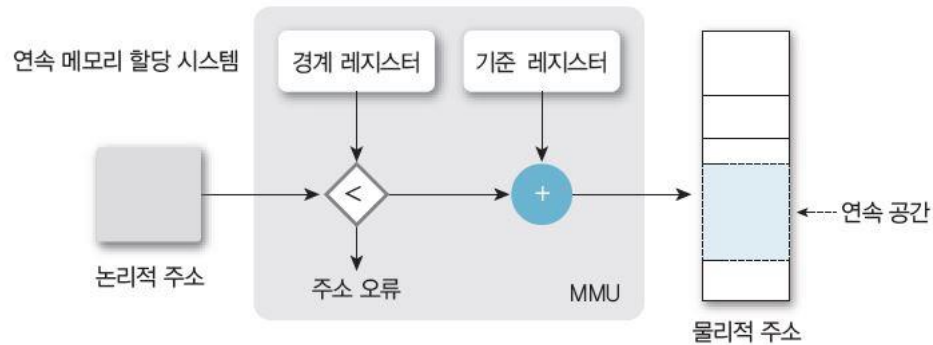


1

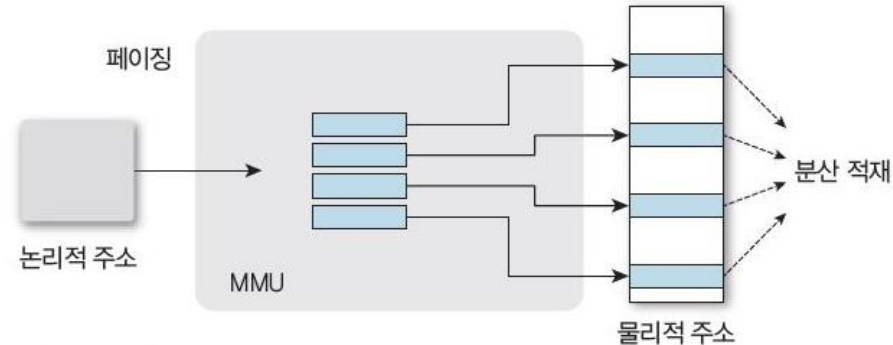
페이징 기법

1 페이징의 개념

- ▶ 작업을 크기가 동일한 페이지로 나눠 처리하는 방법
- ▶ 연속 메모리 할당과 비연속 메모리 할당 예



(a) 연속 메모리 할당



(b) 페이징 시스템을 이용한 비연속 메모리 할당

※ 출처: 그림으로 배우는 구조와 원리 운영체제, 구현회, 한빛아카데미, 2016

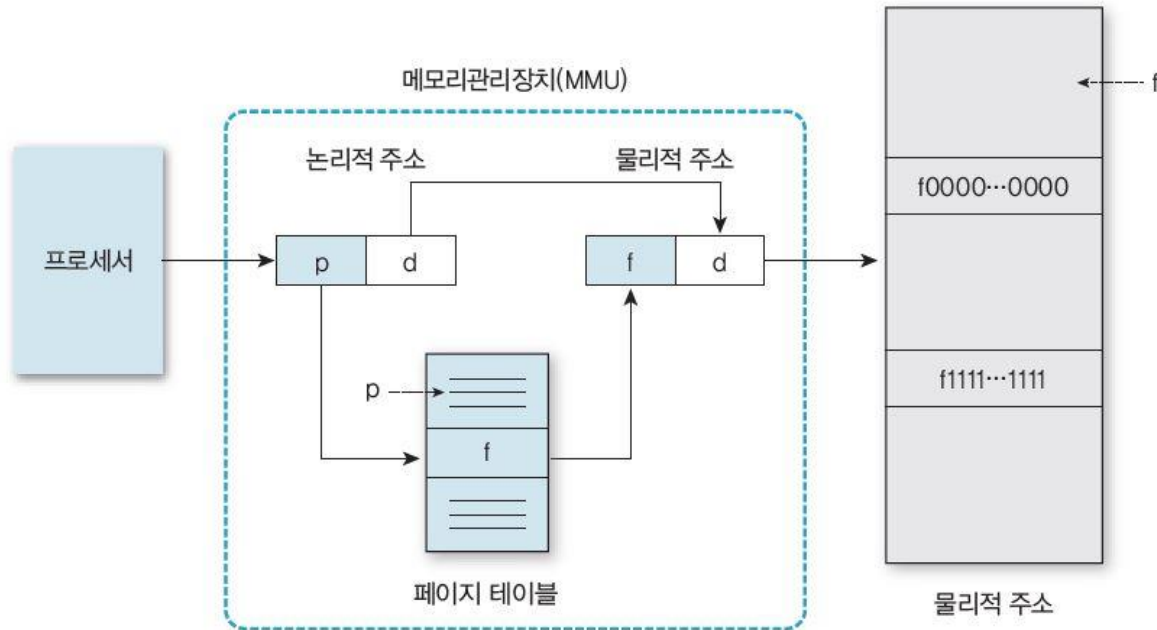
2 페이징 시스템에서 프로그램 실행 준비

- ▶ 프로세스에 필요한 페이지를 결정하여 페이지 번호 부여
- ▶ 메모리의 빈 프레임을 조사하여 프로세스를 적재할 위치 파악
- ▶ 프로세스의 페이지를 빈 프레임에 적재하도록 준비

3 페이징의 특징

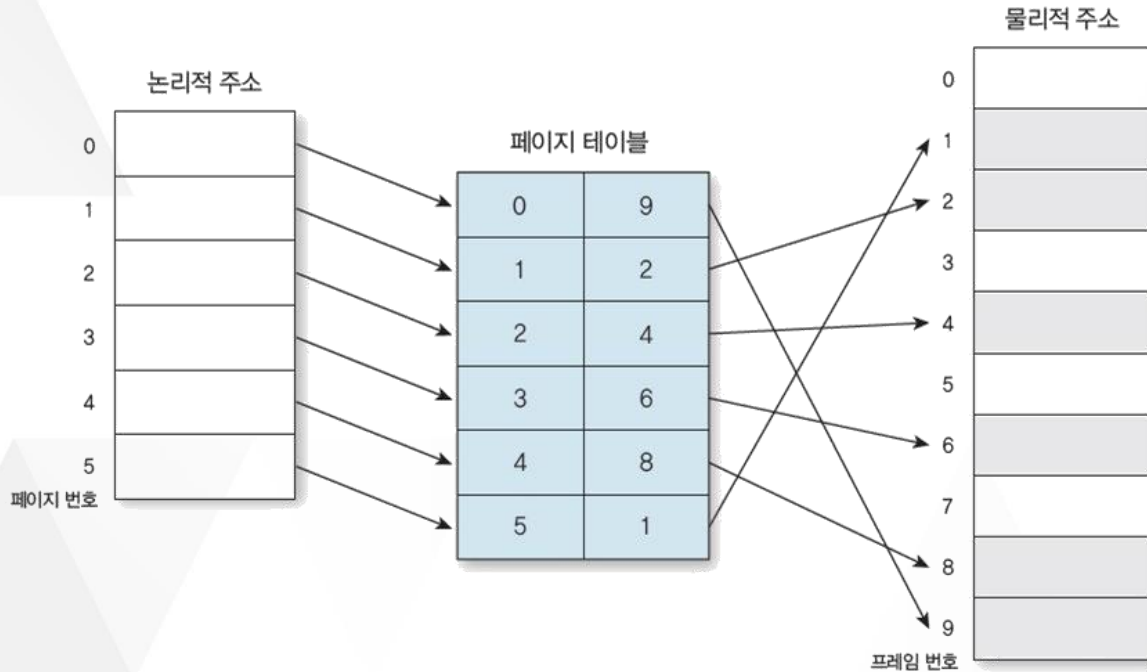
- ▶ 빈 프레임에 어떤 페이지든 적재할 수 있어 메모리 효율적 사용
- ▶ 프레임 간에 외부 단편화도 발생하지 않음
- ▶ 한 프로세스의 페이지를 메인 메모리의 여러 위치에 분산 적재하여 운영체제의 페이지 관리 부담이 큼
- ▶ 프레임 단위로 적재하므로 어떤 프로세스에 필요한 공간이 페이지 크기와 맞지 않으면, 마지막 페이지에 할당된 프레임이 완전히 차지 않아 내부 단편화가 발생 가능

4 페이지 시스템의 하드웨어 구조



※출처: 그림으로 배우는 구조와 원리 운영체제, 구현회, 한빛아카데미, 2016

5 페이지 테이블과 페이징 모델-1



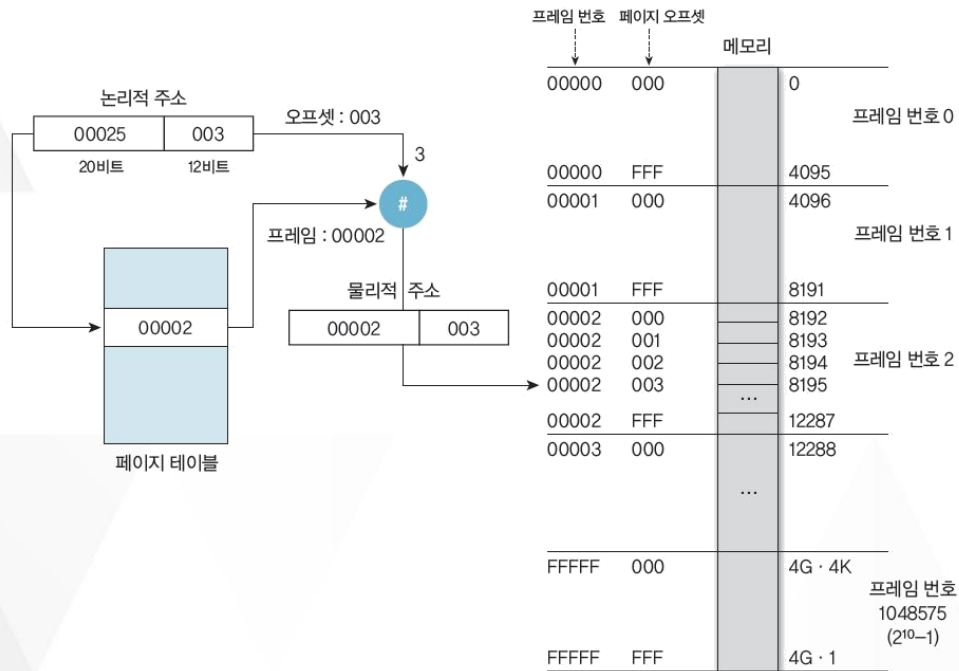
※ 출처: 그림으로 배우는 구조와 원리 운영체제, 구현회, 한빛아카데미, 2016

5 페이지 테이블과 페이징 모델-2

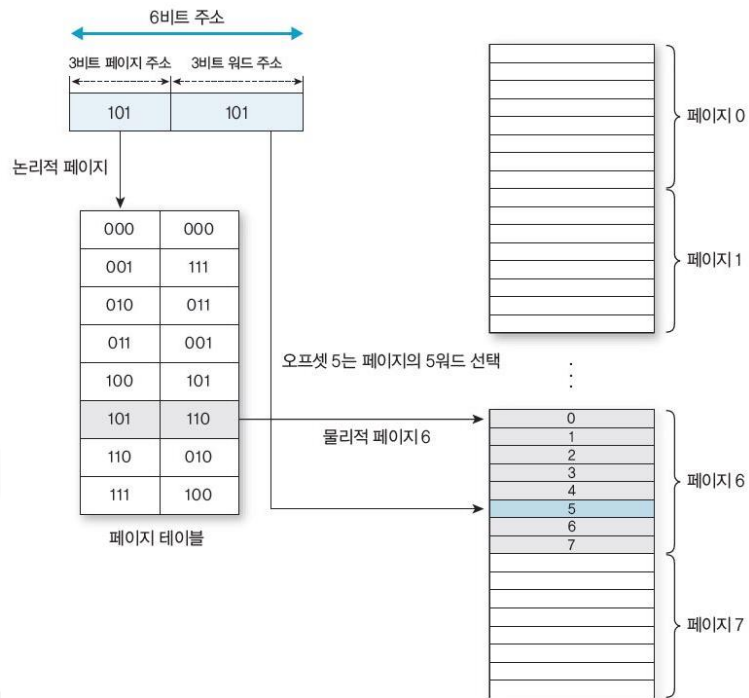


※출처: 그림으로 배우는 구조와 원리 운영체제, 구현회, 한빛아카데미, 2016

6 페이지 테이블을 이용한 물리적 주소 변환 예



※ 출처: 그림으로 배우는 구조와 원리 운영체제, 구현회, 한빛아카데미, 2016

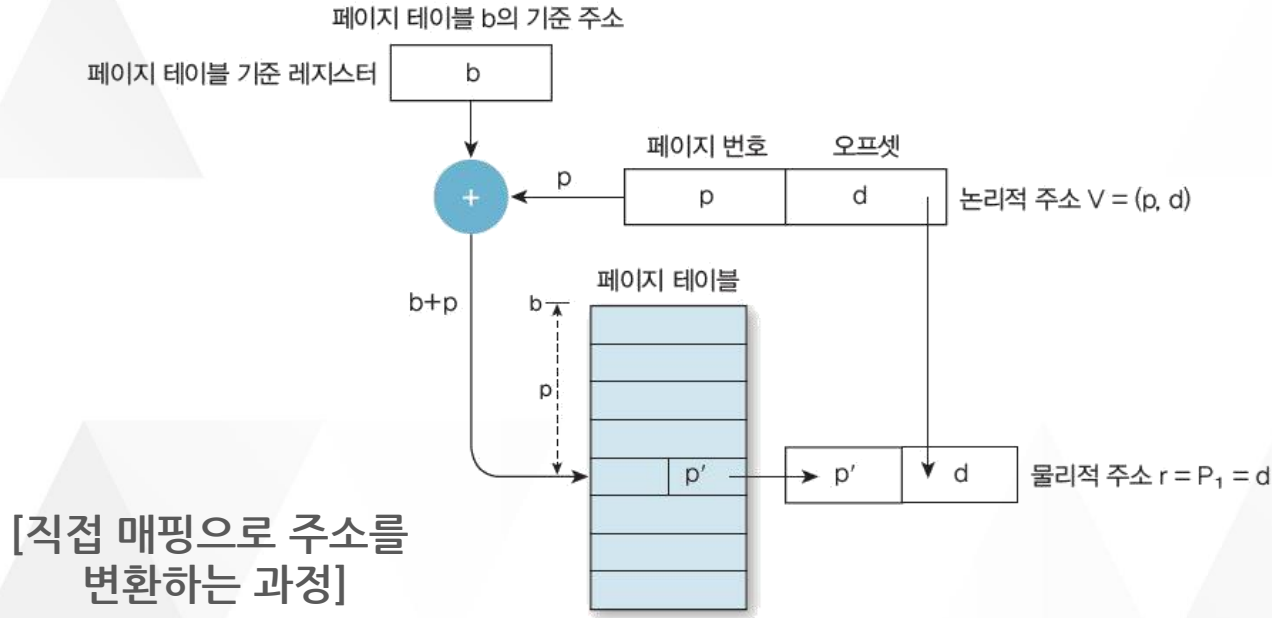


※ 출처: 그림으로 배우는 구조와 원리 운영체제, 구현회, 한빛아카데미, 2016

8 직접 매핑으로 주소 변환

- ▶ 메모리나 캐시에 완전한 페이지 테이블을 유지
- ▶ 프로세스의 메모리를 구성하는 모든 페이지 항목이 페이지 테이블에 있음
- ▶ 직접 매핑으로 주소를 변환하는 과정

8 직접 매핑으로 주소 변환



※ 출처: 그림으로 배우는 구조와 원리 운영체제, 구현회, 한빛아카데미, 2016

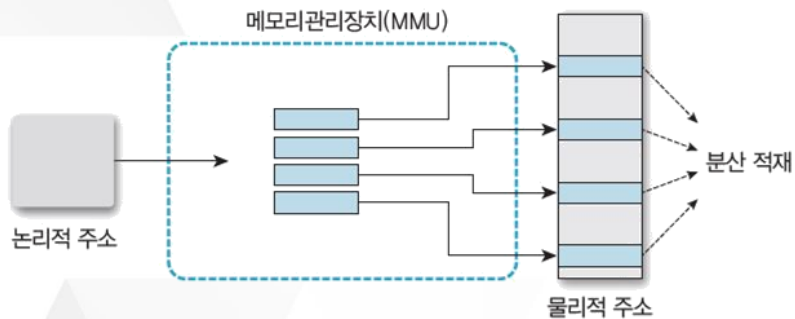
2 세그멘테이션 기법

1 세그멘테이션 개념

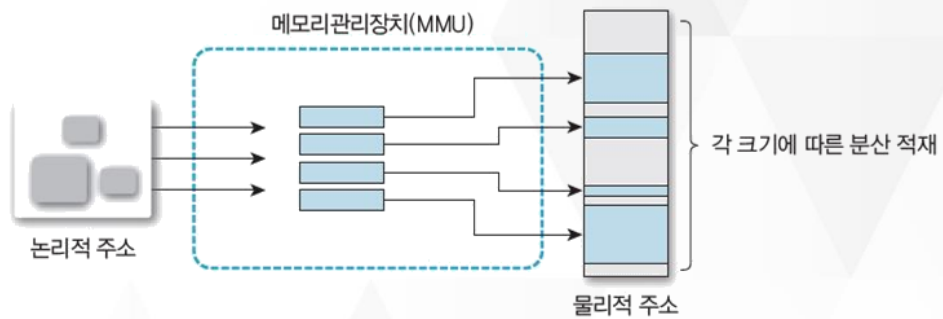
- ▶ 프로세스 관점을 지원하여 메모리를 크기가 변할 수 있는 세그먼트로 나누는 것
- ▶ 프로그램을 구성하는 서브루틴, 프로시저, 함수나 모듈 등으로 세그먼트 구성
- ▶ 각 세그먼트는 연관된 기능을 수행하는 하나의 모듈 프로그램으로 생각하며 메모리의 연속된 위치에서 구성하되 서로 인접할 필요는 없음

1 세그멘테이션 개념

- ▶ 하드웨어 보호 등 관리에 필요한 사항은 페이징과 비슷하거나 동일
- ▶ 프로세스에 따라 세그먼트 크기가 달라 메모리를 크기가 일정한 페이지 프레임으로 나누지 않고 동적 분할(가변 분할) 방법으로 할당



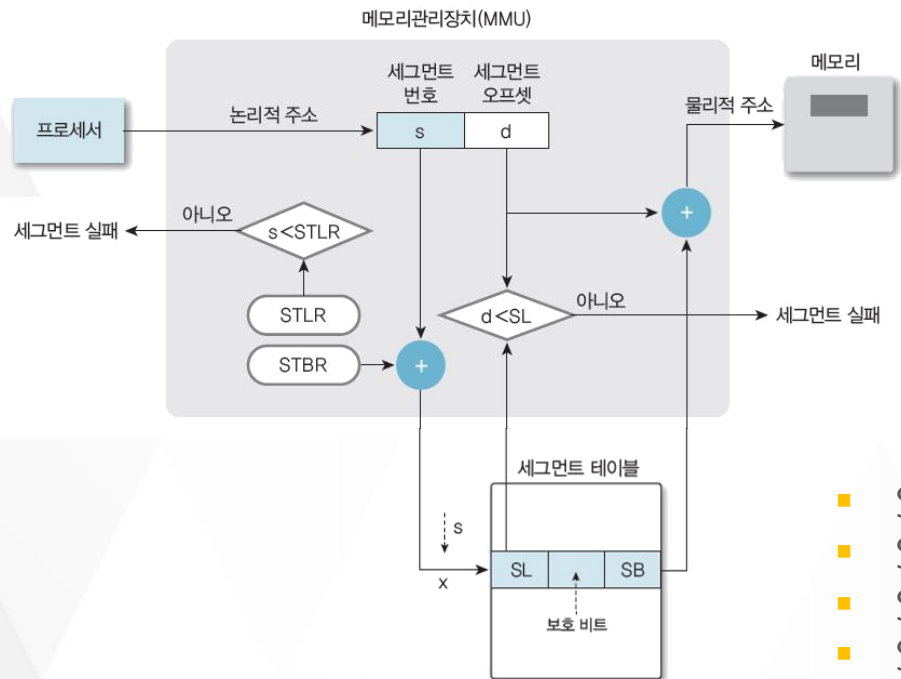
(a) 페이징



(b) 세그멘테이션

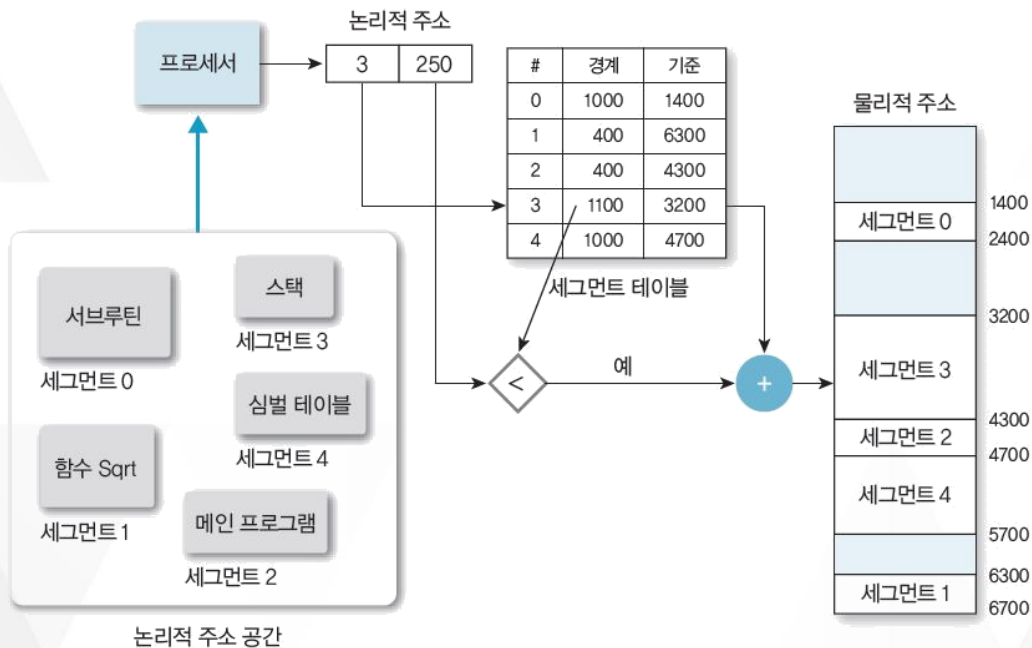
※출처: 그림으로 배우는 구조와 원리 운영체제, 구현회, 한빛아카데미, 2016

3 세그먼트 하드웨어 시스템의 구조



- SL : 세그먼트 경계(길이)
- SB : 세그먼트 기준
- STLR : 세그먼트 테이블 기준 레지스터
- STBR : 세그먼트 테이블 경계 레지스터

※ 출처: 그림으로 배우는 구조와 원리 운영체제, 구현회, 한빛아카데미, 2016



※ 출처: 그림으로 배우는 구조와 원리 운영체제, 구현회, 한빛아카데미, 2016

3

페이징과 세그멘테이션 비교

3 페이징과 세그멘테이션 비교

1 페이징과 세그멘테이션 비교

- ▶ 프로그램을 나눈 모든 세그먼트에서 메모리의 빈 공간을 찾아 할당하는 것이 페이징과 비슷함
- ▶ 페이징과 달리 프로그램을 나누는 크기가 변함
- ▶ 가변 크기 분할 방법처럼 세그멘테이션도 보통 최적 적합 알고리즘이나 최초 적합 알고리즘으로 해결하는 동적 메모리 할당 방법 이용
- ▶ 따라서 외부 단편화가 일어날 수 있는데, 사용 가능한 메모리의 모든 블록이 너무 작아서 세그먼트를 수용할 수 없을 때 발생하며, 대부분 기다리거나 압축하여 더 큰 공간 만듦

3 페이징과 세그멘테이션 비교

1 페이징과 세그멘테이션 비교

- ▶ 세그멘테이션은 동적 대치 알고리즘이므로 원할 때마다 메모리 압축 가능
- ▶ 프로세서 큐에서 몇 번 건너뛰어 크기가 작고 우선권이 더 낮은 프로세스 찾을 수도 있음
- ▶ 외부 단편화 문제는 대부분 평균 세그먼트 크기에 의존하며 일반적으로 평균 세그먼트 크기가 작으면 외부 단편화 또한 작음
- ▶ 페이징은 물리적 주소 없이도 큰 가상 주소 공간이 가능하게 하려고 등장하며 세그멘테이션은 프로그램과 데이터를 논리적으로 독립된 주소 공간으로 나누고 쉽게 공유 및 보호할 수 있게 하려고 등장

3 페이징과 세그멘테이션 비교

2 페이징과 세그멘테이션 비교

| 비교 | 페이징 | 세그멘테이션 |
|-------------------------|-----|--------|
| 프로그래머의 기술 인식 | 아니오 | 예 |
| 선형 주소 공간 | 1 | 많음 |
| 물리적 주소를 초과하는 총 주소 공간 | 예 | 예 |
| 프로시저와 데이터 분리, 보호 여부 | 아니오 | 예 |
| 수용할 수 있는 테이블 크기가 심하게 변화 | 아니오 | 예 |
| 사용자 공유가 용이 | 아니오 | 예 |

※ 출처: 그림으로 배우는 구조와 원리 운영체제, 구현회, 한빛아카데미, 2016