

# 가상 메모리

# 왜 동작하지?

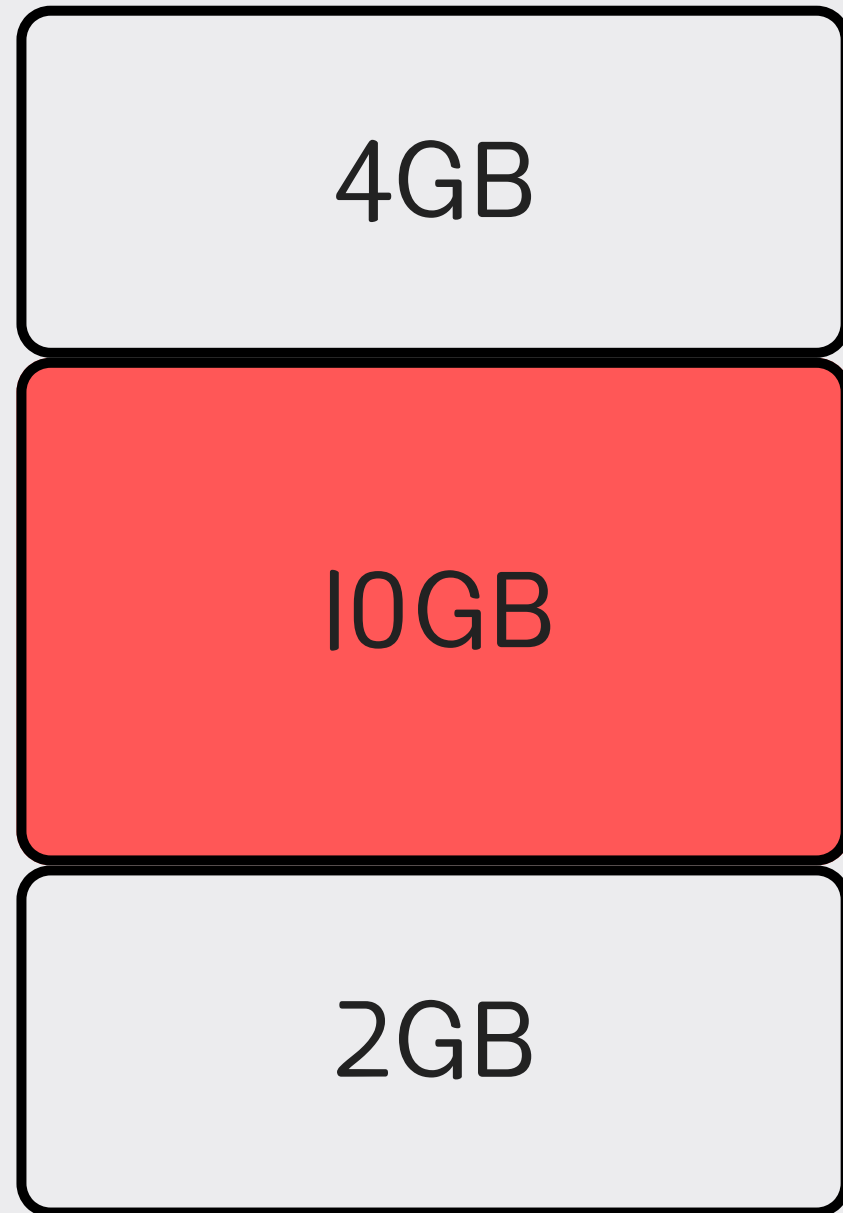
RAM이 16GB 인데

어떻게 수십 개의 프로그램을 동시에 열 수 있을까?

# 가상 메모리

각 프로세스가 독립된 메모리 공간을 가지고 있다는 착각

# 등장한 이유



RAM

- 물리 메모리 보다 크면 실행 불가
- 같은 메모리 쓰면 충돌
- 메모리 흠쳐볼 수 있음

# 페이징

페이징

“가상 메모리를 페이지로 나누어 다루는 기법”

# 간단한 흐름



가상 메모리

페이지 주소	프레임 주소

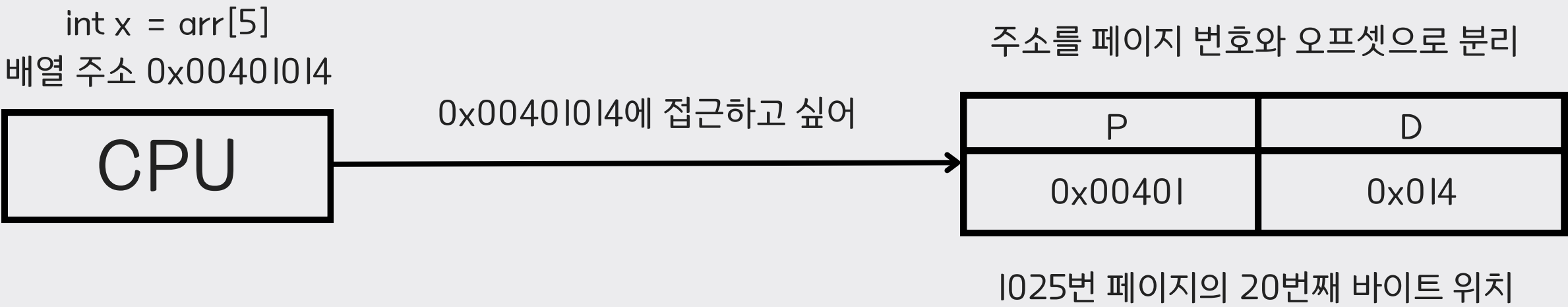
페이지 테이블



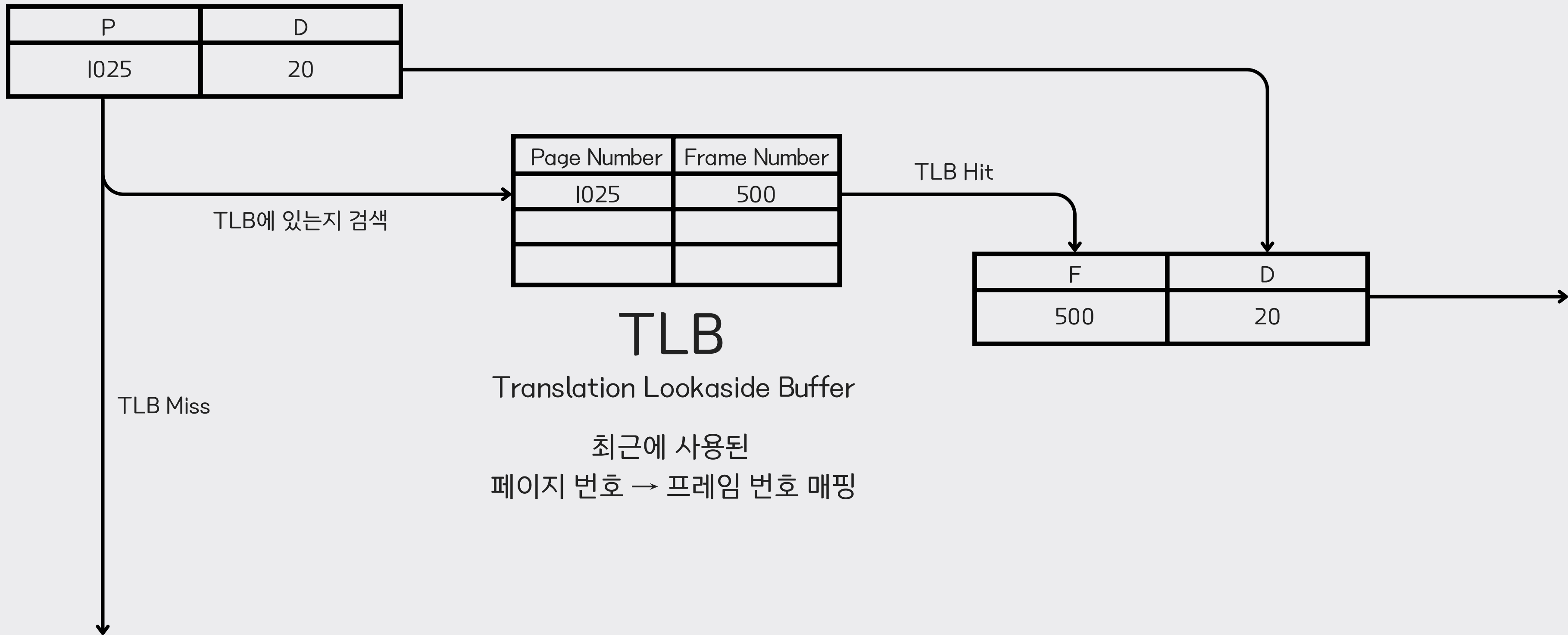
물리 메모리

프로세스

# 가상 주소 생성

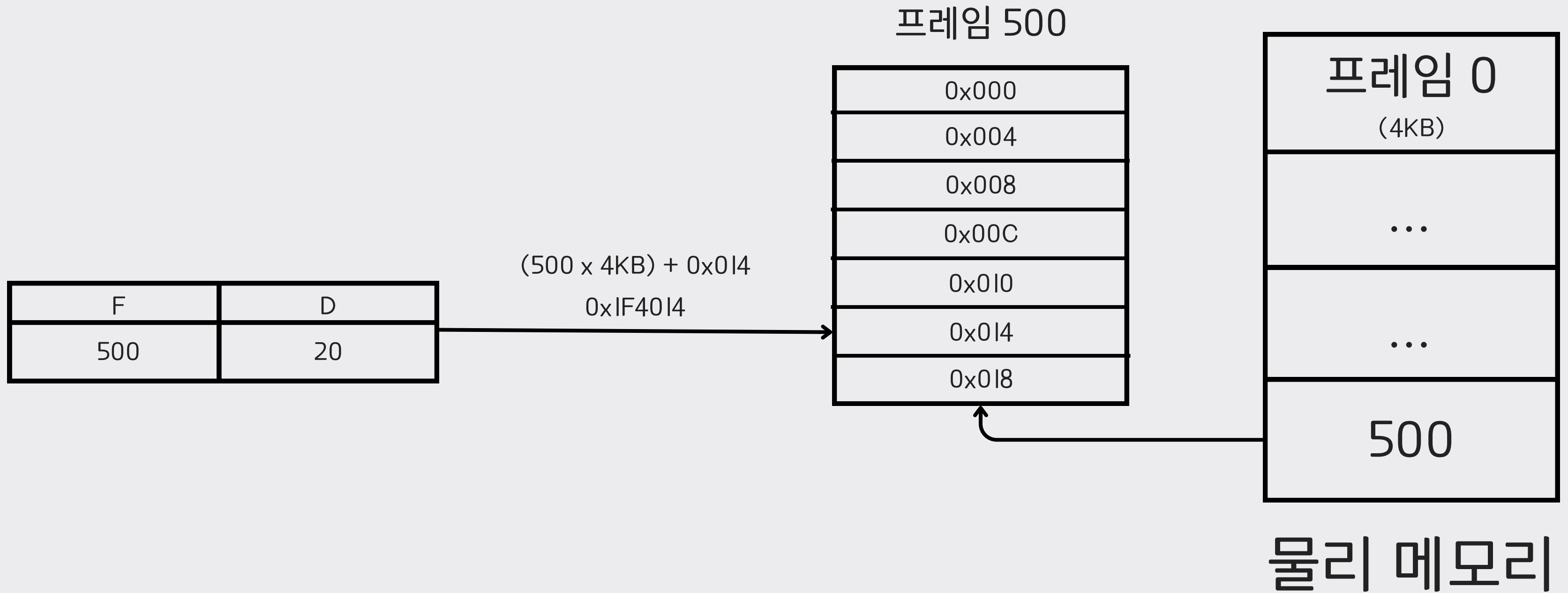


# TLB 확인

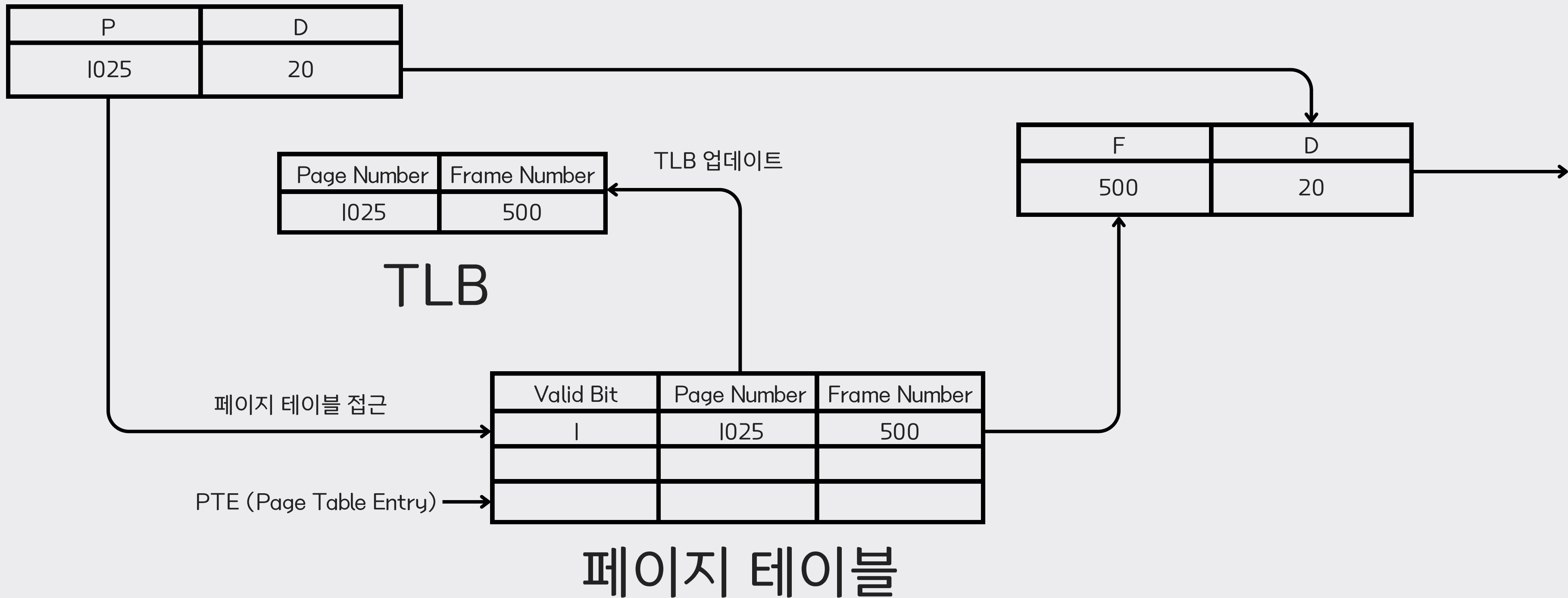




# TLB Hit



# TLB Miss



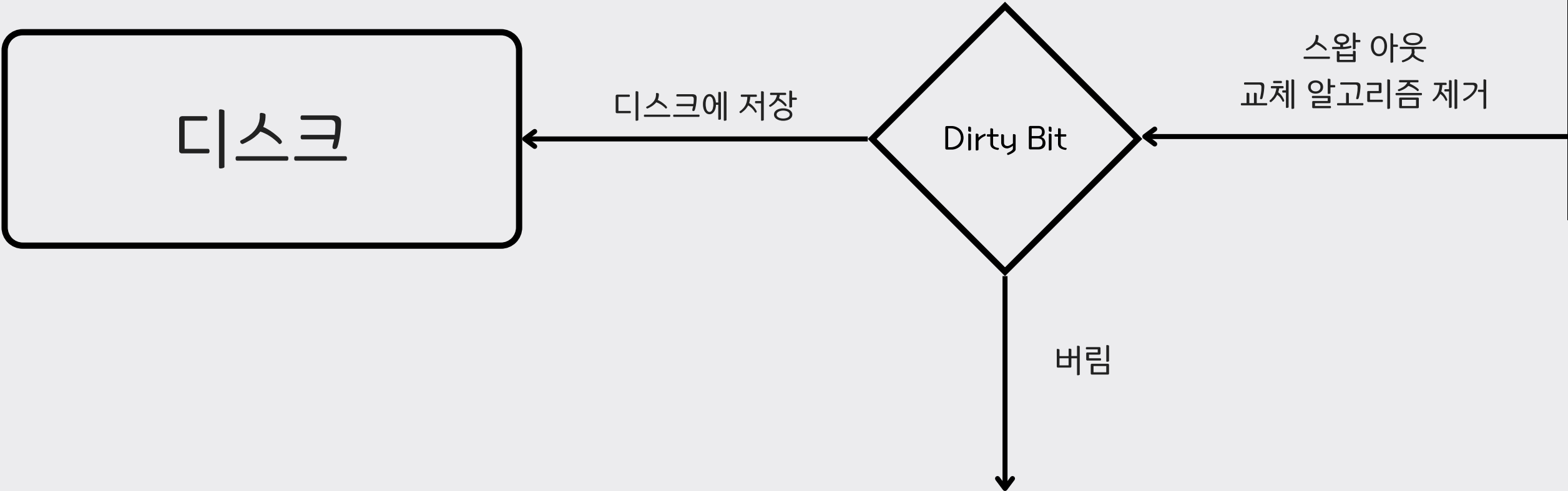
# 페이지 폴트

Valid Bit	Page Number	Frame Number
0	1025	500

페이지 테이블



물리 메모리



# 페이지 폴트

페이지 테이블 업데이트

Valid Bit	Page Number	Frame Number
1	1025	3

페이지 테이블

필요한 페이지 읽음



스왑 인

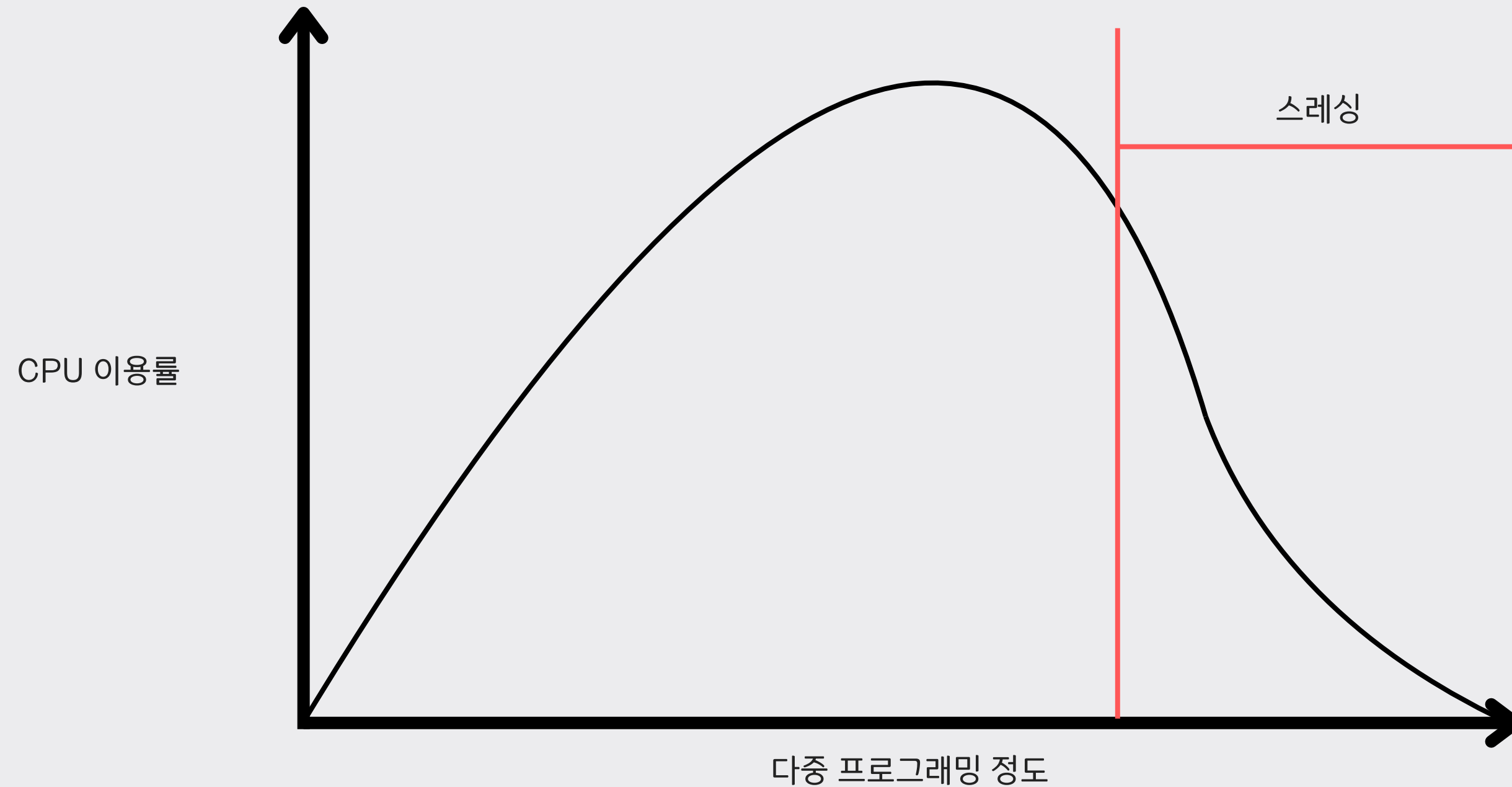


물리 메모리

# 요구 페이징

메모리를 절약하고, 효율적으로 관리할 수 있다.

# 스레싱



CPU가 실제 작업보다 페이지 교체에 더 많은 시간을 쓰는 상태

# 마무리

필요한 것만, 필요할 때 로드한다

# 더 알고 싶다면

페이지 교체 알고리즘

내부/외부 단편화

세그멘테이션

페이지 테이블 구조

워킹 셋



**Q&A**