

Atkhar Hakim knowsare
1705045

① Advantage:- Rather than consuming memory for virtual page number of entries in page table for "each process", inverted page table system will have number of entries equals to number of page ~~frames~~ ~~number~~ for all process. It is hugely memory efficient.

Disadvantage:- We need to traverse whole page table in each memory reference.

Way:- We can use hash-table, where we will have on virtual page number; and each entries in hashtable will be a chain of vpn, ppn, pid .

②

④ When a TLB miss occurs, page table will be ~~enquired~~ searched to get page frame number.

Now, if TLB has empty slots, that slot will be used to hold information (vpn, ppn, R, M) about the page we just searched.

If TLB has no empty slot, one entry will be evicted from TLB. We will use reference

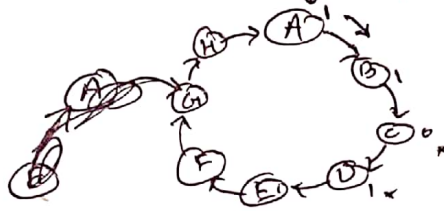
bit to determine which one to evict. Then, that slot in TLB will be used to hold information about the page we just searched.

⑤

Advantage:- We don't need to search k -bits in bitmaps when we need k -allocation units. Because we are keeping linked list here, we will traverse the linked list.

Disadvantage:- If we keep separate lists for process and holes, it will be little complex to compute the whole thing. Further, merging entries of linked list will be complex. ~~See~~ Wrong selection in searching may create unnecessary holes.

②



If pointer or hand starts from A, it will first get C with $R=0$. So, C will be evicted for first page fault.

For next page fault, hand starts searching from D. ~~So~~ D has $R=0$. So, D will be evicted for second page fault.