

운영체제 – 트래싱(thrashing)

1871227 임세진

<https://youtu.be/zEjOX88VU0s>

Contents

01. 메모리 관리 기법 - 페이징

02. 가상 메모리

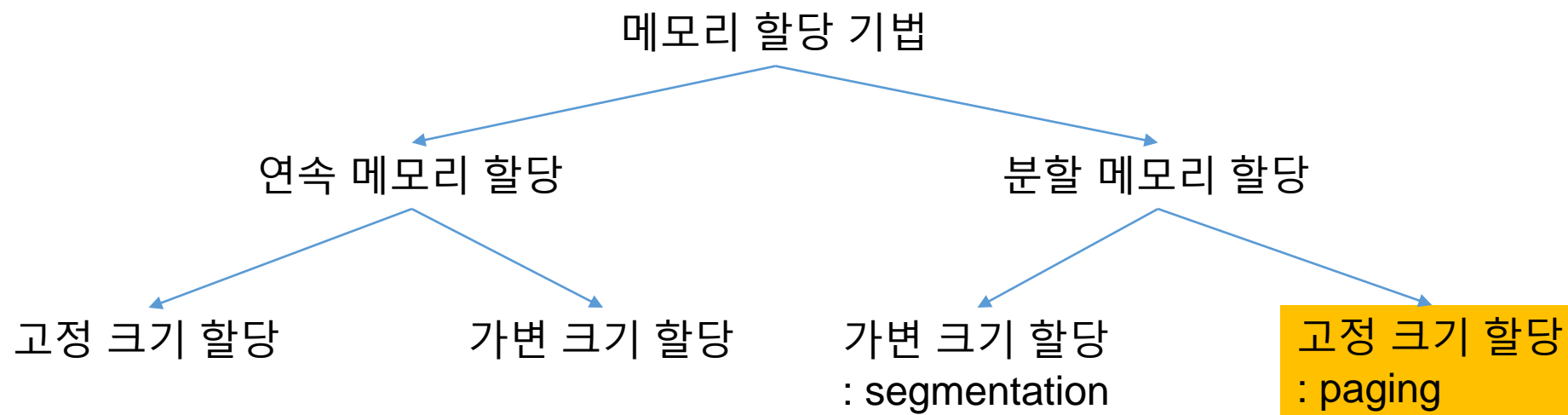
03. 스래싱(thrashing)



1. 메모리 관리 기법 - 페이징

1. 메모리 관리 기법 - 페이징

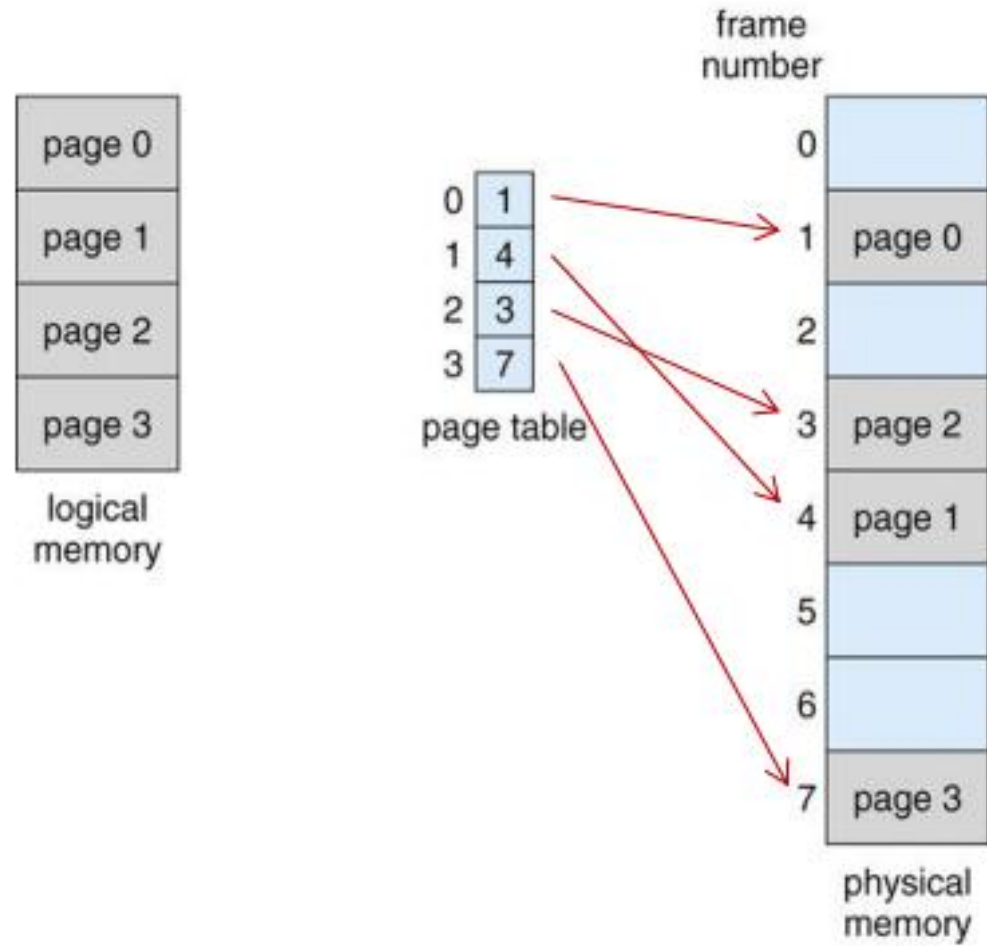
- 메모리 관리 기법



1. 메모리 관리 기법 - 페이징

- 페이징 (paging)
 - 프로세스의 주소공간을 페이지(page)라는 고정 크기로 나누고 물리 메모리도 프레임(frame)이라 불리는 페이지와 동일한 크기의 블록들로 분할하여, 프로세스의 각 페이지를 물리 메모리상의 임의의 빈 프레임에 할당하는 기법
 - 프로세스 주소공간을 모두 동일한 크기의 페이지로 나누기 때문에, 하나의 페이지에는 코드, 데이터, 스택이 섞여 있을 수 있음
 - 세그멘테이션 기법보다 구현이 쉽고, CPU에 의존적이지 않기 때문에 (높은 이식성) 다양한 컴퓨터 시스템에서 쉽게 구현이 가능하고, 메모리 활용 또한 높기 때문에 주로 이 기법을 사용함

1. 메모리 관리 기법 - 페이징

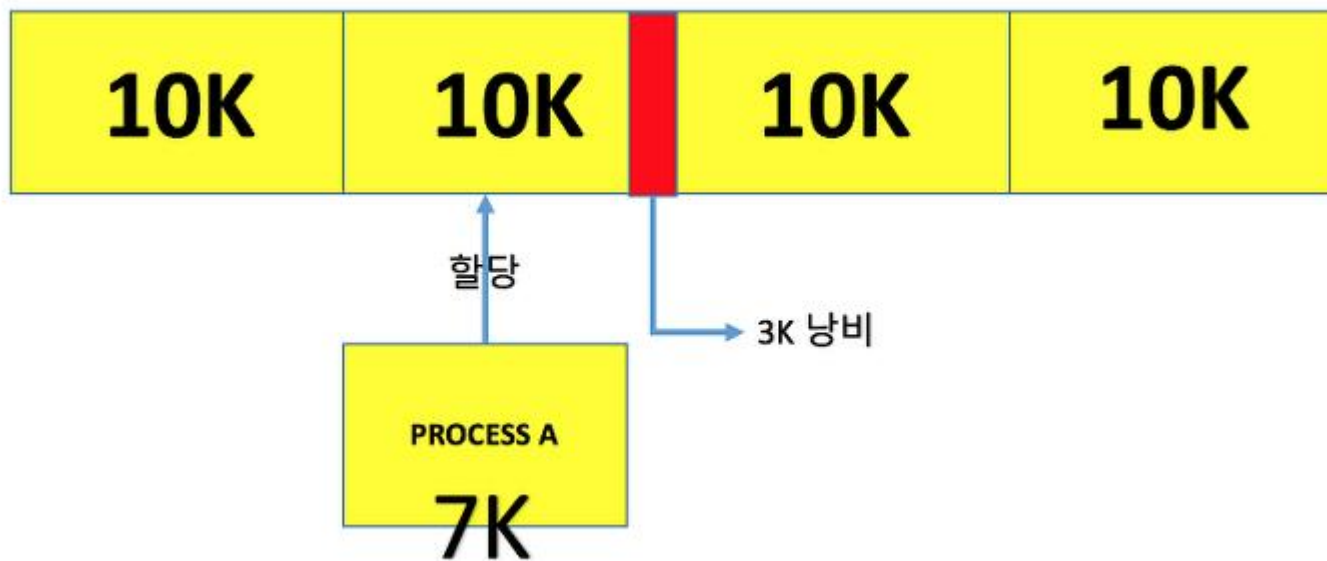


1. 메모리 관리 기법 - 페이징

- 단편화(fragmentation)
 - 프로세스에게 할당해줄 수 없는 조각난 메모리들이 생기는 현상
 - 조각난 메모리를 홀(hole)이라고 부르며, 홀이 너무 작아 프로세스에게 할당할 수 없을 때 단편화가 발생하게 됨
 - 홀이 생기는 위치에 따라 내부 단편화와 외부 단편화로 나뉨
 - 어떤 메모리 할당 정책이든지 단편화는 발생하므로, 단편화로 인한 메모리 낭비를 줄이는 메모리 할당 정책이 좋은 정책임

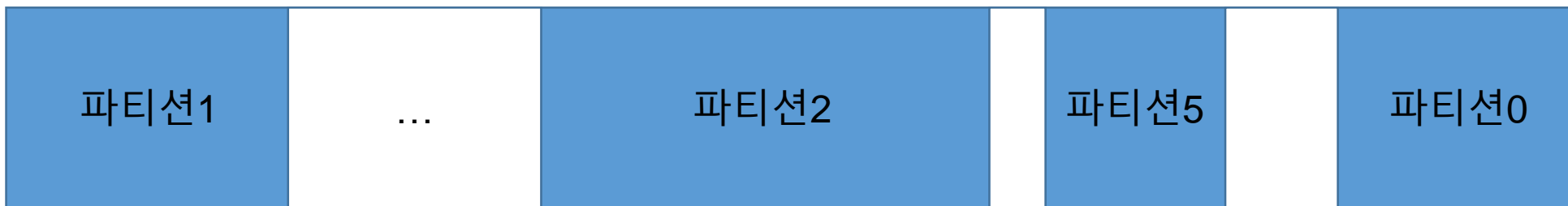
1. 메모리 관리 기법 - 페이징

- 내부 단편화(internal fragmentation)
 - 프로세스에게 할당된 메모리 내부에 작은 홀이 생기는 경우임
 - 고정된 크기로 메모리를 할당하는 정책에서 발생함



1. 메모리 관리 기법 - 페이징

- 외부 단편화(external fragmentation)
 - 메모리가 할당되고 반환되는 과정에서 할당된 메모리들 사이에 작은 홀이 생기는 경우
 - 가변 크기의 메모리 할당 정책에서 나타남

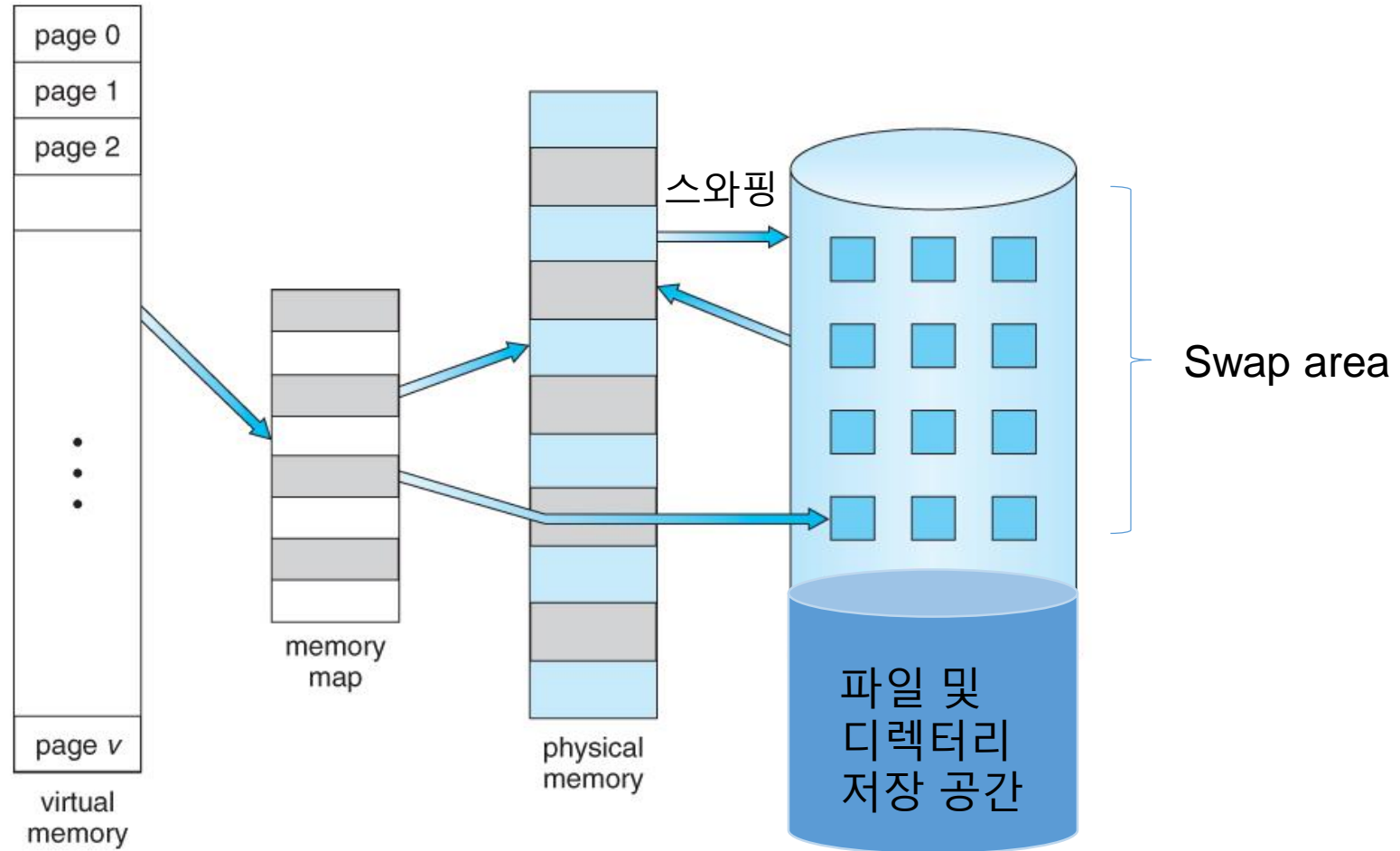


2. 가상 메모리

2. 가상 메모리

- 가상 메모리(virtual memory)
 - 물리 메모리가 가진 크기 한계를 극복하고, 물리 메모리보다 큰 프로그램이나 여러 프로그램을 동시에 실행시킬 수 있도록 하여 사용자가 무한대의 메모리가 있다고 느끼도록 하는 메모리 관리 기법
 - 프로세스가 실행되기 위해서는 프로세스의 코드, 데이터, 스택, 힙 모든 영역이 전부 물리 메모리에 할당되어 있어야 한다는 전제를 완전히 깬 것
 - 프로세스의 영역을 보조 기억 장치(주로 하드 디스크)의 영역으로 확장함

2. 가상 메모리



2. 가상 메모리

- 요구 페이징(demand paging)

= 페이징 + 스와핑

- 페이징 기법을 토대로 프로세스의 일부 페이지를 물리 프레임에 할당하여 실행시키고, 프로세스의 실행 중 페이지가 필요할 때 메모리 프레임을 할당받고 하드 디스크로부터 페이지를 로딩시키는 메모리 관리 기법

2. 가상 메모리

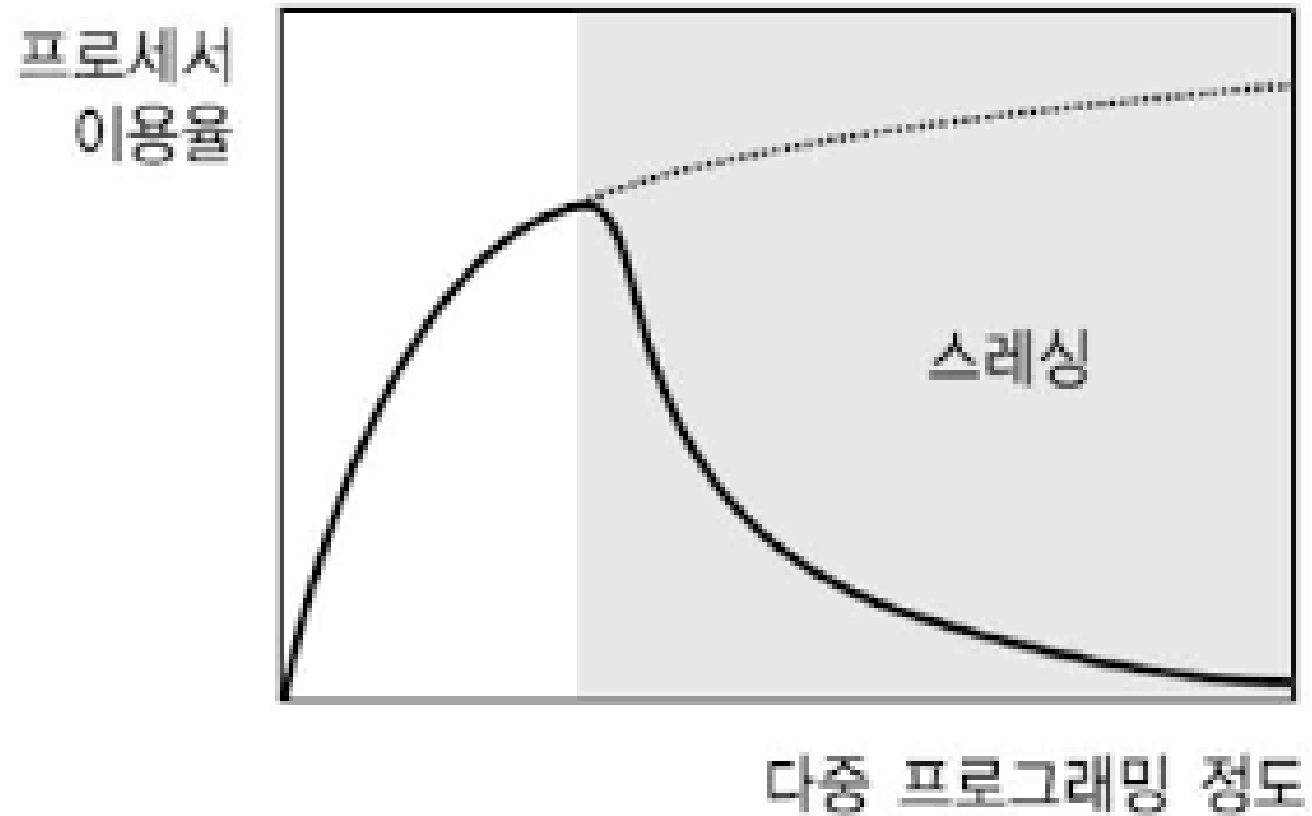
- 페이지 폴트(page fault)
 - 프로세스의 실행 도중 CPU가 액세스하는 주소의 페이지가 물리 메모리에 존재하지 않아 더 이상 실행을 할 수 없을 때 페이지 폴트라고 함
 - 페이지 폴트가 발생하면 커널에 의해 하드 디스크에서 페이지를 로딩하게 됨
 - 만약 물리 메모리에 공간이 있다면 빈 프레임을 할당받아 하드 디스크에서 페이지를 로딩하게 되고, 빈 프레임이 없다면 알고리즘에 의해 프레임을 교체하게 됨

3. 트래싱 (thrashing)

3. 스래싱(thrashing)

- 스래싱(thrashing)
= ‘빈번한 페이지 폴트와 디스크 입출력 증가 현상’
 - 스래싱은 페이지 폴트가 계속적으로 발생하여 메모리 프레임에 페이지가 반복적으로 교체되는 현상
 - 이 과정에서 디스크 입출력이 심각하게 증가하며 CPU는 페이지 입출력을 기다리는데 대부분의 시간을 보내게 되어 CPU 활용률이 대폭 감소하게 됨

3. 스래싱(thrashing)



3. 스래싱(thrashing)

- 스래싱의 원인

- 메모리 량에 비해 실행중인 프로세스의 개수가 과한 경우

: 실행되는 프로세스의 개수가 많아지면 평균적으로 각 프로세스에게 할당되는 메모리 프레임의 개수가 작아져 새 페이지 로딩을 위해 곧이어 필요하게 될 페이지를 스왑-아웃시킬 확률이 높아지기 때문

- 메모리 할당 정책이 잘못된 경우

- 컴퓨터 시스템에 설치된 메모리가 너무 작은 경우

- 특정 시간에 너무 많은 프로세스가 실행되는 경우

3. 스래싱(thrashing)

- 스래싱 해결
 - 스래싱이 발생했다면 몇몇 프로세스를 강제로 종료시켜 다중 프로그래밍 정도를 낮추면 됨

감사합니다😊

