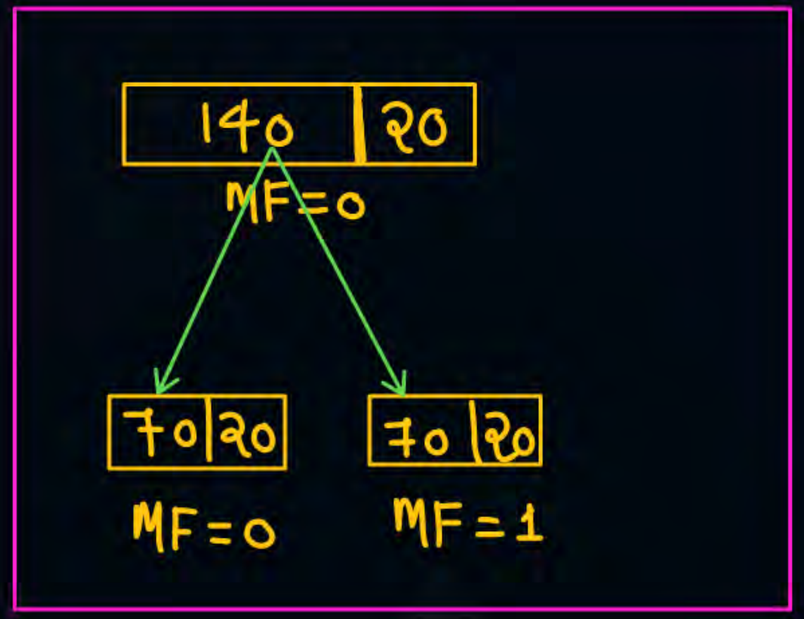


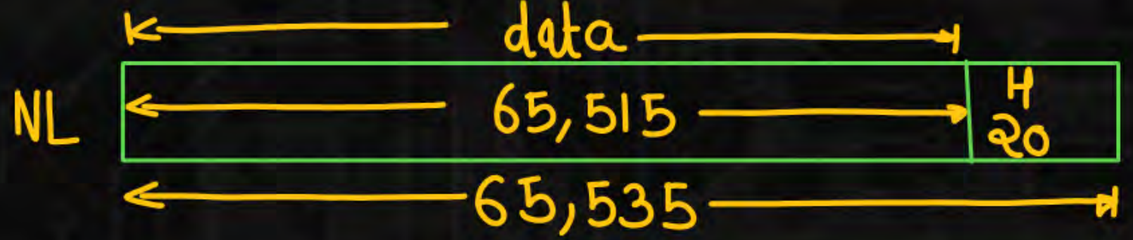
When a datagram is encapsulated in a frame, the total size of the datagram must be less than this maximum size of the frame payload.

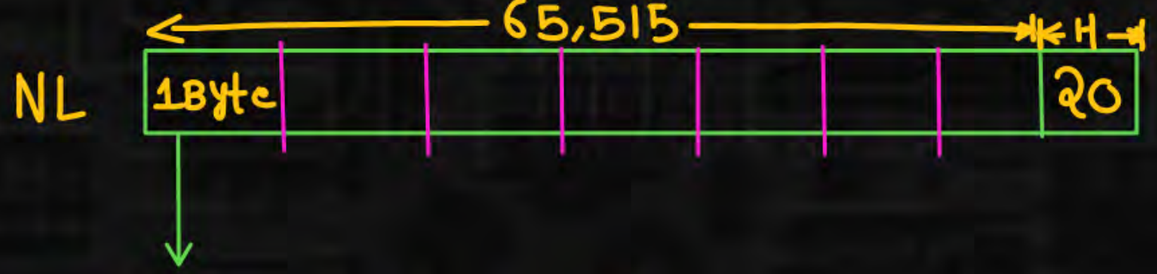
The value of the MTU differs from one physical network protocol to another. For example, the value for a LAN is normally 1500 bytes, but for a WAN it can be larger or smaller.


Host
MTU = 520
ID No = 100
Nl 500 | 20
ID No = 100
500 | 20
MTU = 200B
MTU = 110B
ID No = 100
176 180 | 20
148 140 | 20
176 180 | 20
176 180 | 20
88 90 | 20
88 90 | 20
100 100 100 ID No
2 1 0 x 0
360 180 0 x F
360/8 = 45 180/8 = 22.5 0/8 = 0 x F
2X176/8 = 44 176/8 = 22 0/8 = 0 ✓ S
0 1 1 MF E
5 5 5 HLF T
168 160 196 200 196 200 TL
100 100 ID No
1 0 x 0
270 180 x F
270/8 = 33.75 180/8 = 22.5 x F
176+88/8 = 33 176/8 = 22 ✓ S
1 0 1 MF E
5 5 5 HLF T
108 110 108 110 TL


140 | 20
MF = 0
180 | 20
MF = 1
180 | 20
MF = 1
140 | 20
MF = 0
90 | 20
MF = 1
90 | 20
MF = 1
180 | 20
MF = 1


140 | 20
MF = 0
70 | 20
MF = 0
70 | 20
MF = 1





OFFset = 65,514 $\cong 2^{16}$

$$(S.F) \rightarrow \frac{2^{16}}{1} = 2^{13}$$

$$(S.F) \rightarrow \frac{2^{16}}{8} = 2^{13}$$

Fragment OFFset = 13 bit

↓ maximum Value

$2^{13} - 1 = 8191 \cong 2^{13}$

Final Result

148 | 20	88 | 20	88 | 20	176 | 20	
100	100	100	100	IDNo
44	33	22	0	OFFset
0	1	1	1	MF
5	5	5	5	HLF
168	108	108	196	TL

OFFset = {22, 33, 0, 44}

↓ Increasing order

OFFset { 0, 22, 33, 44}

OFFset = {22, ~~33~~, 0, 44}

↓ Inc. order

OFFset = {0, 22, 44}

(i) | 176 | 20 |

$\frac{176}{8} = 22 = 2^{nd}$ Fragment OFFset Value

(ii) | 88 | 20 | | 176 | 20 |

$\frac{176+88}{8} = 33 = 3^{rd}$ Fragment OFFset Value

(iii) | 88 | 20 | | 88 | 20 | | 176 | 20 |

$\frac{176+88+88}{8} = 44 = 4^{th}$ Fragment OFFset Value

Reassembly

Algorithme

Reassemble Algorithm

If each fragment follow a different path and arrives out of order, the final destination host can reassemble the original datagram from the fragment received by using the following strategy:

1. Identify the fragment with offset = 0 and it is the first fragment.
2. Identify the fragment with MF = 0 and it is the last fragment.
3. Divide the data length of the first fragment by 8. The second fragment has an offset value equal to that result $\frac{176}{8} = 22 \rightarrow$ 2nd Fragment offset value
4. Divide the data length of the first and second fragment by 8. The third fragment has an offset value equal that result. $\frac{176+88}{8} = 33 \rightarrow$ 3rd Fragment offset value
5. Repeat this process as many times as possible to cover all the fragment.


THANK YOU
GW
SOLDIERS !