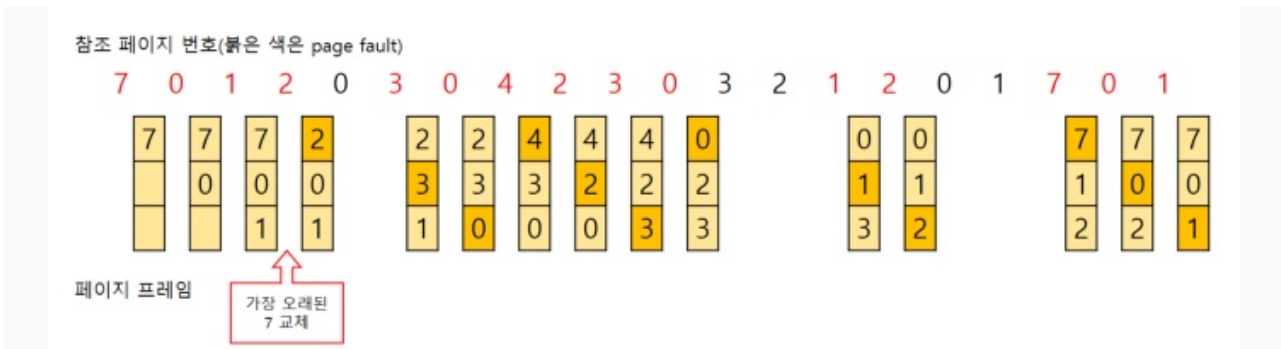


## 페이지 교체 알고리즘

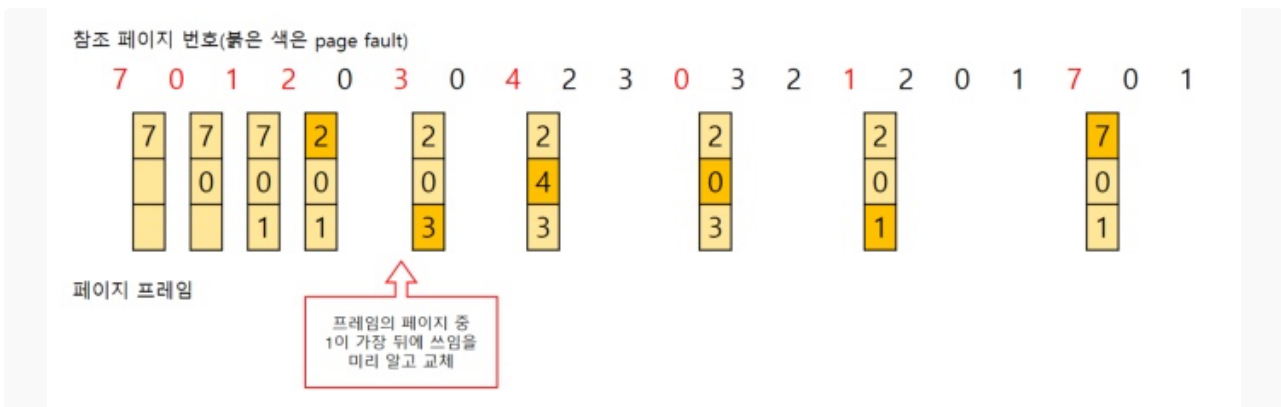
페이지 교체 알고리즘은 메모리 관리에서 페이지 부재(page fault)가 발생할 때 어떤 페이지를 교체할지를 결정하는 알고리즘입니다. 페이지 부재란 프로세스가 필요한 페이지를 메모리에 찾지 못하고 디스크에서 가져와야 할 때 발생합니다.

### FIFO(First-In, First-Out)



가장 먼저 들어온 페이지를 우선적으로 교체합니다. 구현이 간단하며, 페이지의 시간적 특성을 반영하기 때문에 예측성이 높습니다. 하지만 최신 페이지가 교체되지 않는 문제점이 있습니다.

### 최적 페이지 교체(Optimal Page Replacement)



미래에 가장 오랫동안 사용하지 않을 페이지를 교체합니다. 최적의 교체 알고리즘이지만 구현이 불가능할 정도로 많은 예측이 필요합니다.

### LRU(Least Recently Used)

참조 페이지 번호(붉은 색은 page fault)

7 0 1 2 0 3 0 4 2 3 0 3 2 1 2 0 1 7 0 1

7	7	7	2	2	4	4	4	0	1	1	1
	0	0	0	0	0	0	3	3	3	0	0
		1	1	3	3	2	2	2	2	2	7

페이지 프레임

가장 오랫동안 사용  
하지 않은 1 교체

가장 오랫동안 사용하지 않은 페이지를 교체합니다. FIFO 알고리즘의 단점을 보완한 알고리즘으로, 실제로 가장 많이 사용됩니다. 하지만 페이지 참조 시간을 기록해야 하기 때문에 구현이 복잡합니다.

## LFU(Least Frequently Used)

참조 페이지 번호(붉은 색은 page fault)

7 0 1 2 0 3 0 4 2 3 0 3 2 1 2 0 1 7 0 1

7	7	7	2	2	4	4	3	1	7	1
	0	0	0	0	0	0	0	0	0	0
		1	1	3	3	2	2	2	2	2

페이지 프레임

각각 한 번씩 참조한  
페이지 2, 3 중 LRU에 따라  
더 오래 쓰지 않은 2 교체

사용 빈도가 가장 적은 페이지를 교체합니다. 최적 알고리즘과 비슷하게 동작하지만, 사용 횟수를 기록하고 있어야 하기 때문에 메모리 사용량이 많습니다.