Assignment 2

Date / /

Page No.

Name ⇒ Aditya singh
Roll NO ⇒ 230101022g

Q1 Ans logical (virtual) address are translated into physical address using MMU (memory management unit)

- logical Address → divided into page number + off set
- page Number → mapped using page Table to frame number
- Physical Address = Frame Number + offset

Q2 Ans Internal & External Fragmentation
- Internal Fragment · A 100KB position used by a 90KB process → 10KB wasted
- External Fragmented Free memory exist but in scattered blocks

Techniques
- paging
- segmentation
- Building system allocation

Q3 ~~...~~ memory · divided into fixed-sized pages
- process allocated non-contaguous frames

Date / /
Page No.

Trade off : overhead - Pages consumes memory.

- speed : Address translation source (solved by TLB)
- Fragmentat: Eliminates external but causes internal fragmentation within last page

Q4 OS - Hardware Interaction (virtual memory)

- Hardware support.

→ Page Table → stores mappings
→ ~~tat~~ TLB (Translation looking buffer) - speed up translation

→ MMU → Perfoms translation
ex - Accessing page not in RAM - OS triggers page fault, loads pages from disk.

Q5: virtual Address = 16 bits → Address space
2^{16} = 65,536 bytes

page size = 1KB = 1024 bytes 2^{10}
virtual pages = $2^{16} / 2^{10} = 2^6$ = 64 pages
each entry = 2 bytes
pages Table size = 64×2 = 128 bytes

Date / /
Page No.

Q6 $P_1 = 212$ KB $P_2 = 417$ KB $P_3 = 112$ KB
$P_4 = 426$ KB

Step	Action/Algo Rule	Allocated Block(s)	Remaining
0	Start		1000
1	Allocate $P_1 = 212$	$P1 \rightarrow 212$	$1000-212=788$
2	Allocate $P_2 = 417$	$P_1 \rightarrow 212$ $P_2 \rightarrow 417$	$788-417=371$
3	Allocate $P_3 = 112$	$P_1 \rightarrow 212$ $P_2 - 417$ $P_3 - 112$	$371-112=259$
4	Try Allocate $P_4 = 420$	P_4 cannot fit	free 259

Total Allocate $= 212 + 417 + 112 = 741$
unused $= 259$

Q7 (A) FIFO

REF	Frames	Page fault	Evicted
7	7, -, -	✓	
0	7, 0, -	✓	
1	7, 0, 1	✓	
2	2, 0, 1	✓	
0	2, 0, 1	×	7
3	2, 3, 1	✓	—
0	2, 3, 0	✓	0
4	4, 3, 0	✓	1
2	4, 2, 0	✓	2
3	4, 2, 3	✓	3
0	0, 2, 3	✓	0
3	0, 2, 3	×	4
2	0, 2, 3	×	2

Total FIFO page fault = 10

Date / /
Page No.

B optimal (Balady's optimal)

Ref	Frames (F_1, F_2, F_3)	Page fault	Evicted
7	7, -, -	✓	-
0	7, 0, -	✓	-
1	7, 0, 1	✓	-
2	2, 0, 1	✓	7
0	2, 0, 1	x	-
3	2, 0, 3	✓	1
0	2, 0, 3	x	-
4	2, 0, 4	✓	3
2	2, 0, 4	x	-
3	2, 0, 3	✓	4
0	2, 0, 3	x	-
3	2, 0, 3	x	-
2	2, 0, 3	x	-

Total optimal page fault = 7

C LRU

Ref	Frames (F_1, F_2, F_3)	Page fault	Evicted
7	7, -, -	✓	-
0	7, 0, -	✓	-
1	7, 0, 1	✓	-
2	2, 0, 1	✓	7
0	2, 0, 1	x	-
3	2, 0, 3	✓	1
0	2, 0, 3	x	-
4	4, 0, 3	✓	2
2	4, 2, 3	✓	0
3	4, 2, 3	x	-
0	0, 2, 3	✓	4

Date / /
Page No.

3	0, 2, 3	x	—
2	0, 2, 3	x	—

Total LRU page fault = 9

→ which perform Best
- optimal perform best user perfect future knowledge
- LRU is practical policy that approximely optimal.
- FIFO often perform worst and in susceptible to Belady's anomaly

Q8 Disk write = 10ms, 10,000 ms
memory write = 100ms
Extra time disks page = 10,000,000
− 100 = 9,999,900
30% of 1000 pages = 300
Total overhead = 300 × 9,999,900
= 3sec

optization = use write-back with dirty bit on page buffering