PRACTICE PROBLEMS BASED ON TLB

Formula for **L-level paging scheme with TLB** is:

EAT = A (E + M) +(1 - A) (E+(L+1) x M)

where, L = no of levels, A = TLB hit ratio, M = memory access time, E = TLB access time, 1 - A = TLB miss ratio

Problem-01

Consider a two-level paging scheme with a TLB. Assume no page fault occurs. It takes 20 ns to search the TLB and 100 ns to access the physical memory. If TLB hit ratio is 80%, the effective memory access time is _______ msec.

Solution

Given

- Number of levels of page table L = 2
- TLB access time = 20 ns
- Main memory access time = 100 ns
- TLB Hit ratio = 80% = 0.8

Calculating TLB Miss Ratio

TLB Miss ratio
= 1 – TLB Hit ratio
= 1 – 0.8 = 0.2

Calculating Effective Access Time

Substituting values in the above formula, we get
Effective Access Time
= 0.8 x (20 ns + 100 ns) + 0.2 x (20 ns + (2+1) x 100 ns)
= 0.8 x 120 ns + 0.2 + 320 ns = 96 ns + 64 ns = 160 ns
Thus, effective memory access time = 160 ns.

Problem-02

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?

A. 54
B. 60
C. 20
D. 75

Solution

Given EAT = A(E+M) + (1-A)(E+2M)   (L=1 level)

- 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

Option (C) is correct.

Week 9

Tutorial
Virtual Memory

Exercise 1.

The Effective Access Time (EAT)

EAT = (1 – p) x Memory Access + p x page fault time

In a **demand-paging system**, it takes **250 ns** to satisfy a memory access when the requested page is in the resident set.

If it is **not in the resident set**, then the request takes **10 ms** if a free frame is found or the page to be replaced is not modified**.** Such requests are 3% of all the accesses**.**

Otherwise, if there is no free frame and the page to be replaced is modified, then it takes **20 ms.** Such pages are 7% of all the accesses**.**

If the Page Fault Rate PFR in the system is 10%, then what will be the **EAT**?

Solution

The Effective Access Time (EAT)

EAT = (1 – p) x Memory Access + p x page fault time

EAT = 0.9 × 0.00025 + 0.03 x 10 + 0.07 × 20
= 0.000225 + 0.3 + 1.4
= 1.700225 ms

+

10% PFR

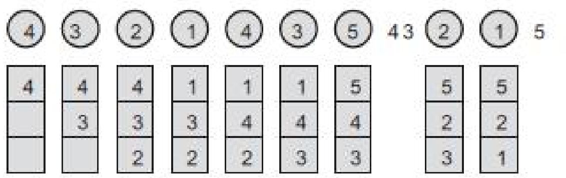
Exercise 2.

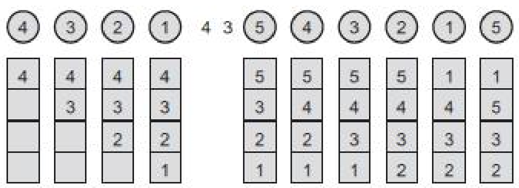
Calculate the number of page faults for the following reference string using **FIFO algorithm** with frame size as **3** and then as **4.**

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

FIFO - the oldest page is chosen

Solution





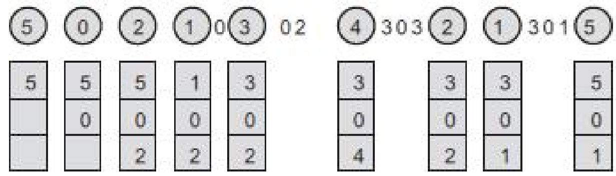
Exercise 3.

Calculate the number of page faults for the following reference string using **optimum algorithm** with frame size as **3.**

5 0 2 1 0 3 0 2 4 3 0 3 2 1 3 0 1 5

OA – replace a page that will not be used for longest period of time

Solution



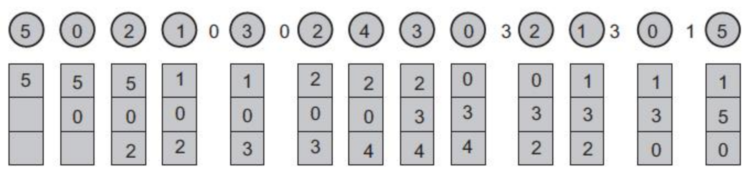
Exercise 4.

Calculate the number of page faults for the following reference string using **LRU page-replacement algorithm** with frame size as **3.**

5 0 2 1 0 3 0 2 4 3 0 3 2 1 3 0 1 5

throw out the page that has been unused for the longest time.

Solution



PRACTICE PROBLEM BASED ON PAGE FAULT

- 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

In a multilevel paging scheme using TLB without any possibility of page fault, effective access time is given by:

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

In a multilevel paging scheme using TLB with a possibility of page fault, effective access time is given by:

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

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?

a. 21 ns
b. 30 ns
c. 23 ns
d. 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.

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

$= 10^{-6} \times (20\text{ ns} + 10\text{ ms}) + (1 - 10^{-6}) \times 20\text{ ns} = 10^{-6} \times 10\text{ ms} + 20\text{ ns} = 10^{-5}\text{ ms} + 20\text{ ns}$

$= 10\text{ ns} + 20\text{ ns} = 30\text{ ns}$

Option (B) is correct.

PRACTICE PROBLEMS BASED ON PAGE REPLACEMENT ALGORITHMS

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:

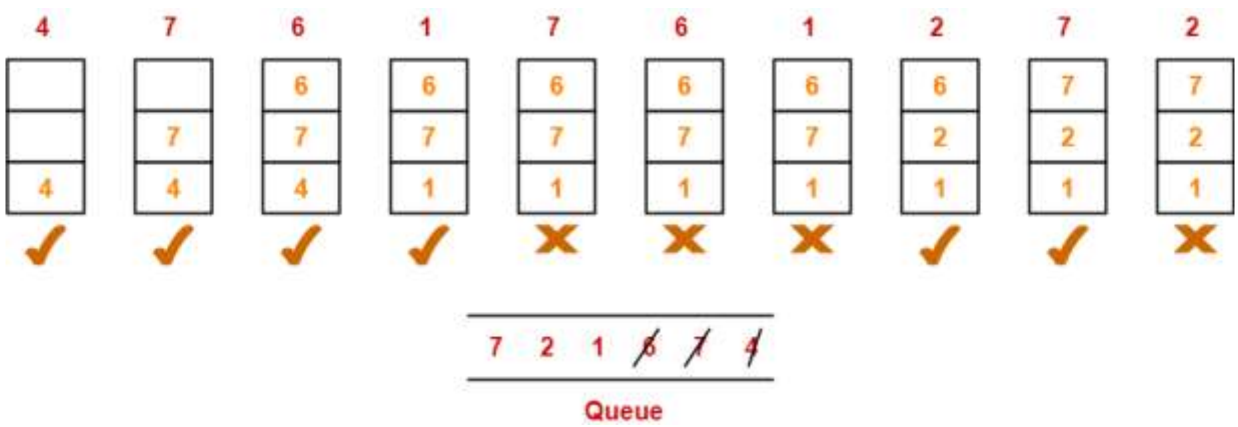
A system uses 3-page frames for storing process pages in main memory. It uses the **First in First out (FIFO) page replacement** policy. Assume that all the page frames are initially empty. What is the total number of page faults that will occur while processing the page reference string given below?

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

Also calculate the hit ratio and miss ratio.

Solution

Total number of references = 10



From here,

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%

Alternatively,
Miss ratio
= 1 – Hit ratio
= 1 – 0.4
= 0.6 or 60%

Problem-02:

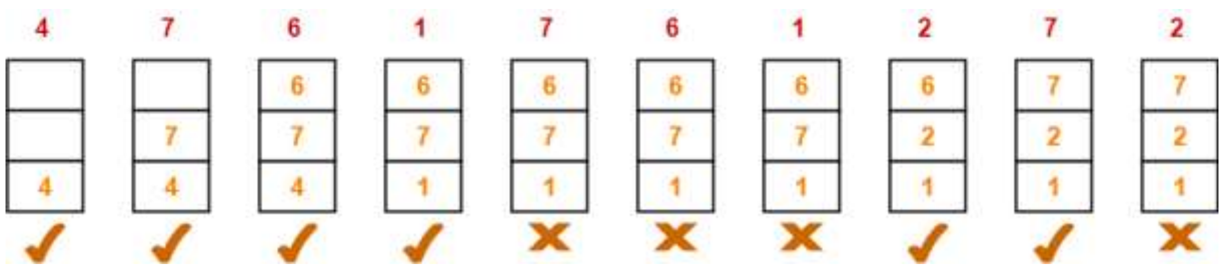
A system uses 3-page frames for storing process pages in main memory. It uses the **Least Recently Used (LRU) page replacement** policy. Assume that all the page frames are initially empty. What is the total number of page faults that will occur while processing the page reference string given below?

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%

Problem-03:

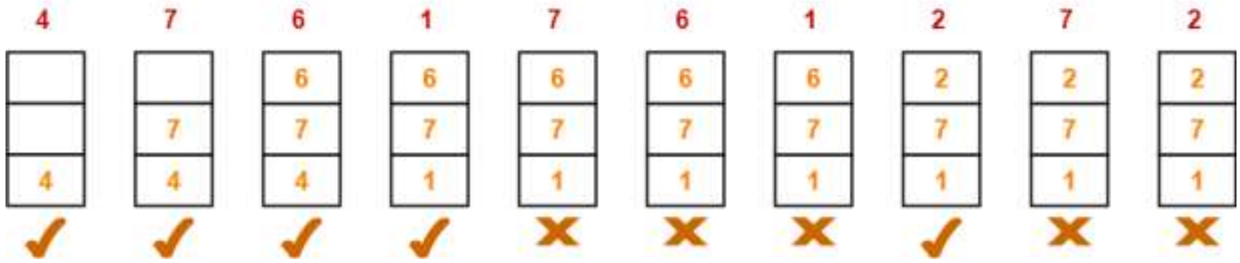
A system uses 3-page frames for storing process pages in main memory. It uses the **Optimal page replacement** policy. Assume that all the page frames are initially empty. What is the total number of page faults that will occur while processing the page reference string given below?

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

Also calculate the hit ratio and miss ratio.

Solution

Total number of references = 10



From here,

Total number of page faults occurred = 5

In the similar manner as above

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

Solutions

1. FCFS:

```
-------------------------------------------------------------
|          P1          | P2 |    P3    | P4 |       P5       |
-------------------------------------------------------------
0                      10   11         13   14               19
```

Performance statistics:

Process	Arrival Time	Burst Time	Priority	Finish Time	Turnaround Time	Waiting Time
P_1	0	10	3	10	10	0
P_2	0	1	1	11	11	10
P_3	0	2	3	13	13	11
P_4	0	1	4	14	14	13
P_5	0	5	2	19	19	14
Average					13.4	9.6

2. SJF (nonpreemptive):

```
-------------------------------------------------------------
| P2 | P4 |   P3   |     P5     |           P1              |
-------------------------------------------------------------
0    1    2        4            9                           19
```

Performance statistics:

Process	Arrival Time	Burst Time	Priority	Finish Time	Turnaround Time	Waiting Time
P_1	0	10	3	19	19	9
P_2	0	1	1	1	1	0
P_3	0	2	3	4	4	2
P_4	0	1	4	2	2	1
P_5	0	5	2	9	9	4
Average					7	3.2

3. Round Robin:

```
-------------------------------------------------------------------------------------------
| P1 | P2 | P3 | P4 | P5 | P1 | P3 | P5 | P1 | P5 | P1 | P5 | P1 | P5 | P1 | P1 | P1 | P1 | P1 |
-------------------------------------------------------------------------------------------
0    1    2    3    4    5    6    7    8    9    10   11   12   13   14   15   16   17   18   19
```

Performance statistics:

Process	Arrival Time	Burst Time	Priority	Finish Time	Turnaround Time	Waiting Time
P_1	0	10	3	19	19	9
P_2	0	1	1	2	2	1
P_3	0	2	3	7	7	5
P_4	0	1	4	4	4	3
P_5	0	5	2	14	14	9
Average					9.2	5.4

Week 11

Tutorial

Ex.1

A system has four processes and five allocable resources. The current allocation and maximum needs are as follows:

	Allocated	*Maximum*	*Available*
Process A	1 0 2 1 2	1 1 2 1 3	0 0 x 1 1
Process B	2 0 1 1 0	2 2 2 1 0	
Process C	1 1 0 1 0	2 1 3 1 0	
Process D	1 1 1 1 0	1 1 2 2 1	

What is the smallest value of x for which this is a safe state?

Hint:

Try putting values for x = 0, 1, 2…, until you find the solution.

Ex.2

Consider the following set of processes, with the length of the CPU burst given in milliseconds:

Process	Burst Time
P_1	10
P_2	1
P_3	2
P_4	1
P_5	5

The processes are assumed to have arrived in the order P1, P2, P3, P4, P5 **all** at time 0.

a. Draw the Gantt charts that illustrate the execution of these processes using the following scheduling algorithms: FCFS, SJF, and RR (quantum = 1).
b. What is the turnaround time of each process for each of the scheduling algorithms in part a?
c. What is the waiting time of each process for each of these scheduling algorithm?

Solutions

a). 26

b). 2^{10}: same as the page size

c). 21: 31 (entire address) - 10 (offset)

d). 2^{16}: the number of pages

e). 22: 21 (page frame) + 1 (valid)

Ex.3

Given the following stream of page references by an application, calculate the number of page faults the application would incur with the following page replacement algorithms. Assume that all pages are initially free.

Reference Stream: **A B C D A B E A B C D E B A B**

a. FIFO page replacement with 3 physical pages available.

b. LRU page replacement with 3 physical pages available.

Solutions

FIFO

11 page faults

A	B	C	D	A	B	E	A	B	C	D	E	B	A	B
A	A	A	B	C	D	A	A	A	B	E	E	C	D	D
	B	B	C	D	A	B	B	B	E	C	C	D	B	B
		C	D	A	B	E	E	E	C	D	D	B	A	A
√	√	√	√	√	√	√			√	√		√	√	

LRU

12 page faults

A	B	C	D	A	B	E	A	B	C	D	E	B	A	B
A	A	A	B	C	D	A	B	E	A	B	C	D	E	E
	B	B	C	D	A	B	E	A	B	C	D	E	B	A
		C	D	A	B	E	A	B	C	D	E	B	A	B
√	√	√	√	√	√	√			√	√	√	√	√	

Ex.4

Consider a simple paging system with the following parameters:

- 2^{31} bytes of addressable physical memory;
- page size of 2^{10} bytes;
- 2^{26} bytes of logical address space.

(a) How many bits are in a logical address?

(b) How many bytes in a frame?

(c) How many bits in the physical address specify the frame?

(d) How many entries in the page table?

(e) How many bits in each page table entry (assume each page table entry includes a valid/invalid bit).