CS & IT ENGINEERING

Operating System

Memory Management

Lecture -2

By- Vishvadeep Gothi sir

Recap of Previous Lecture

Topic Memory Management

Topic Memory Management Technique

Topic Contiguous Memory Management Technique

Topics to be Covered

Topic Non-Contiguous MMT

Topic Paging

Topic Address Translation

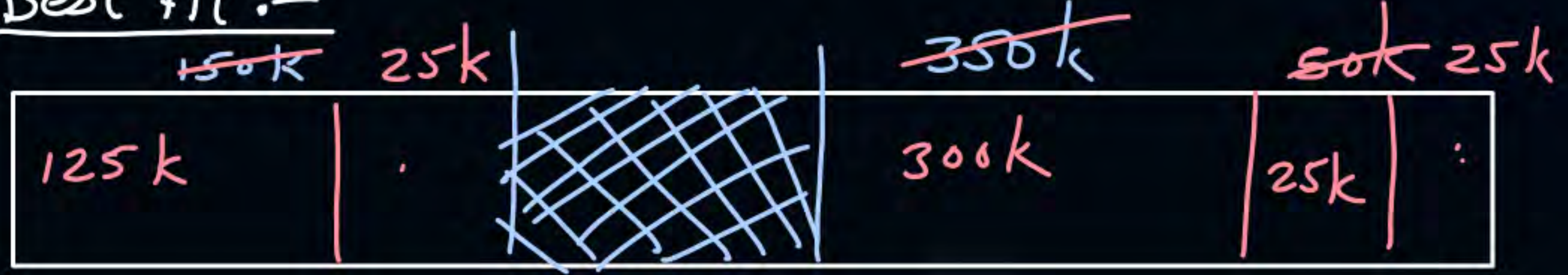
Fixed partition contiguous mmt ⇒ Best fit works better

variable ——— " ——————— " ——— ⇒ worst fits works better

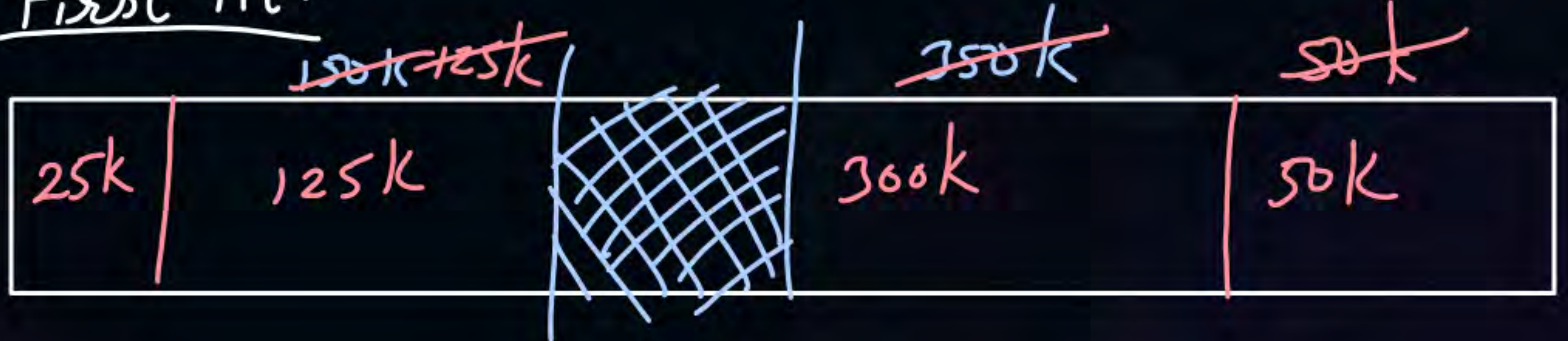
[MCQ]

#Q. Consider the requests from processes in given order 300K, 25K, 125K, and 50K. Let there be two blocks of memory available of size 150K followed by a block size 350K. Which of the following partition allocation schemes can satisfy the above requests?

Best fit:-



First fit:-



A. Best fit but not first fit

B. First fit but not best fit

C. Both First fit & Best fit

D. neither first fit nor best fit

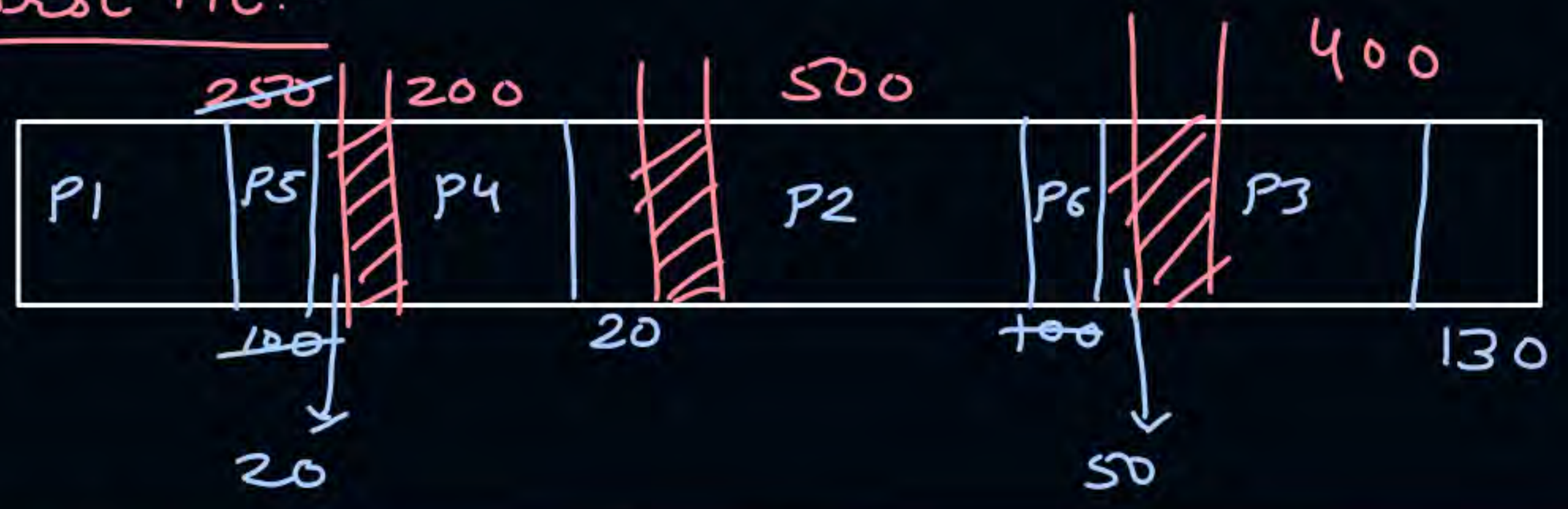
[NAT]

#Q. Consider variable partition MMT where there are 4 ~~partitions~~ holes of size 250MB, 200MB, 500MB and 400MB. The following process requests are made in the given order:

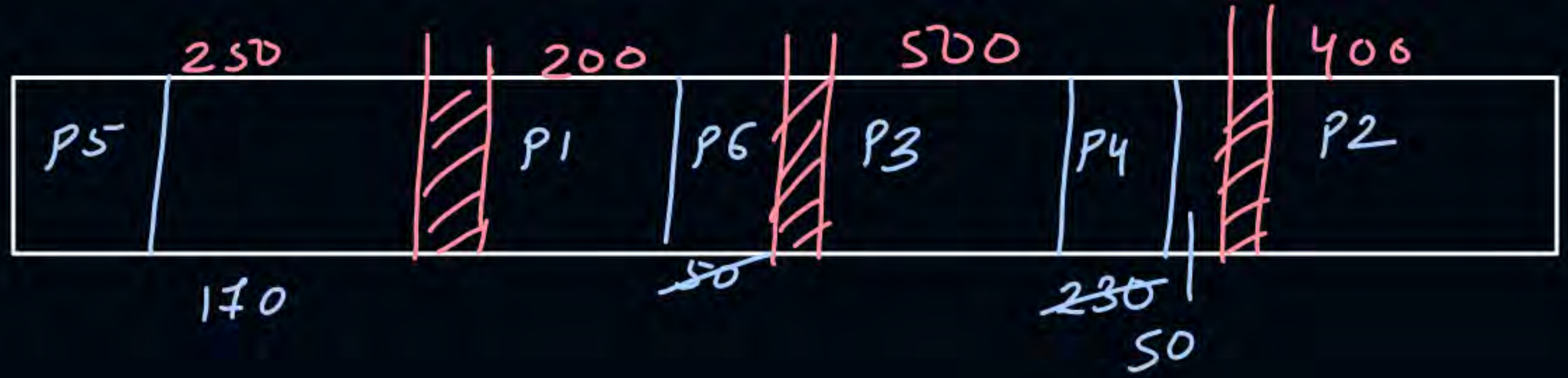
Process	Size
P1	150MB
P2	400MB
P3	270MB
P4	180MB
P5	80MB
P6	50MB

Provide how the processes are stored for First fit, Best fit and Worst Fit policies?

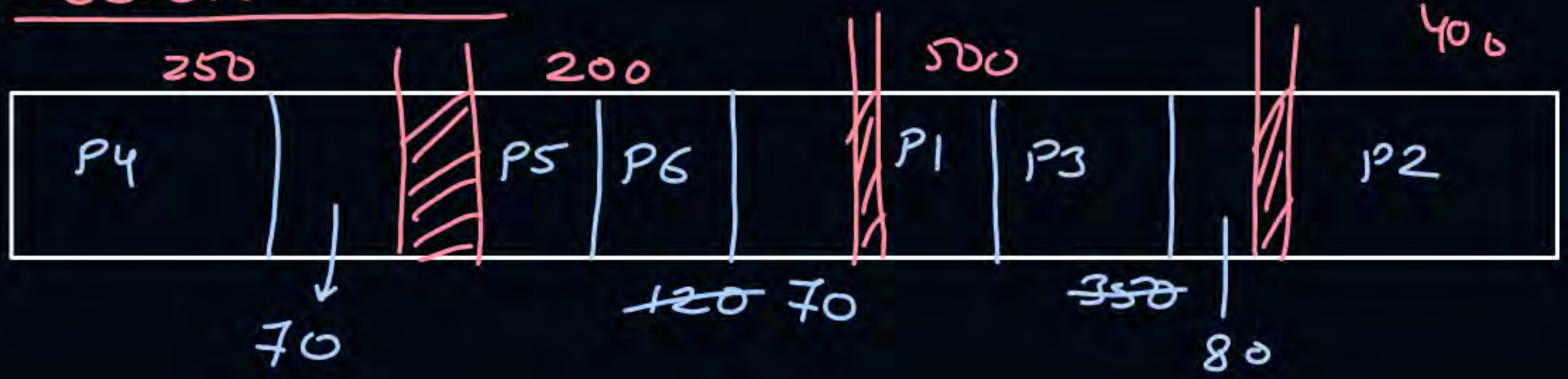
First fit:-



Best fit:-



3:- worst fit :-



Non-Contiguous MMT :-

Process is divided into partitions and each partition can be stored in memory anywhere.

Types

Paging

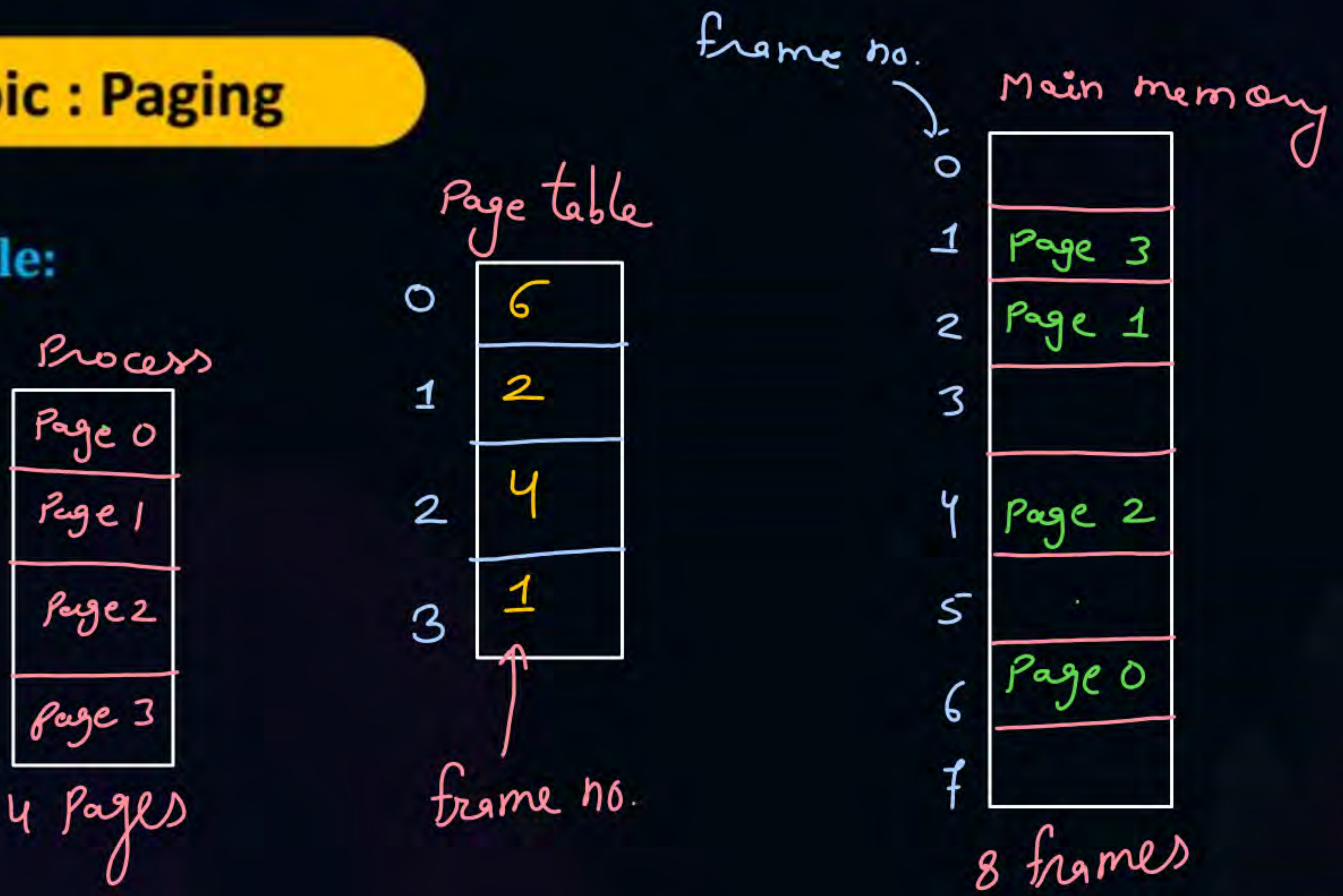
Segmentation

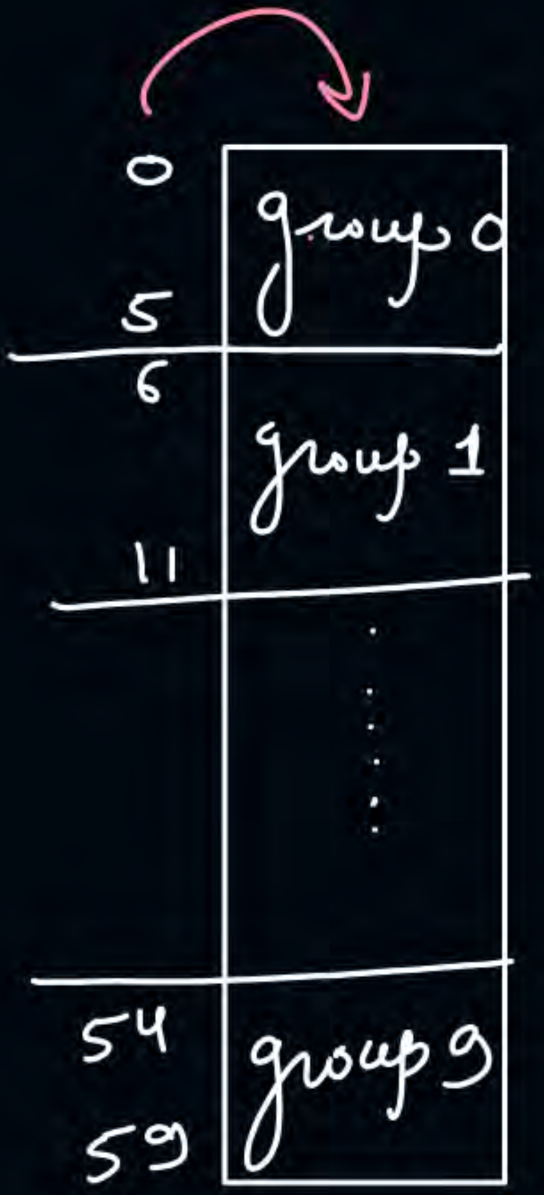
Topic : Paging

- Process is divided in equal size of pages 4B
- Physical memory is divided in same equal size of frames
  main
- Pages are scattered in frames

Topic : Paging

Example:





Roll no $\longrightarrow$ group no $\longrightarrow$ search in list $\longrightarrow$ Room no.

A page table is maintained to denote which page is stored in which frame.

$\boxed{\Rightarrow \text{No. of entries in a page table} = \text{no. of pages in the process}}$

$\boxed{\text{Page table entry size} = \text{frame no.} + \text{extra bits}}$

Topic : Paging

Consider

- A process has 4 pages
- Main memory has 8 frames

Process

00	Page 00
01	Page 01
10	Page 10
11	Page 11

Page table

00	110
01	010
10	100
11	001

Frame no.

Main memory

000	
001	Page 11
010	Page 01
011	
100	Page 10
101	
110	Page 00
111	

no. of bits needed for page no. $= \log_2$(no. of pages in process)

— 〃 — 〃 — frame no. $= \log_2$(no. of frames in memory)

ex:-

no. of pages	page no.
4	2-bits
8	3-bits
16	4-bits
$128k = 2^{17}$	17 bits
2^{23}	23 bits

Assume,

Process $\Rightarrow$ 4 pages

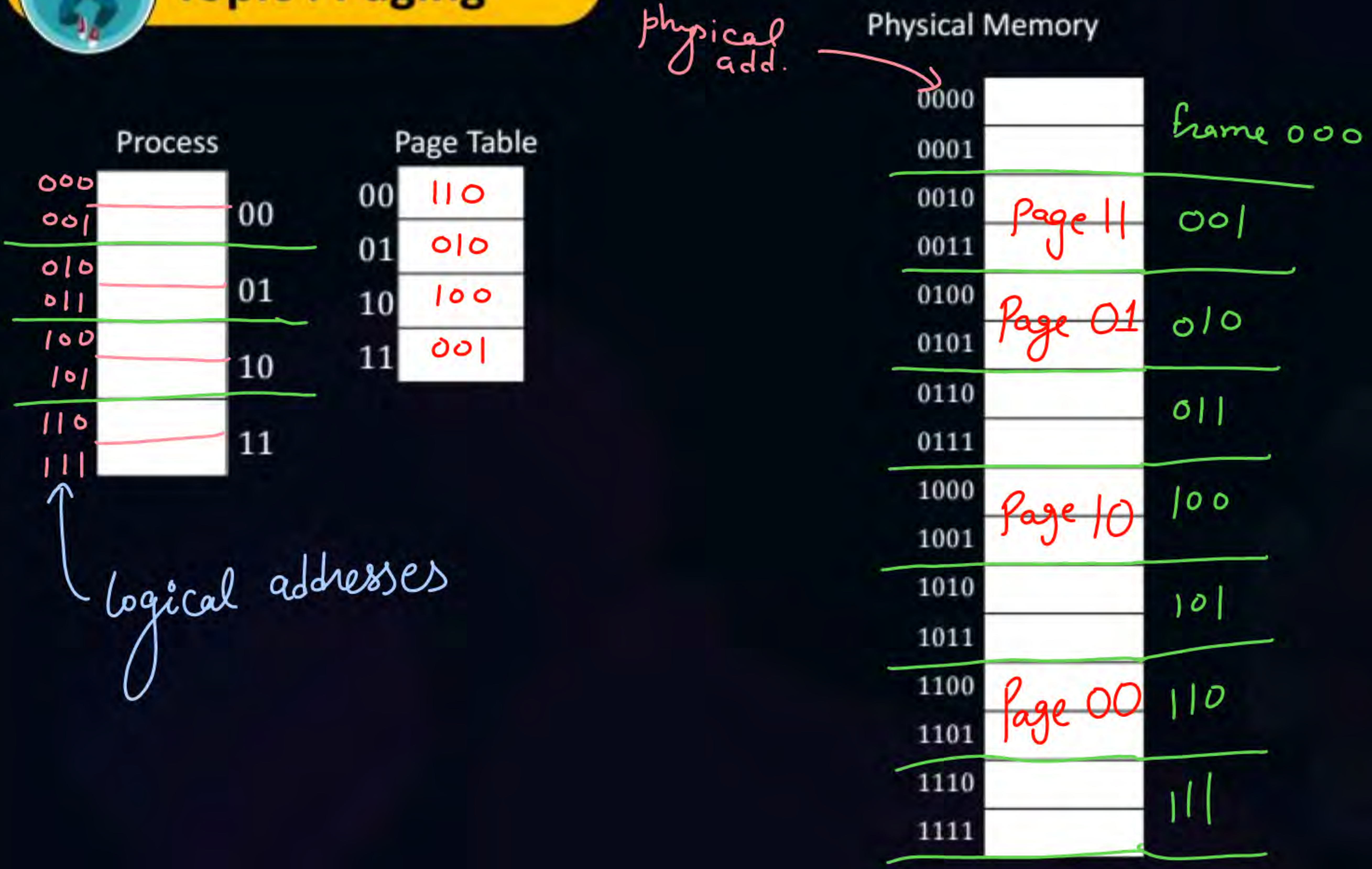
memory $\Rightarrow$ 8 frames

Page size $\Rightarrow$ 2 bytes

Process size = 8 bytes = 2^3 B $\Rightarrow$ logical add. = 3-bits

memory size = 16 bytes = 2^4 B $\Rightarrow$ physical add. = 4-bits

Topic : Paging



Page Table

00	110
01	010
10	100
11	001

CPU generates logical add.

↓

find page no. 'p' to which logical address belongs

↓

search in page table for page no. p

↓

frame no. 'f'

↓

goto frame no. f and get the content

Logical address is divided into 2 parts

Page no.	byte no. of the page
(p)	(d) (offset)

example:-

Logical address $\Rightarrow$ 011

01	1
p	d

no. of bits for d $= \log_2$(page size in bytes)

Page size	d
$16\ B = 2^4\ B$	4 bits
$2KB = 2^{11}\ B$	11 bits

physical add. is divided into 2 parts

frame no.	byte no. of the page
(f)	(d) offset

<u>example:-</u>

physical add. => 1011

101	1
↓ f	↓ d

2 mins Summary

Topic Non-Contiguous MMT

Topic Paging

Topic Address Translation

Happy Learning

THANK - YOU