TANU SHREE
CSCE 611 - OS

WEEK-4 HOMEWORK

Problem:1

Given, page table lookup time = 500ns
TLB lookup time = 100ns
Avg. address translation time = 200ns

To find: TLB hit-rate.

→ Let TLB hit rate be x.

In case of TLB hit, only TLB is accessed, but in case of TLB miss, both TLB and page table accessed (assuming one page table).

∴ Avg. address Translation time = Time taken in hit + Time taken in miss..

$$\Rightarrow 200 = x(100) + (1-x)(100+500)$$
$$\Rightarrow 200 = 100x + 600 - 600x$$
$$\Rightarrow 500x = 400$$
$$x = \frac{400}{500} = 0.8$$

∴ The hit rate is 0.8 or 80%.

Problem-2

Segmentation and paging are sometimes combined into one scheme to improve eachother. When the size of page table becomes too large (in case of large logical address space), the page table consumes a lot of memory. To overcome this, segmentation of page table is done. This is called segmented paging.

The address provided by CPU in this case would be partitioned as:

segment no. page no. offset.

Paged segmentation on the other hand, are used when the segments in segmented paging becomes too large sometime and it takes a lot of time for allocation. By paging the segments, memory wasted due to external fragmentation is reduced and allocation is simplified too.

Problem-3

Segmented paging scheme-

When there is large address space, the page table becomes very large resulting in consuming large memory space. To avoid that segmentation of page table is done, where the parts of page table are segmented in the segment table. Each entry in the segment table point to pages that belong to that segment.

Hashed page tables-

In this, the virtual address (spaces higher than 32 bits) is hashed into hashed tables.
Each element in the hash table has virtual page no., the value of mapped page and pointer to next element.
The virtual page no. is matched against the virtual address (first field) and if matched, corresponding mapped address is used to form

the desired memory address.

In hash table, problem of collision can occur. It happens when multiple pages belonging to different processes hash to same entry in the page table. Collisions can be detected by comparing the virtual address with page table entries but traversing the list for that results in overhead. To overcome such overhead, segmented paging is preferred because each page table entry only contains the information of one page.

When program occupies small portion of its large virtual address space, in that case hash tables are preferred over segmented paging.

Problem-4 :

Page fault : It occurs when the page is not present in the physical memory. It means it is present in secondary memory but not yet loaded into frame of physical memory.

a.) TLB miss with no page fault.
This scenario is possible. When the page is brought in memory but it has been removed from the TLB.

b.) TLB miss and page fault
Possible. The page is not found in TLB as well as in the memory. It means page fault occured.

c.) TLB hit and no page fault

Possible. Page is found both in the TLB and in the physical memory.

d.) TLB hit and page fault

Not possible: TLB hit means page is in the TLB, and since TLB is the cache of page table so, page must be in the memory, page fault cannot occur.