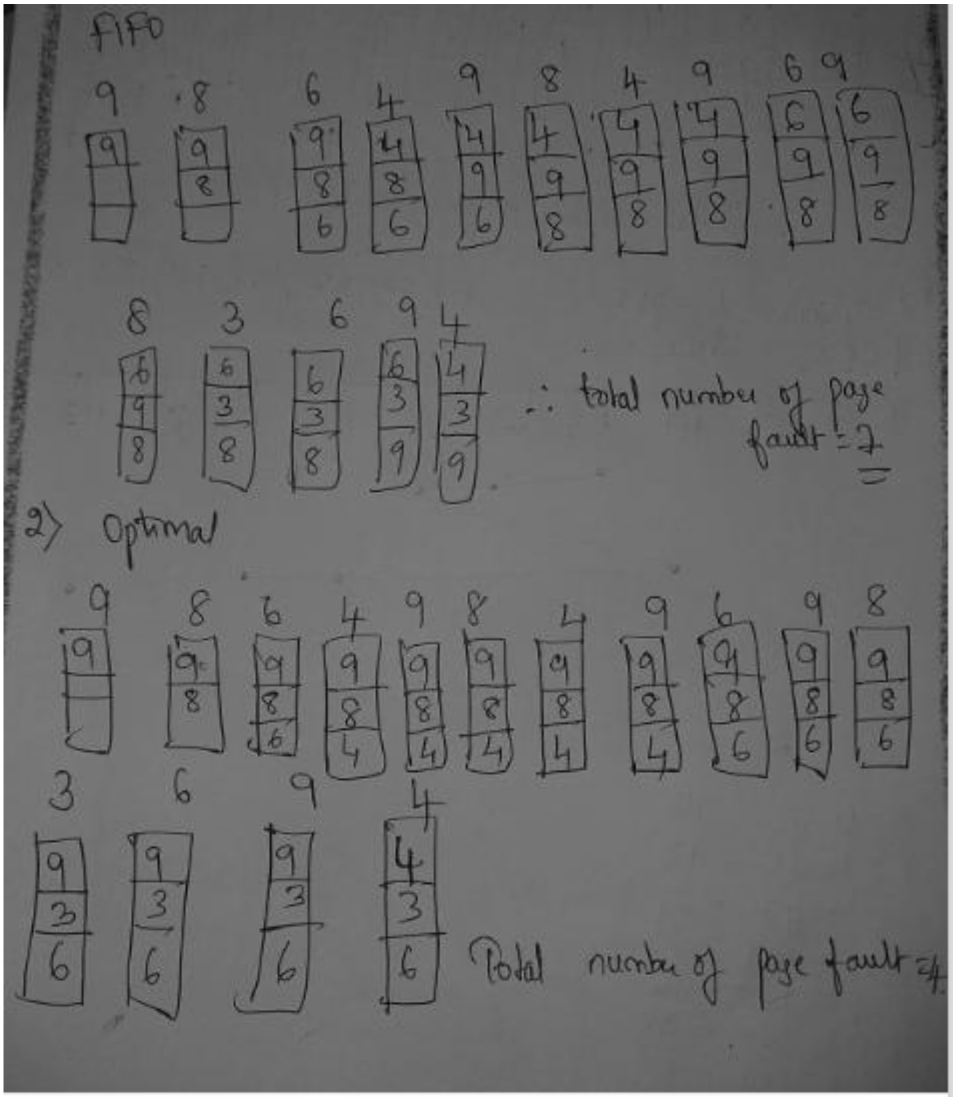
INTERNAL ASSESSMENT TEST 3 – August 2021
SOLUTION

Sub:	**OPERATING SYSTEMS**					Sub Code:	17CS64	Branch: **ISE & CSE**
Date:	22-06-2021	Duration:	90 min’s	Max Marks:	50	Sem / Sec:	VI	OBE

Answer any 5 Questions (5 X 10 = 50)

1 (a)	What do you mean by Page fault? Describe the steps in handling page faults. If there is ever a reference to a page, first reference will trap to OS ⇒ page fault 1. OS looks at internal table in PCB to decide: Invalid reference ⇒ abort. Valid but Just not in memory. 3. Get empty frame.- check free frame list 4. Swap page into frame. 5. Reset tables, internal table in PCB and page table validation bit = 1. 6. Restart instruction:	8	CO 1	2
(b)	. What is Belady’s anamoly? Belady’s anamoly: The hit ratio is decreasing in place of increasing although we have increased the frame size. This unusual behavior is only observed sometimes. It doesn't mean that every time the frame size is increased the page faults will increase.	2	CO 1	1

2 (a)	What is the need of Page replacement Algorithms? For the following page reference string 9, 8, 6, 4, 9, 8, 4, 9, 6, 9, 8, 3, 6, 9, 4 calculate the number of page faults using following page replacement algorithms with 3 frames. 1. First In First Out 2. Optimal 3. Least Recently Used	10	CO1	3



FIFO

9	8	6	4	9	8	4	9	6	9
9	9	9	4	4	4	4	4	6	6
	8	8	8	9	9	9	9	9	9
		6	6	6	8	8	8	8	8

8	3	6	9	4
6	6	6	6	4
9	3	3	3	3
8	8	8	9	9

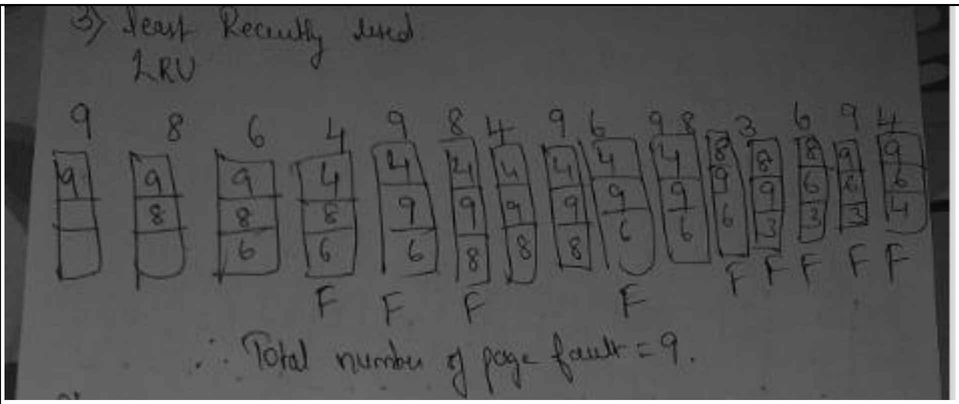
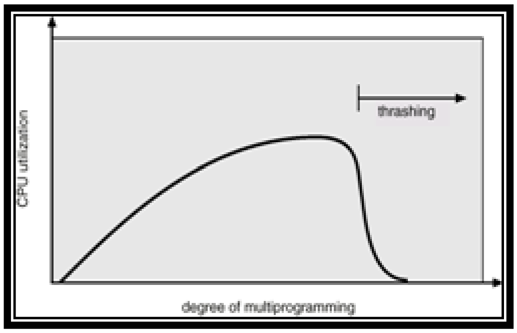
∴ total number of page fault = 7

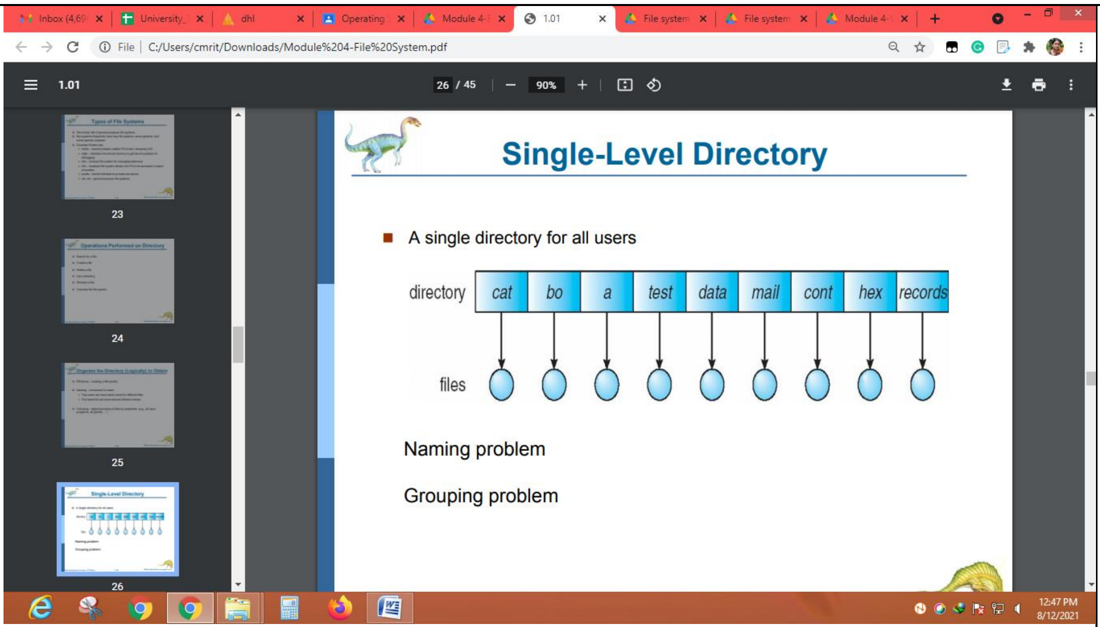
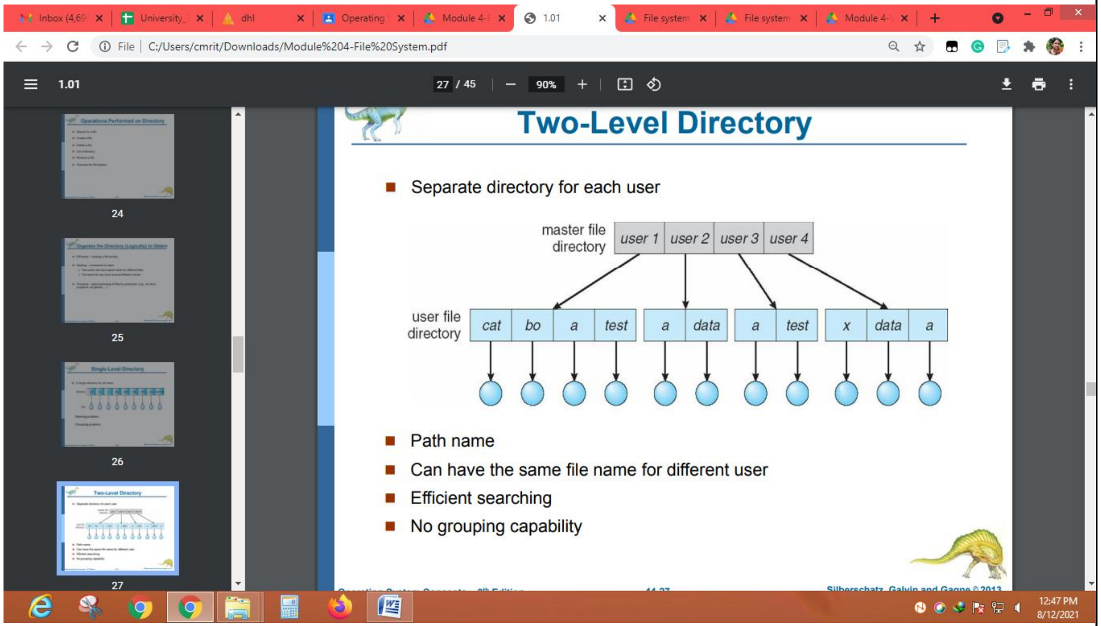
2) Optimal

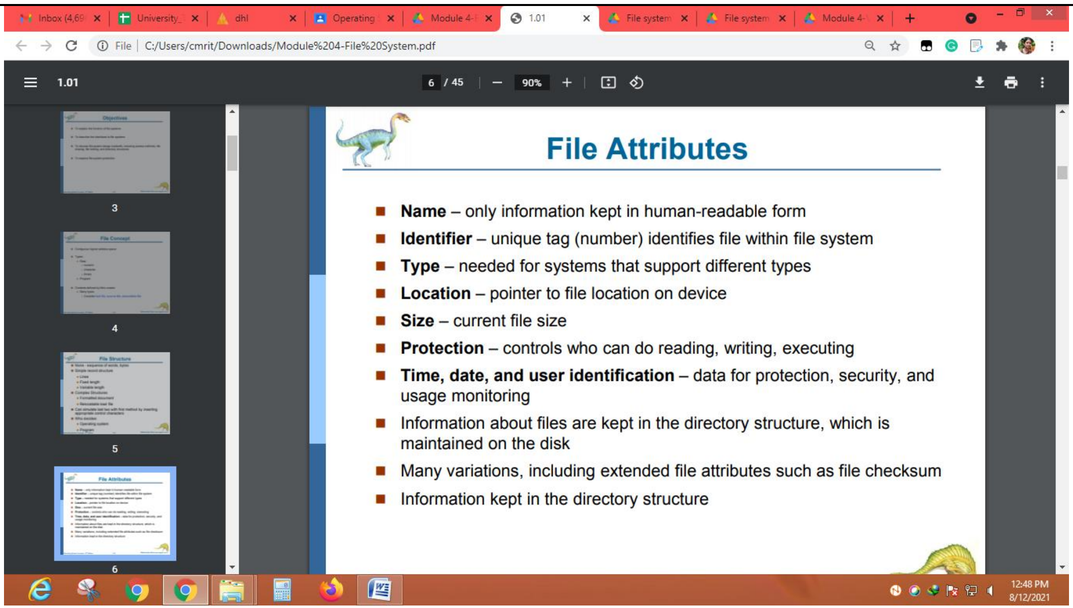
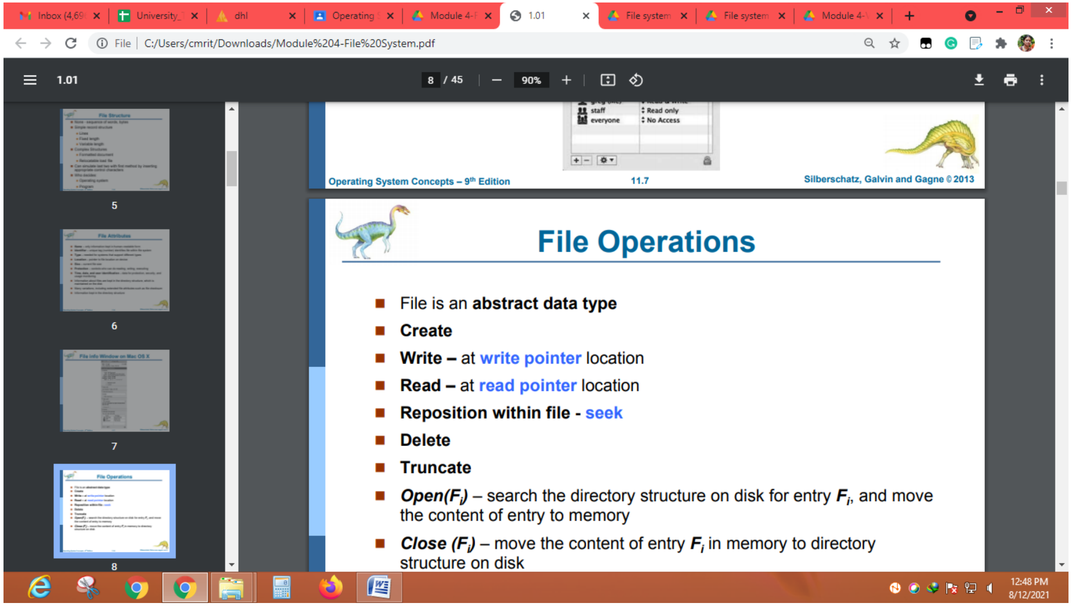
9	8	6	4	9	8	4	9	6	9	8
9	9	9	9	9	9	9	9	9	9	9
	8	8	8	8	8	8	8	8	8	8
		6	4	4	4	4	4	6	6	6

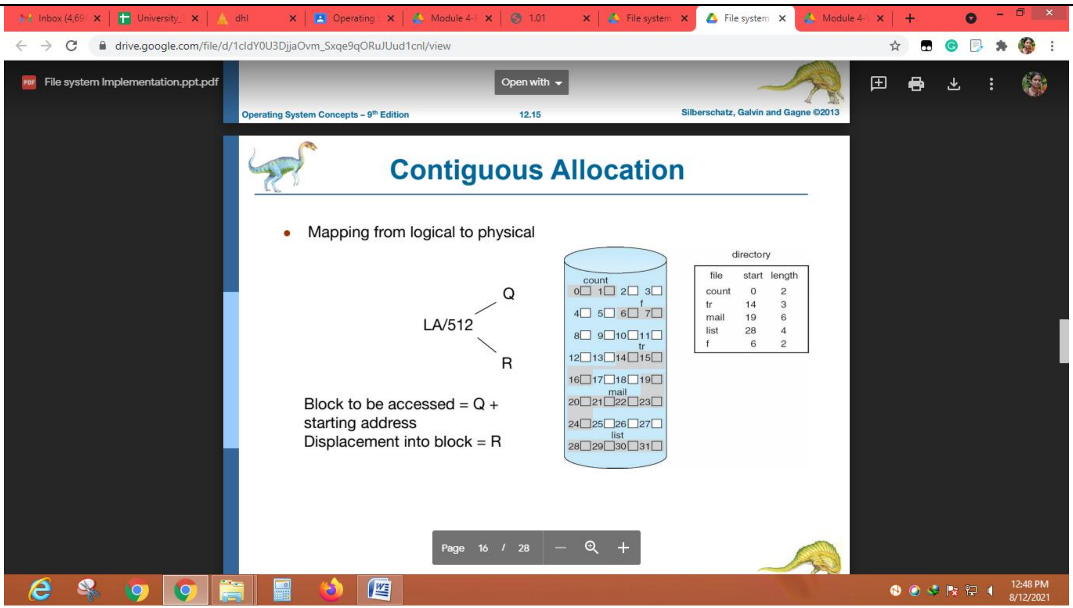
3	6	9	4
9	9	9	4
3	3	3	3
6	6	6	6

Total number of page fault =4

	 3) Least Recently used LRU 9 8 6 4 9 8 4 9 6 9 8 3 6 9 4 9 | 9 8 | 9 8 6 | 4 8 6 | 4 9 6 | 4 9 8 | 4 9 8 | 4 9 8 | 4 9 6 | 4 9 6 | 8 9 6 | 8 9 3 | 8 6 3 | 9 8 3 | 9 6 4 F F F F F F F F F ∴ Total number of page fault = 9.			
3 (a)	What is thrashing? Explain the cause of thrashing?  ● If a process does not have “enough” pages, the page-fault rate is very high. This leads to: ● low CPU utilization. ● operating system thinks that it needs to increase the degree of multiprogramming. ● another process added to the system. ● **Thrashing** ≡ a process is busy swapping pages in and out.	4	CO 1	2
(b)	Explain the Single level and Two level Directory structure. Compare both methods with respect to merits and demerits.	6	CO 1	2

	 Inbox (4,69 University dhl Operating Module 4- 1.01 File system File system Module 4- File | C:/Users/cmrit/Downloads/Module%204-File%20System.pdf 1.01 26 / 45 90% 23 24 25 26 12:47 PM 8/12/2021 **Single-Level Directory** ■ A single directory for all users directory cat bo a test data mail cont hex records files Naming problem Grouping problem  Inbox (4,69 University dhl Operating Module 4- 1.01 File system File system Module 4- File | C:/Users/cmrit/Downloads/Module%204-File%20System.pdf 1.01 27 / 45 90% 24 25 26 27 12:47 PM 8/12/2021 **Two-Level Directory** ■ Separate directory for each user master file directory user 1 user 2 user 3 user 4 user file directory cat bo a test a data a test x data a ■ Path name ■ Can have the same file name for different user ■ Efficient searching ■ No grouping capability Silberschatz, Galvin and Gagne ©2013			
4 a)	What is File? List all the attributes of a file and the operations that are performed on files.	4	CO 1	1

	 C:/Users/cmrit/Downloads/Module%204-File%20System.pdf 1.01 6 / 45 90% **File Attributes** ■ **Name** – only information kept in human-readable form ■ **Identifier** – unique tag (number) identifies file within file system ■ **Type** – needed for systems that support different types ■ **Location** – pointer to file location on device ■ **Size** – current file size ■ **Protection** – controls who can do reading, writing, executing ■ **Time, date, and user identification** – data for protection, security, and usage monitoring ■ Information about files are kept in the directory structure, which is maintained on the disk ■ Many variations, including extended file attributes such as file checksum ■ Information kept in the directory structure  C:/Users/cmrit/Downloads/Module%204-File%20System.pdf 1.01 8 / 45 90% staff Read only everyone No Access Operating System Concepts – 9th Edition 11.7 Silberschatz, Galvin and Gagne ©2013 **File Operations** ■ File is an **abstract data type** ■ **Create** ■ **Write –** at write pointer location ■ **Read –** at read pointer location ■ **Reposition within file - seek** ■ **Delete** ■ **Truncate** ■ ***Open(F_i)*** – search the directory structure on disk for entry F_i, and move the content of entry to memory ■ ***Close (F_i)*** – move the content of entry F_i in memory to directory structure on disk			
b)	With supporting diagrams, explain contiguous and Linked File allocation methods.	6	CO 1	2



Operating System Concepts – 9th Edition 12.15 Silberschatz, Galvin and Gagne ©2013

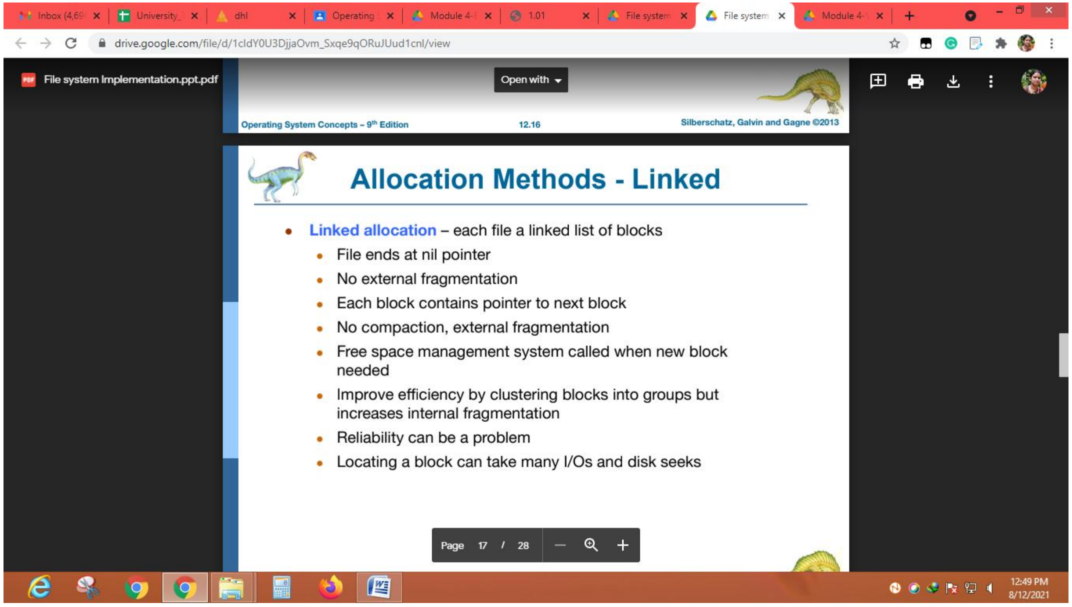
Contiguous Allocation

- Mapping from logical to physical

Block to be accessed = Q + starting address
Displacement into block = R

directory

file	start	length
count	0	2
tr	14	3
mail	19	6
list	28	4
f	6	2

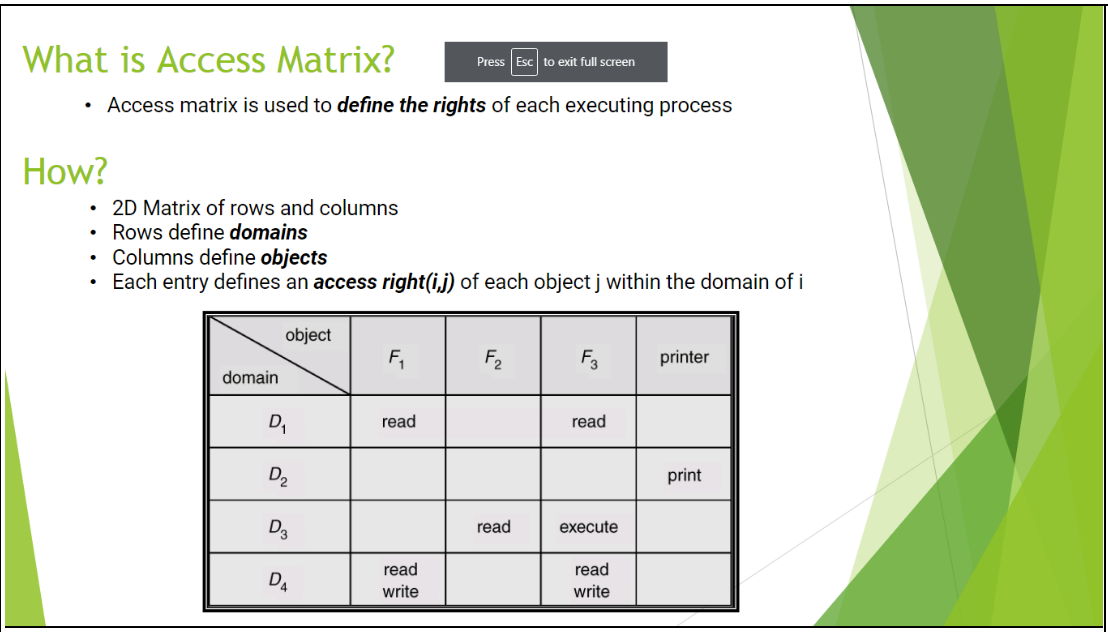


Operating System Concepts – 9th Edition 12.16 Silberschatz, Galvin and Gagne ©2013

Allocation Methods - Linked

- Linked allocation – each file a linked list of blocks
  - File ends at nil pointer
  - No external fragmentation
  - Each block contains pointer to next block
  - No compaction, external fragmentation
  - Free space management system called when new block needed
  - Improve efficiency by clustering blocks into groups but increases internal fragmentation
  - Reliability can be a problem
  - Locating a block can take many I/Os and disk seeks

5 (a)	What is Access Matrix? Explain the access matrix model of implementing protection in Operating System.	8	CO 1	2



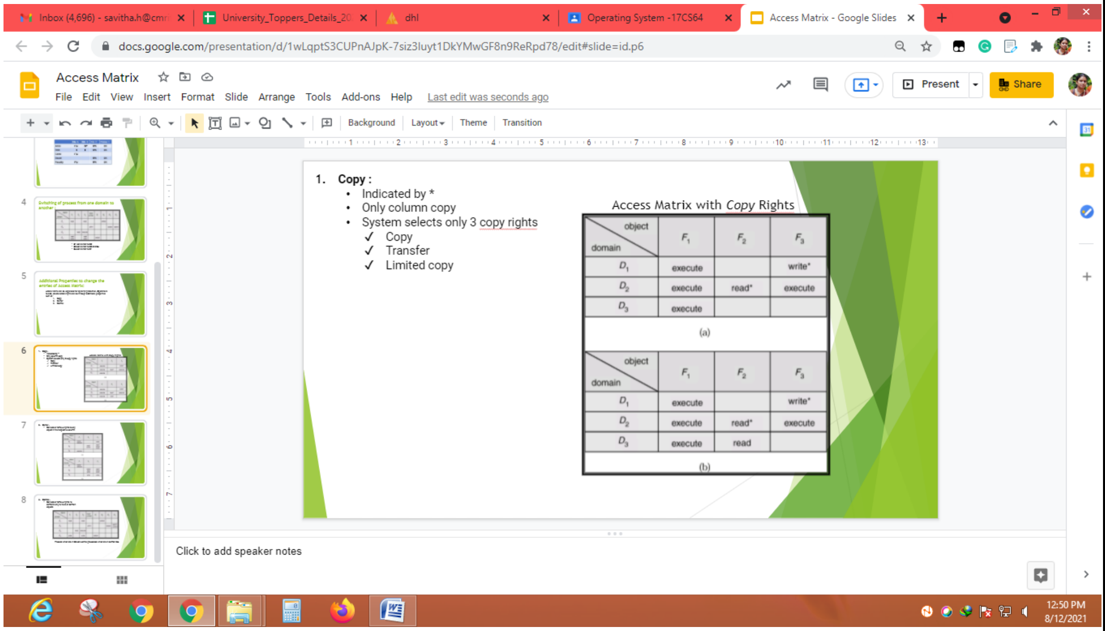
What is Access Matrix?

- Access matrix is used to ***define the rights*** of each executing process

How?

- 2D Matrix of rows and columns
- Rows define ***domains***
- Columns define ***objects***
- Each entry defines an ***access right(i,j)*** of each object j within the domain of i

domain \ object	F_1	F_2	F_3	printer
D_1	read		read	
D_2				print
D_3		read	execute	
D_4	read write		read write	


Inbox (4,696) - savitha.h@cmr
University_Toppers_Details_20
dhl
Operating System -17CS64
Access Matrix - Google Slides
docs.google.com/presentation/d/1wLqptS3CUPnAJpK-7siz3luyt1DkYMwGF8n9ReRpd78/edit#slide=id.p6
Access Matrix
File Edit View Insert Format Slide Arrange Tools Add-ons Help Last edit was seconds ago
Present
Share
Background Layout Theme Transition
Click to add speaker notes
12:50 PM
8/12/2021

1. **Copy :**
   - Indicated by *
   - Only column copy
   - System selects only 3 copy rights
     - ✓ Copy
     - ✓ Transfer
     - ✓ Limited copy

Access Matrix with *Copy* Rights

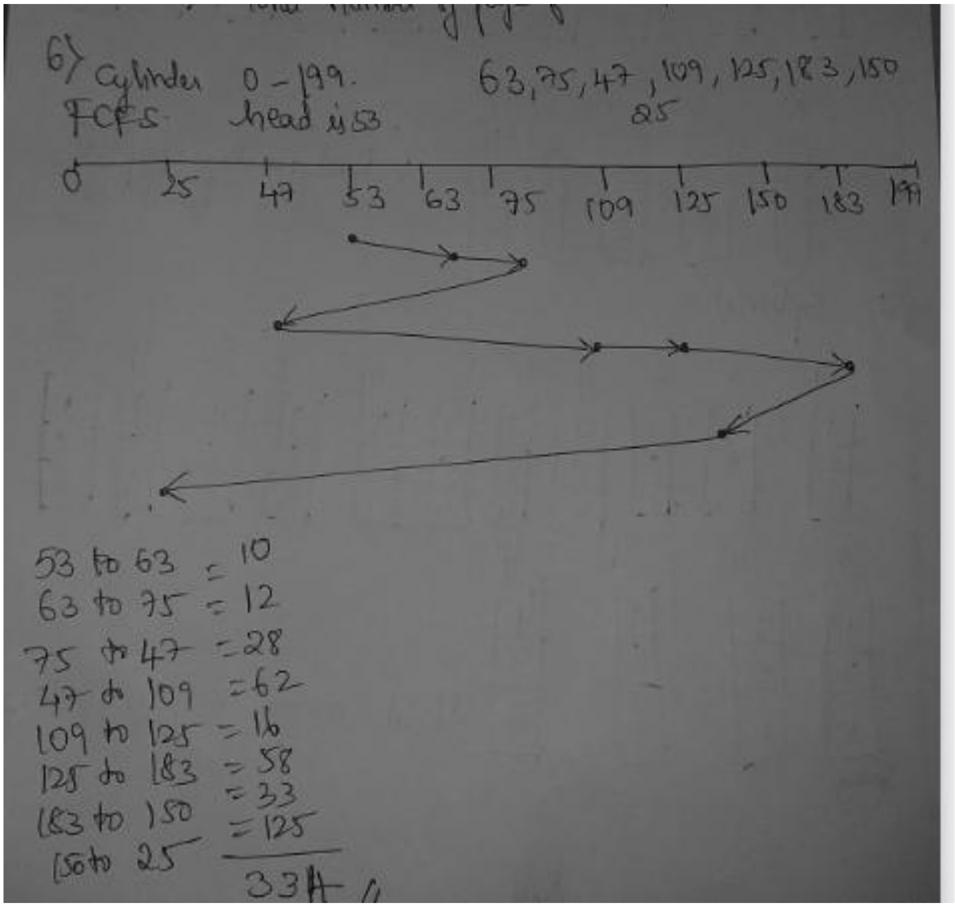
domain \ object	F_1	F_2	F_3
D_1	execute		write*
D_2	execute	read*	execute
D_3	execute		

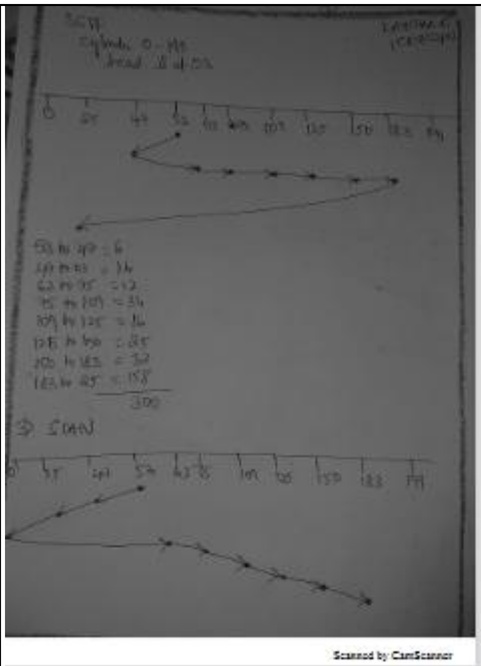
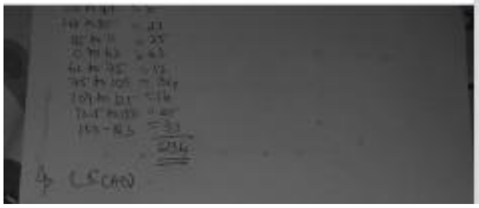
(a)

domain \ object	F_1	F_2	F_3
D_1	execute		write*
D_2	execute	read*	execute
D_3	execute	read	

(b)

(b)	What are the types of failures that may happen in Disk? Type of Failure: • Disk needs to be replaced if there is complete failure. • More frequently, one or more sectors become defective. • Some times, few disk will come with defective sectors from factory.	2	CO1	1

6	Consider a disk queue with requests for I/O to blocks on cylinders 63, 75, 47, 109, 125, 183, 150, 25. The head is initiallyat cylinder number 53 and its previous serviced request was 45. The cylinders are numbered from 0 to 199. Calculate the total head movement (innumber of cylinders) incurred while servicing these requests is by applying following algorithms. 1. FCFS 2. SSTF 3. SCAN 4. C SCAN.  6) Cylinder 0 – 199. 63,75,47,109,125,183,150 25 FCFS head is 53 53 to 63 = 10 63 to 75 = 12 75 to 47 = 28 47 to 109 = 62 109 to 125 = 16 125 to 183 = 58 183 to 150 = 33 150 to 25 = 125 334	10	CO 1	3

	 SSTF Cylinder 0 - 199 Head at 53 LAKSHMI K [illegible] 0 25 49 53 62 75 109 125 150 183 199 53 to 49 = 4 49 to 62 = 13 62 to 75 = 13 75 to 109 = 34 109 to 125 = 16 125 to 150 = 25 150 to 183 = 33 183 to 25 = 158 300 3) SCAN 0 25 49 53 62 75 109 125 150 183 199 Scanned by CamScanner  53 to 49 = 4 49 to 25 = 24 25 to 0 = 25 0 to 62 = 62 62 to 75 = 13 75 to 109 = 34 109 to 125 = 16 125 to 150 = 25 150 - 183 = 33 236 4) C-SCAN			