

OS_페이지 교체 알고리즘

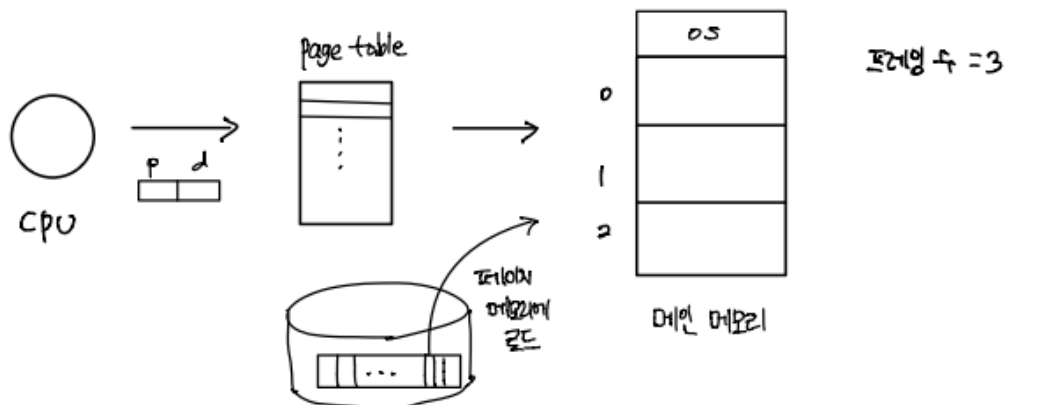
페이지 교체 알고리즘

- FIFO
- OPT
- LRU
-

FIFO (First In First Out)

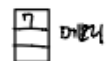
메모리에 먼저 로드된 페이지를 먼저 내보낸다(victim으로 선택한다)

- Simplest
 - idea: 초기화 코드는 더 이상 사용되지 않을 것
- 예제
 - 페이지 참조열 7 0 1 2 0 3 0 4 2 3 0 3 2 1 2 0 1 7 0 1
 - number of frames = 3 (메모리 프레임 3개)
 - 15 page faults



- 처음 power on \Rightarrow 빈 메모리 상태 .

① 페이지 참조열 = 7 \Rightarrow 빈 메모리 상태라 page fault 발생 \Rightarrow 하드디스크에서 page 7 찾아서 (page fault count = 1) 메모리에 로드



② 페이지 참조열 = 0 \Rightarrow 메모리에 있음, page fault 발생 (page fault count = 2)



③ 페이지 참조열 = 1 \Rightarrow 메모리에 있음, PF 발생 (PF count = 3)



④ 페이지 참조열 2 \Rightarrow 메모리에 없음, PF 발생 (PF count = 4), 메모리에서 7번에 2 넣음(FIFO)



:

page	7	0	1	2	0	3	0	4	2	3	0	3	2	1	2	0	1	7	0	1
Frame (메모리)	7	0	1	2	2	3	0	4	2	3	0	0	0	1	2	2	2	7	0	1
				1	1	2	3	0	4	2	3	3	3	0	1	1	1	2	7	0
				0	0	1	2	3	0	4	2	2	2	3	0	0	0	1	2	7
PF 발생	0	0	0	0	X	0	0	0	0	0	0	X	X	0	0	X	X	0	0	0

↓ 메모리 0 존재
 ↓ 메모리 3 존재 ↓ 메모리 2 존재 ↓ 메모리 0 존재 ↓ 메모리 1 존재

∴ PF 발생: 19회

↳ 위 표는 세로로 보시면 됩니다! ex) 페이지 2를 로드할 때 메모리 (2.1.0)은 이전 메모리 (1.0.7)이 2가 있어서 가장 처음 로드된 7이 버지고 2가 새로 메모리에 로드된 것으로 뜻합니다. 페이지 교체 발생했으므로 PF 발생 0

현재 프레임은 위 표와 같이 로드된 것은 아래쪽으로 밀리지 않습니다!!

가장 먼저 로드된 페이지 번호를 찾기 편하도록 표현하기 위해 큐처럼 나타낸 것입니다.

- Belady's Anomaly
 - 프레임 수 (= 메모리 용량) 증가에 PF(page fault) 회수 증가?

OPT (Optimal)

- 규칙: 가장 오랫동안 쓰이지 않을 페이지를 victim으로 선택 => page fault가 가장 적다
- 예제
 - 페이지 참조열: 7 0 1 2 0 3 0 4 2 3 0 3 2 1 2 0 1 7 0 1
 - 프레임 수: 3
 - page fault: 9

page	7	0	1	2	0	3	0	4	2	3	0	3	2	1	2	0	1	7	0	1
Frame	7	7	7	2	2	2	2	2	2	2	2	2	2	2	2	2	2	7	7	7
		0	0	0	0	0	0	4	4	4	0	0	0	0	0	0	0	0	0	0
				1	1	1	3	3	3	3	3	3	3	1	1	1	1	1	1	1
PF	0	0	0	0	X	0	X	0	X	X	0	X	X	0	X	X	X	0	X	X

↓ 뒤 page를 보면 7, 0, 1 중 7이 가장 늦게 사용됨
 ↓ 뒤 page를 보면 2, 1, 0 중 1이 가장 늦게 사용됨
 ↓ 뒤 page를 보면 2, 0, 3 중 0이 가장 늦게 사용됨
 ↓ 뒤 page를 보면 4, 2, 3 중 2가 가장 늦게 사용됨
 ↓ 3은 이제 사용 X
 ↓ 2는 이제 사용 X

PF 횟수: 9

- Unrealistic => 뒤에 오는 page(미래) 알수 없음
 - cf) SJF CPU 스케줄링 알고리즘 (shortest job first) => 어떤 작업이 얼마나 걸릴지 모름, 비현실적

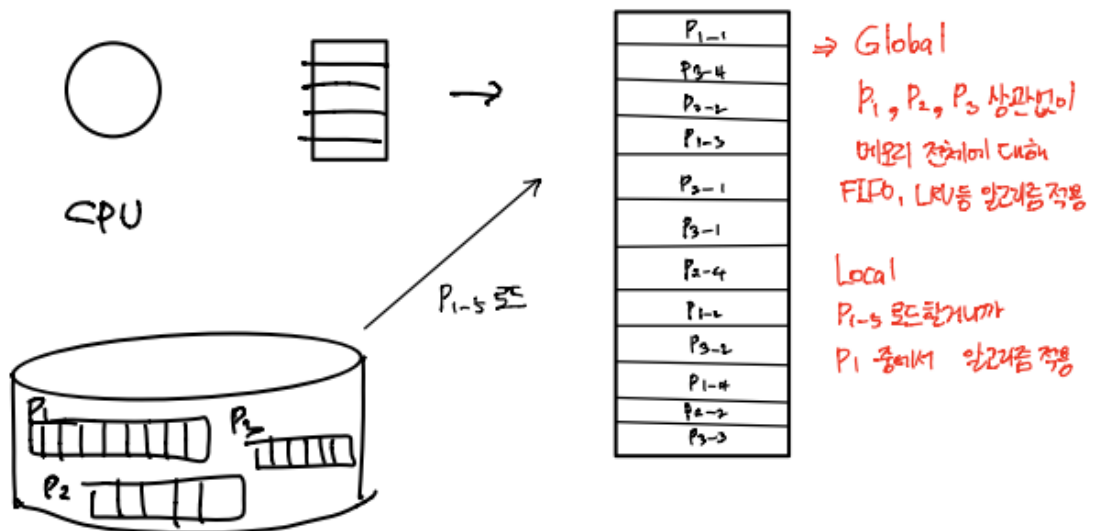
LRU (Least Recently Used)

- 규칙: 가장 오랫동안 사용되지 않은 페이지를 victim으로 선택
 - idea: 책상에 사용되지 않으면 나중에도 사용되지 않을 것
- 예제
 - 페이지 참조열: 7 0 1 2 0 3 0 4 2 3 0 3 2 1 2 0 1 7 0 1
 - 프레임 수: 3
 - page fault: 12

page	7	0	1	2	0	3	0	4	2	3	0	3	2	1	2	0	1	7	0	1
Frame	7	0	1	2	0	3	0	4	2	3	0	3	2	1	2	0	1	7	0	1
		7	0	1	2	0	3	0	4	2	3	0	3	2	1	2	0	1	7	0
			7	0	1	2	2	3	0	4	2	2	0	3	3	1	2	0	1	7
PF	0	0	0	0	X	0	X	0	0	0	0	X	X	0	X	0	X	0	X	X

↓ not 사용된지 제일 먼저
victim = 7
 ↓ 0 가장 최근 사용
 ↓ 1 사용된지 제일 먼저
victim = 1

Global vs Local Replacement



- Global Replacement
 - 메모리 상의 모든 프로세스 페이지에 대해 교체

- Local Replacement
 - 메모리 상의 자기 프로세스 페이지에 대해 교체
- 성능 비교
 - Global replacement가 더 효율적일 수 있다.