Problems on paging

For Main Memory-

- Physical Address Space = Size of main memory
- Size of main memory = Total number of frames x Page size
- Frame size = Page size
- If number of frames in main memory = 2^X, then number of bits in frame number = X bits
- If Page size = 2^X Bytes, then number of bits in page offset = X bits
- If size of main memory = 2^X Bytes, then number of bits in physical address = X bits

For Process-

- Virtual Address Space = Size of process
- Number of pages the process is divided = Process size / Page size
- If process size = 2^X bytes, then number of bits in virtual address space = X bits

For Page Table-

- Size of page table = Number of entries in page table x Page table entry size
- Number of entries in pages table = Number of pages the process is divided
- Page table entry size = Number of bits in frame number + Number of bits used for optional fields if any

NOTE-

- In general, if the given address consists of 'n' bits, then using 'n' bits, 2^n locations are possible.
- Then, size of memory = 2^n x Size of one location.
- If the memory is byte-addressable, then size of one location = 1 byte.
- Thus, size of memory = 2^n bytes.
- If the memory is word-addressable where 1 word = m bytes, then size of one location = m bytes.
- Thus, size of memory = 2^n x m bytes.

PRACTICE PROBLEMS BASED ON PAGING AND PAGE TABLE-

Problem-01:

Calculate the size of memory if its address consists of 22 bits and the memory is 2-byte addressable.

Solution-

We have-

- Number of locations possible with 22 bits = 2^{22} locations
- It is given that the size of one location = 2 bytes

Thus, Size of memory

= 2^{22} x 2 bytes

= 2^{23} bytes

= 8 MB

Problem-02:

Calculate the number of bits required in the address for memory having size of 16 GB. Assume the memory is 4-byte addressable

Solution-

Let 'n' number of bits are required. Then, Size of memory = 2^n x 4 bytes.

Since, the given memory has size of 16 GB, so we have-

2^n x 4 bytes = 16 GB

2^n x 4 = 16 G

2^n x 2^2 = 2^{34}

2^n = 2^{32}

$\therefore$ n = 32 bits

3. Consider a machine with 64 MB physical memory and a 32 bit virtual address space. If the page size is 4 KB, what is the approximate size of the page table?

1.16 MB

2. 8 MB

3. 2 MB
4.24 MB
Solution-

Given-

- Size of main memory = 64 MB
- Number of bits in virtual address space = 32 bits
- Page size = 4 KB

We will consider that the memory is byte addressable.

Number of Bits in Physical Address-

Size of main memory
= 64 MB
= 2^{26} B
Thus, Number of bits in physical address = 26 bits

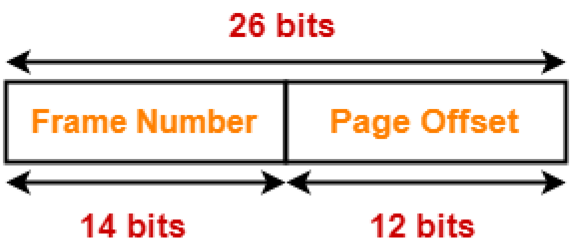
Number of Frames in Main Memory-

Number of frames in main memory
= Size of main memory / Frame size
= 64 MB / 4 KB
= 2^{26} B / 2^{12} B
= 2^{14}
Thus, Number of bits in frame number = 14 bits

Number of Bits in Page Offset-

We have,
Page size
= 4 KB
= 2^{12} B
Thus, Number of bits in page offset = 12 bits

So, Physical address is-



Physical Address

Process Size-

Number of bits in virtual address space = 32 bits

Thus,

Process size

= 2^{32} B

= 4 GB

Number of Entries in Page Table-

Number of pages the process is divided

= Process size / Page size

= 4 GB / 4 KB

= 2^{20} pages

Thus, Number of entries in page table = 2^{20} entries

Page Table Size-

Page table size

= Number of entries in page table x Page table entry size

= Number of entries in page table x Number of bits in frame number

= 2^{20} x 14 bits

= 2^{20} x 16 bits (Approximating 14 bits ≈ 16 bits)

= 2^{20} x 2 bytes

= 2 MB

PRACTICE PROBLEMS BASED ON Effective Access Time

- A page fault occurs when the referenced page is not found in the main memory.
- Page fault handling routine is executed on the occurrence of page fault.
- The time taken to service the page fault is called as **page fault service time**.

Effective Access Time (without page faults) =

Hit ratio of TLB x { Access time of TLB + Access time of main memory }

+

Miss ratio of TLB x { Access time of TLB + (L+1) x Access time of main memory }

where L = Number of levels of page table

Effective Access Time (with page faults) =

Page fault rate x { Effective Access time without page fault + Page fault service time }

+

(1 - Page fault rate) x { Effective Access time without page fault }

Problem-01:

Let the page fault service time be 10 ms in a computer with average memory access time being 20 ns. If one page fault is generated for every 10^6 memory accesses, what is the effective access time for the memory?

0. 21 ns
1. 30 ns
2. 23 ns
3. 35 ns

Solution-

Given-

- Page fault service time = 10 ms
- Average memory access time = 20 ns
- One page fault occurs for every 10^6 memory accesses

Page Fault Rate-

It is given that one page fault occurs for every 10^6 memory accesses.

Thus,

Page fault rate= $1 / 10^6$

$= 10^{-6}$

Effective Access Time With Page Fault-

It is given that effective memory access time without page fault = 20 ns.

Now, substituting values in the above formula, we get-Effective access time with page fault

$= 10^{-6} \times \{ 20 \text{ ns} + 10 \text{ ms} \} + (1 - 10^{-6}) \times \{ 20 \text{ ns} \}$

$= 10^{-6} \times 10 \text{ ms} + 20 \text{ ns}$

$= 10^{-5} \text{ ms} + 20 \text{ ns}$

$= 10 \text{ ns} + 20 \text{ ns}$

$= 30 \text{ ns}$

Thus, Option (B) is correct.

Problem-02:

Suppose the time to service a page fault is on the average 10 milliseconds, while a memory access takes 1 microsecond. Then, a 99.99% hit ratio results in average memory access time of-

0. 1.9999 milliseconds
1. 1 millisecond
2. 9.999 microseconds
3. 1.9999 microseconds
4. None of these

Hence, average memory access time = p*t1 + (1-p)*t2

=(99.99*1 + 0.01*(10*1000 + 1))/100

=1.9999 *10^-6 sec

Problem-03:

A system uses FIFO policy for page replacement. It has 4 page frames with no pages loaded to begin with. The system first accesses 100 distinct pages in some order and then access the same 100 pages but now in the reverse order. How many page faults will occur?

Solution

:Access to 100 pages will cause 100 page faults. When these pages are accessed in reverse order, the first four accesses will not cause page fault. All other access to pages will cause page faults. So total number of page faults will be 100 + 96.

Problem 4:

In which one of the following page replacement policies, Belady's anomaly may occur?

(A) FIFO
(B) Optimal
(C) LRU
(D) MRU

Answer (A)

Belady's anomaly proves that it is possible to have more page faults when increasing the number of page frames while using the First in First Out (FIFO) page replacement algorithm

2. A paging scheme uses a Translation Lookaside buffer (TLB). The effective memory access takes 160 ns and a main memory access takes 100 ns. What is the TLB access time (in ns) if the TLB hit ratio is 60% and there is no page fault?

1. 54
2. 60
3. 20
4. 75

Solution-

Given-

- Effective access time = 160 ns
- Main memory access time = 100 ns
- TLB Hit ratio = 60% = 0.6

Calculating TLB Miss Ratio-

TLB Miss ratio

= 1 – TLB Hit ratio

= 1 – 0.6

= 0.4

Calculating TLB Access Time-

Let TLB access time = T ns.

Substituting values in the above formula, we get-

160 ns = 0.6 x { T + 100 ns } + 0.4 x { T + 2 x 100 ns }

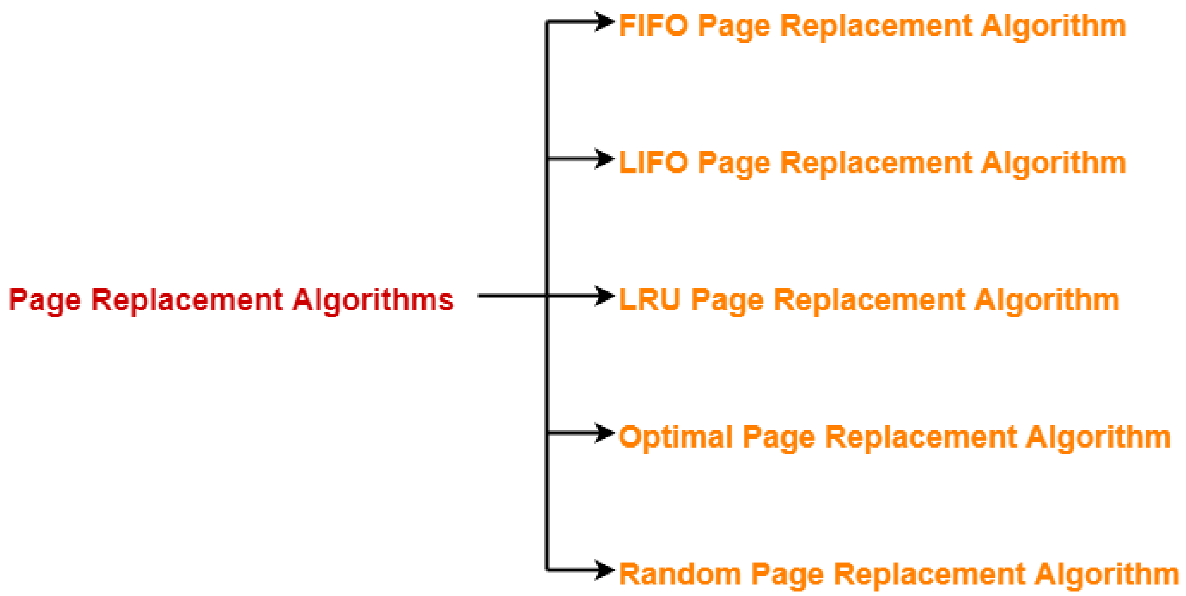
160 = 0.6 x T + 60 + 0.4 x T + 80

160 = T + 140

T = 160 – 140

T = 20

Page Replacement Algorithms-



Problem-01:

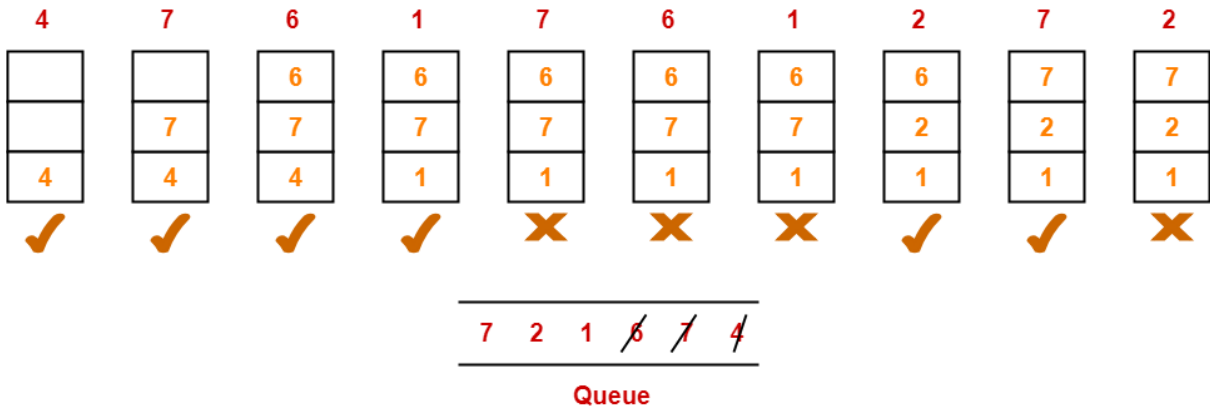
A system uses 3 page frames for storing process pages in main memory. It uses the First in First out (FIFO) page replacement policy. Assume that all the page frames are initially empty. What is the total number of page faults that will occur while processing the page reference string given below-

4 , 7, 6, 1, 7, 6, 1, 2, 7, 2

Also calculate the hit ratio and miss ratio.

Solution-

Total number of references = 10



Total number of page faults occurred =6

Calculating Hit ratio-

Total number of page hits

= Total number of references – Total number of page misses or page faults

= 10 – 6

= 4

Thus, Hit ratio

= Total number of page hits / Total number of references

= 4 / 10

= 0.4 or 40%

Calculating Miss ratio-

Total number of page misses or page faults = 6

Thus, Miss ratio= Total number of page misses / Total number of references

= 6 / 10

= 0.6 or 60%

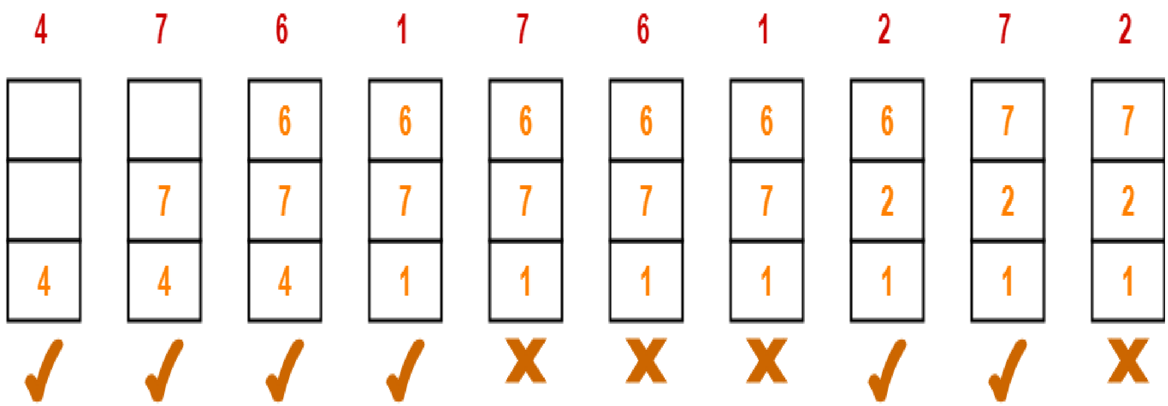
2. A system uses 3 page frames for storing process pages in main memory. It uses the Least Recently Used (LRU) page replacement policy. Assume that all the page frames are initially empty. What is the total number of page faults that will occur while processing the page reference string given below-

4 , 7, 6, 1, 7, 6, 1, 2, 7, 2

Also calculate the hit ratio and miss ratio.

Solution-

Total number of references = 10



From here,

Total number of page faults occurred = 6

in the similar manner as above-

- Hit ratio = 0.4 or 40%
- Miss ratio = 0.6 or 60%

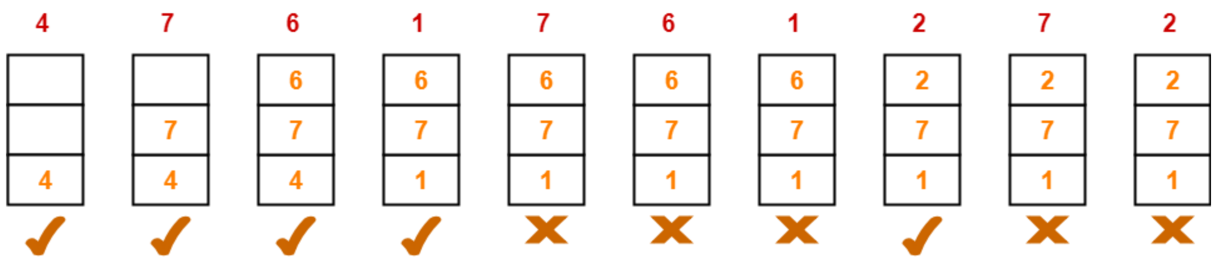
3. A system uses 3 page frames for storing process pages in main memory. It uses the Optimal page replacement policy. Assume that all the page frames are initially empty. What is the total number of page faults that will occur while processing the page reference string given below-

4 , 7, 6, 1, 7, 6, 1, 2, 7, 2

Also calculate the hit ratio and miss ratio.

Solution-

Total number of references = 10



Total number of page faults occurred = 5

- Hit ratio = 0.5 or 50%
- Miss ratio = 0.5 or 50%