운영 체제 4차 (2015/5/27)

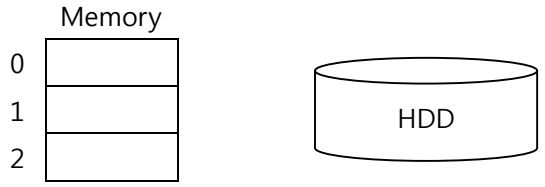
학번: _____ 이름: _____

I. Demand Paging

- Physical memory에는 3개의 page frame이 있다.
- A,B,C, … 는 virtual page number이다.
- Virtual address 의 offset은 2자리이다.

  예: <B/22> - virtual address

  <1/22> - physical address



다음 각 메모리 접근 때 발생하는 사건들 때문에 TLB와 page table에 변화가 있는 경우에만 **그 내용**(0,1,A,B,C 등)을 해당 셀의 ( )안에 기입하기로 한다 이 때 Page replacement가 필요하면 **1번 frame**을 선택해서 내보내기로 한다.

TLB에서 교체가 일어나는 경우 invalid한 entry를 우선 내보낸다.

Software loaded TLB를 가정한다. 즉 TLB miss에 의해 OS trap이 발생하며 추가적 page 교체가 필요한 경우 page replacement service를 커널 내부에서 일반 함수로 호출한다.

a) Read from <B/22>

TLB

Virtual Page #	Page frame #	valid	use	modified
B ( )	2 ( )	1 ( )	1 (가)	0 (나)
C ( )	- ( )	0 ( )	- ( )	- ( )

Page Table

	Page frame	valid	use	modified
A	1 (다)	1 (라)	1 ( )	1 ( )
B	2 ( )	1 ( )	1 ( )	0 ( )
C	0 ( )	1 ( )	1 ( )	0 ( )
D	0 ( )	0 ( )	- ( )	- ( )

1. 발생하는 사건은?
   - I. OS trap by TLB miss
   - II. Call page replacement service in kernel

   1) 없음 2) I 3) II 4) I, II
2. 가 ~ 라 의 내용은?

   1) 모름 2) 0 3) 1 4) 2

b) Write to <C/22> // a)와 상관없는 독립 사건

TLB

Virtual Page #	Page frame #	valid	use	modified
B ( )	2 ( )	1 ( )	1 ( )	0 ( )
C ( )	- (가)	0 (나)	- (다)	- (라)

Page Table

	Page frame	valid	use	modified
A	1 ( )	1 ( )	1 ( )	1 (마)
B	2 ( )	1 ( )	1 ( )	0 ( )
C	0 ( )	1 ( )	1 ( )	0 (바)
D	0 ( )	0 ( )	- ( )	- ( )

3. 발생하는 사건은?
   - I. OS trap by TLB miss
   - II. Call page replacement service in kernel

   1) 없음 2) I 3) II 4) I, II
4. 가 ~ 바 의 내용은?

   1) 모름 2) 0 3) 1 4) 2

c) Read from <D/22> // b)와 상관없는 독립 사건

TLB

Virtual Page #	Page frame #	valid	use	modified
B ( )	2 ( )	1 (가)	1 (나)	0 ( )
C (다)	- (라)	0 ( )	- ( )	- ( )

Page Table

	Page frame	valid	use	modified
A	1 (마)	1 (바)	1 ( )	1 (사)
B	2 (아)	1 ( )	1 ( )	0 (자)
C	0 ( )	1 ( )	1 ( )	0 ( )
D	0 (차)	0 (카)	- (타)	- ( )

5. 발생하는 사건은?
   - I. OS trap by TLB miss
   - II. Call page replacement service in kernel

   1) 없음 2) I 3) II 4) I, II
6. 가 ~ 타 의 내용은?

   1) 모름 2) 0 3) 1 4) 2

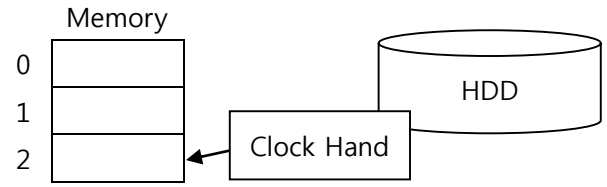
운영 체제 4차 (2015/5/27)

학번: _____ 이름: _____ 2/3

II. Emulate Use and Modified bits

Page replacement가 필요하면 **clock algorithm**으로 교체할 frame을 선택한다.

Clock hand는 현재 2번 frame을 가리키고 있고, 2,0,1,2,0,1,2, ...로 움직인다.



Emulate modified bit

1. Initially, mark all pages as read-only, even data pages
2. On write, trap to OS. OS sets software "modified" bit, and marks page as read-write.
3. Whenever page comes back in from disk, mark read-only

Emulate use bit

1. Mark all pages as invalid, even if in memory
2. **On read** to invalid page, trap to OS
3. OS sets use bit, and marks page read-only

*2'. **On write**, trap to OS (either invalid or read-only)*

3'. Set use and modified bits, mark page read-write

4. When clock hand passes by, reset use and modified bits and mark page as invalid again

a) Read from <A/22>

Page table for MMU

	Page frame	valid	Read only
A	1 ( )	0 (가)	0 (나)
B	0 ( )	1 ( )	1 ( )
C	2 (다)	1 (라)	1 ( )
D	- ( )	0 ( )	0 ( )

Page Table

	Page frame	valid	Read only	modified	use
A	1 ( )	1 ( )	1 (마)	0 (바)	0 (사)
B	0 ( )	1 ( )	0 ( )	0 ( )	0 ( )
C	2 ( )	1 ( )	1 (아)	1 (자)	1 (차)
D	- ( )	0 ( )	0 ( )	0 ( )	0 ( )

7. 발생하는 사건은?
   - I. OS trap by Page fault
   - II. Access violation(write on read only page)

   1) 없음 2) I 3) II 4) I, II
8. 가 ~ 차 의 내용은?

   1) 모름 2) 0 3) 1 4) 2

b) Write to <B/22> // a)와 상관없는 독립 사건

Page table for MMU

	Page frame	valid	Read only
A	1 ( )	0 ( )	0 ( )
B	0 ( )	1 (가)	1 (나)
C	2 ( )	1 ( )	1 (다)
D	- ( )	0 ( )	0 (라)

Page Table

	Page frame	valid	Read only	modified	use
A	1 ( )	1 ( )	1 ( )	0 (마)	0 ( )
B	0 ( )	1 ( )	0 (바)	0 (사)	1 (아)
C	2 ( )	1 ( )	1 ( )	1 (자)	1 ( )
D	- ( )	0 ( )	0 ( )	0 (차)	0 ( )

9. 발생하는 사건은?
   - I. OS trap by Page fault
   - II. Access violation(write on read only page)

   1) 없음 2) I 3) II 4) I, II
10. 가 ~ 차 의 내용은?

    1) 모름 2) 0 3) 1 4) 2

운영 체제 4차 (2015/5/27)

학번: _____ 이름: _____ 3/3

c) Read from <D/22> // b)와 상관없는 독립 사건

Page table for MMU

	Page frame	valid	Read only
A	1 ( )	0 ( )	0 ( )
B	0 ( )	1 (가)	1 ( )
C	2 ( )	1 (나)	0 ( )
D	- (다)	0 (라)	0 (마)

Page Table

	Page frame	valid	Read only	modified	use
A	1 ( )	1 ( )	1 ( )	0 ( )	0 ( )
B	0 (바)	1 (사)	0 (아)	0 (자)	0 (차)
C	2 (카)	1 (타)	0 (파)	1 (하)	1 (거)
D	- (너)	0 (더)	0 (러)	0 (머)	0 (버)

11. 발생하는 사건은?

I. OS trap by Page fault

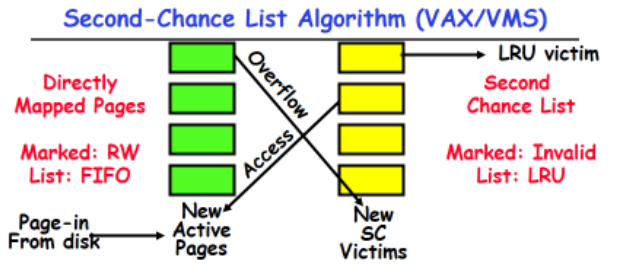
II. Access violation(write on read only page)

1) 없음 2) I 3) II 4) I, II

12. 가 ~ 버 의 내용은?

1) 모름 2) 0 3) 1 4) 2

III. Second chance: mapped/unmapped list

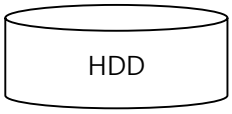


- Split memory in two: Active list (RW), SC list (Invalid)
- Access pages in Active list at full speed
- Otherwise, Page Fault
  - Always move overflow page from end of Active list to front of Second-chance list (SC) and mark invalid
  - Desired Page On SC List: move to front of Active list, mark RW
  - Not on SC list: page in to front of Active list, mark RW; page out LRU victim at end of SC list

55

Memory

0	
1	
2	
3	



a) Read from <D/22>

Page Table for MMU

	Page frame	valid
A	1 ( )	1 ( )
B	0 (가)	0 (다)
C	2 ( )	1 (라)
D	- (나)	0 (마)
E	3 ( )	0 (바)
F	- ( )	0 ( )

Page Table for OS

	Page frame	valid
A	1 ( )	1 ( )
B	0 ( )	1 (자)
C	2 ( )	1 ( )
D	- (사)	0 (차)
E	3 ( )	1 (카)
F	- (아)	0 (타)

Mapped page list(FIFO)	2 (파)	0 (거)	Second chance list(LRU)
	1 (하)	3 (너)	

13. 발생하는 사건은?

I. OS trap by Page fault

II. Access violation(write on read only page)

1) 없음 2) I 3) II 4) I, II

14. 가 ~ 너 의 내용은?

1) 모름 2) 0 3) 1 4) 2 5) 3