1. Explain Belady's anomaly with an exampl

Solution:

Consider this reference string:

1, 2, 3, 4, 1, 2, 5, 1, 2, 3, 4

& 3 free frames are available.

Apply FIFO page replacement.

1	1	1	4	4	4	5	5	5
	2	2	2	1	1	1	3	3
		3	3	3	2	2	2	4

The number of page faults for 3 frames = 9

Now Apply FIFO page replacement for above reference string with 4 free frame

1	1	1	1	5	5	5	5	4
	2	2	2	2	1	1	1	1
		3	3	3	3	2	2	2
			4	4	4	4	3	3

The number of page faults for 4 frames = 10

∴ The page fault rate may increase as the number of allocated frames increases. This unexpected result is known as Belady's anoma

(2) Consider this reference string:

1, 2, 3, 4, 1, 2, 5, 1, 2, 3, 4

& 3 free frames are available.

Apply FIFO page replacement.

1	1	1	4	4	4	5	5	5
	2	2	2	1	1	1	3	3
		3	3	3	2	2	2	4

The number of page faults for 3 frames = 9

Now Apply FIFO page replacement for above reference string with 4 free frame

1	1	1	1	5	5	5	5	4
	2	2	2	2	1	1	1	1
		3	3	3	3	2	2	2
			4	4	4	4	3	3

The number of page faults for 4 frames = 10

∴ The page fault rate may increase as the number of allocated frames increases. This unexpected result is known as Belady's anoma

2. A process has 16 bytes of logical memory, mapped in 4 bytes pages into 32 bytes of physical memory. (Presumably some other processes would be consuming the remaining 16 bytes of physical memory). How many bits are reserved for page number & page offset in the logical address. Suppose the logical address is 5. Calculate memory & page table.

Solution:-

page 0	0	a
	1	b
	2	c
	3	d
page 1	4	e
	5	f
	6	g
	7	h
page 2	8	i
	9	j
	10	k
	11	l
page 3	12	m
	13	n
	14	o
	15	p

Logical Memory

0	5
1	6
2	1
3	2

page table

0	
4	
8	
12	
16	
20	
24	e f g h
28	

Physical memory

Above diagram shows 32-byte memory with 4-byte pages.

Now consider the given logical address is 5.

physical address

- Now logical address should be mapped to physical address.

- Logical address divided into 2 parts: a page number & a page offset.

- page table contains the base address of each page in physical memory. This base address is combined with the page offset to define the physical memory address.

Consider logical address 5.

This address belongs to page 1 in the above ~~eb~~ example.

page number acts as index to page table
So page 1 has frame 6 in the page table.

$\therefore$ Physical address $= [(6 \times 4) + 1]$

$= 24 + 1 = \underline{\underline{25}}.$

[$\because$ 1 is offset here for logical address 5.]

(Note:-
For more details refer figure 8.9 in Memory Management Strategies Chapter in the text book Operating System Concepts by Peter. B. Galvin.)

* For page number $\underline{2 \text{ bits}}$ are reserved.
[(m-n) bits, higher order bits]

* For page offset $\underline{2}$ bits are reserved.
[(n bits), lower order bits]

Size of the logical address space is $2^m = 2^4$
$= 16$ bytes.

page size is $2^n = 2^2 = 4$ bytes.

$\therefore$ $m = 4$ & $n = 2$

page number $(m-n)$ bits $= (4-2) = 2$ bits.

page offset (n) bits $= 2$ bits.

3) Consider a paging system with the page table stored in memory. If a memory reference takes 200 ns how long does a paged memory reference take? If we add a TLB & 75 percent of all page-table references are TLB hits, what is the effective memory reference time?

Soln:-

A paged memory reference would take 400 ns.

i.e 200 ns to access the page table & 200 ns to access the word in memory.

$\therefore$ The Effective memory access time is $= 0.75 \times (200\text{ ns}) + 0.25 \times (400\text{ ns})$

$\boxed{\text{E.A.T.} = 250\text{ ns.}}$

4) Consider the following segment table:

Segment	Base	Length
0	219	600
1	2300	14
2	90	100
3	1327	580
4	1952	96.

What are the physical address for the following logical addresses? If the address generates a segment fault, indicate it as "Segment fault".

a) 0, 430

b) 1, 10

c) 2, 500

d) 3, 400

e) 4, 112.

Solution:

a) 219 + 430 = 649 (Add 430 to base of Segment 0)

b) 2300 + 10 = 2310 (Add 10 to ~~Seg~~ base of Segment 1).

c) Segment fault [Add 500 to base of segment 2 i.e $90 + 500 = 590$. Here $500 > 100$ (length of given segment 2 is 100)]

d) $1327 + 400 = 1727$

e) Segement fault (here $112 > 96$ i.e length of given Segment 4)

5) Assume that we have a demand page memory. The page table is held in registers. It takes 8 milliseconds to service a page fault if an empty frame is available or if the replaced page is not modified, & 20 milliseconds if the replaced page is modified. Memory access time is 100 ns.

Assume that the page to be replaced is modified 70 percent of the time. What is the maximum acceptable page-fault rate for an effective access time of no more than 200ns?

Solution:-

Note: 1 nanosec = 0.001 microseconds.

1 ms = 1000,000 ns.

~~Effective Access Time = (1-P) * Memory access time + P * Page fault service + P *~~

$$\boxed{EAT = (1-p) * \text{Memory Access time} + p * \text{Page fault Service} + p * \text{page fault Service after page replaced}}$$

$$0.2 \text{ micro sec} = (1-P) * 0.1 \text{ microsec} + (0.3P) * 8 \text{ millisec} + (0.7P) * 20 \text{ millisec}$$

$$(0.2 - 0.1) \text{ microsec} = -0.1 \cdot P + 2400p + 1400p$$

$$0.1 \sim= 16,400p$$

$$P \sim = 0.000006$$

5) Consider a demand-paging System with a paging disk that has an average access & transfer time of 20 millisecs. Addresses are translated through a page table in main memory, with an access time of 1 microsecond per memory access. Thus, each memory reference through the page table takes 2 accesses. To improve this time, we have added an associative memory that reduces access time to one memory reference, if the page-table entry is in the associative memory.

Assume that 80 percent of the accesses are in the associative memory & that, of the remaining, 10 percent (or 2 percent of the total) cause page faults. What is the effective memory access time?

Solution:

$$EAT = (0.8) * (1 \text{ microsec}) + (0.18) * 2 \text{ microsec} + (0.02) * (20002 \text{ microsec})$$

$$= 401.2 \text{ microsec}$$

$$= 0.4012 \text{ millisec}$$

7) Consider the following sequence of memory references from a 460 word program.

10, 11, 104, 170, 73, 309, 185, 245, 246, 434, 458, 364.

i) Show the reference string assuming page size of 100 words.

ii) Find page fault rate for the above reference string assuming 200 words of primary memory available & FIFO & LRU replacement algorithms.

Solution:

i) 0, 0, 1, 1, 0, 3, 1, 2, 2, 4, 4, 3

This reference string can be rewritten as

0, 1, 0, 3, 1, 2, 4, 3 → reference string.

ii)

Considering above reference string.

FIFO Replacement Algorithms

Its given that 200 words of primary memory available so 2 frames are available.

0	0	3	3	4	4
	1	1	2	2	3

page faults = 6.

LRU replacement algorithms

0 1 0 3 1 2 4 3

0	0	3	3	3
	1	1	2	4

page faults = 5

8. On a system using demand paging memory it takes 0.12 micro seconds to satisfy a memory request, if the page is in memory. If the page is not in memory the request takes 5000 μs. What would the page fault rate need to be to achieve an effective access time 1000 μs? Assume the s/m is only running a single process & the CPU is idle during the page swaps.

Solution :-

Effective access time $= (1-P) \times ma + P \times$ page fault time

$$EAT = (1-p) \times 0.12\,\mu s + p \times 5000\,\mu s$$

$$1000\mu = (1-p) \times 0.12\,\mu s + p \times 5000\,\mu s$$

$$1000\,\mu s = 0.12\,\mu s - (p \times 0.12\,\mu s) + p \times 5000\,\mu s$$

$$1000\,\mu s = 0.12\,\mu s - (4999.88 \times p)$$

$$999.88 = 4999.88 \times p$$

$$p = \underline{0.1999}$$