11/8/23

Alternative Spring Break

Mechanism of Page Eviction

Today: which page to evict?

☆ -> flush TLB, clear PTE present bit for the evicted page
-> allocate space in swap, write evicted page to swap
-> track page => swap loc
-> zero out old content before mapping it to a new page

multiple pages => frame
tracked by frame metadata
xk: core map entry

Question in class: why write page to swap instead of using disk directly (map page to swap)

-> architecture:
code and data needs to be in memory to be accessed by the CPU
disk is block addressable, much much slower in access latency

There's a non traditional persistent device: non volatile memory (byte addressable, slower access than DRAM, but can be accessed as memory directly), offers an interesting design space

Eviction Policies: What page/frame to evict?

-> FIFO
  -> pick the page that's brought in first (longest time in memory)
  -> a queue of frames in order of allocation
  -> doesn't care about access patterns

Page accessed in this order : A, B, C, D, A, B, E, A, B, C, D, E

FIFO 3 frames

Frame 1	A	A	A	D	D	D	E	E	E	E	E	E
Frame 2	-	B	B	B	A	A	A	A	A	C	C	C
Frame 3	-	-	C	C	C	B	B	B	B	B	D	D
	x	x	x	x	x	x	x			x	x	

(9 PFs)

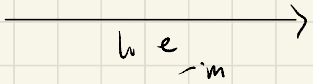
Belady's Anomaly

More frames may cause more page faults with FIFO policy

FIFO 4 frames

Frame 1	A	A	A	A	A	A	E	E	E	E	D	D
Frame 2	-	B	B	B	B	B	B	A	A	A	A	E
Frame 3	-	-	C	C	C	C	C	C	B	B	B	B
Frame 4	-	-	-	D	D	D	D	D	D	C	C	C
	x	x	x	x			x	x	x	x	x	x

(10 PFs)



-> Least Recently Used
  -> evict page that's least recently used
  -> good when access exhibits locality
  -> worst case: N frames N+1 pages accessed in order
  -> How to implement this?
    -> SW: queue, when a page is used, bring it to the back, head of queue is the LRU page
    -> HW: implement HW queue, or add timestamp update to PTE

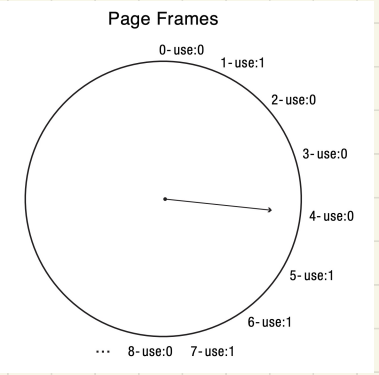
needs to be done on every access
(TLB miss& TLB hits)

-> Clock Algorithm
  -> approximates LRU
  -> uses access bit (PTE) to estimate if the page is used recently
  -> algorithm starts at clock hand (stateful, advances each time the algo runs)

-> set upon TLB miss (have to access PTE anyway)

if (access bit == 0) { found our page to evict, advance clock hand }
else { set access bit to 0 & advance clock hand }

needs to also do a TLB shootdown to ensure future access to the page updates the access bit back to 1



☆ takes access pattern into account, but doesn't care about whether the page has been written to.