

## [OS 6강]-기억장치 관리기법(주, 가상) [출제빈도 '상']

### 1. 기억장치 관리 전략

#### 1) 반입(Fetch) 전략

- 보조기억장치의 프로그램이나 데이터를 **언제** 주기억장치로 적재할 것인지를 결정

#### 2) 배치(Placement) 전략

- 주기억장치의 **어디**에 위치시킬 것인지를 결정

① 최초 적합(First Fit) : 첫 번째 배치시키는 방법 (속도↑, 공간↓)

② 최적 적합(Best Fit) : 단편화를 가장 작게 남기는 분할 영역에 배치시키는 방법 (속도↓, 공간↑)

③ 최악 적합(Worst Fit) : 단편화를 가장 많이 남기는 분할 영역에 배치시키는 방법 (속도↓, 공간↓)

#### 3) 교체(Replacement) 전략

- 주기억장치의 모든 영역이 이미 사용중인 상태에서 주기억장치에 배치하려고 할 때, 이미 사용되고 있는 영역 중에서 **어느** 영역을 교체하여 사용할 것인지를 결정 (FIFO, OPT, LRU, LFU, NUR, SCR)

[기출문제]

First Fit, Best Fit, Worst Fit 방법에 대해서 10K 프로그램이 할당될 부분?

영역 1	9K
영역 2	15K
영역 3	10K
영역 4	30K

---> First Fit

---> Best Fit

---> Worst Fit

★★★★☆

\*단편화 (fragmentation)  
- 내부 : 할당 후 남은 공간  
(15K->5K, 30K->20K)  
- 외부 : 할당하지 못한 공간  
(9K)

## [OS 6강]-기억장치 관리기법(주, 가상)

### 2. 단편화 해결 방법

★★★★☆

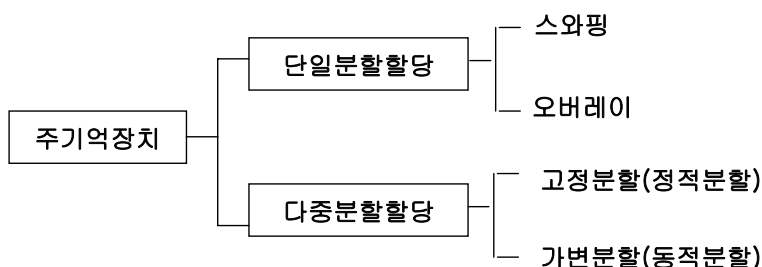
#### 1) 통합(Coalescing) 기법

: 주기억장치 내에 **인접**해 있는 단편화된 공간을 하나의 공간으로 통합

#### 2) 집약(Compaction) 기법, 압축, 쓰레기 수집(Garbage Collection)

: 주기억장치 내에 **분산**되어 있는 단편화된 빈 공간을 결합하여 하나의 큰 가용 공간을 만드는 작업

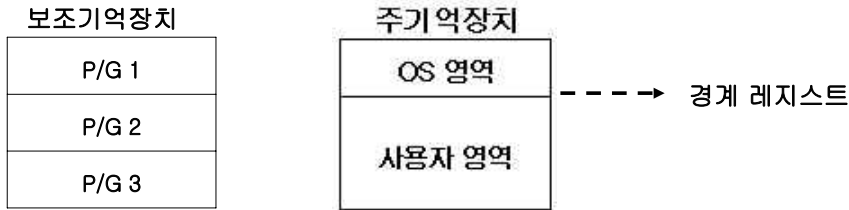
### 3. 주기억장치 할당 기법



## [OS 6강]-기억장치 관리기법(주, 가상)

### 4. 단일분할 할당(단일 프로그래밍) > 스와핑(Swapping)

: 하나의 프로그램 전체를 주기억장치에 할당하여 사용하다 필요에 따라 다른 프로그램과 교체하는 기법



### 5. 단일분할 할당(단일 프로그래밍) > 오버레이(Overlay) ★☆☆☆☆

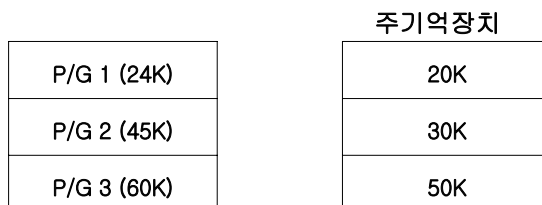
: 실행되어야 할 작업의 크기가 커서 사용자 기억 공간에 수용될 수 없을 때 작업의 모든 부분들이 동시에 주기억 장소에 상주해 있을 필요가 없다. 이때 작업을 분할하여 필요한 부분만 교체하는 방법



## [OS 6강]-기억장치 관리기법(주, 가상)

### 6. 다중분할 할당(다중 프로그래밍) > 고정 분할 ★☆☆☆☆

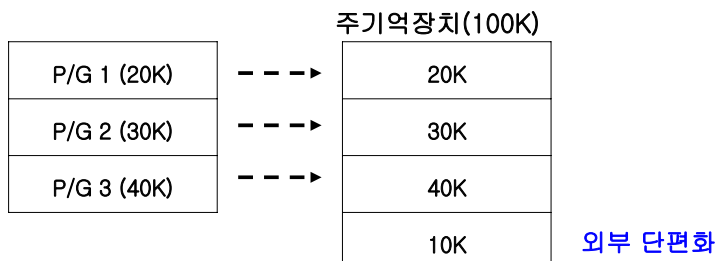
: 주기억장치를 미리 몇 개의 고정된 개수와 크기의 부분으로 분할하여 여러 개의 프로그램이 동시에 적재되어 실행되게 하는 방법



### 7. 다중분할 할당(다중 프로그래밍) > 가변 분할 ★☆☆☆☆

: 고정 분할 할당 기법의 단편화를 줄이기 위한 것으로, 미리 주기억장치를 분할해 놓는 것이 아니라 프로그램을 주기억장치에 적재하면서 필요한 만큼의 크기로 영역을 분할하는 기법

ex) 식당 전체 공간에서 칸막이를 이용해서 손님의 수에 따라 자리를 만들어 줌

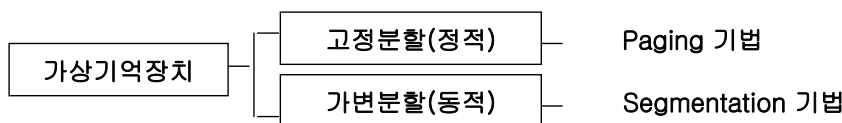


## [OS 6강]-기억장치 관리기법(주, 가상)

### 1. 가상기억장치 개요 ★☆☆☆☆

- 보조기억장치의 일부분을 주기억장치처럼 사용하는 것
- 용량이 적은 주기억장치를 마치 큰 용량이 있는 것처럼 사용하는 것
- 프로그램을 여러 개의 작은 블록으로 나누어서, 프로그램 실행 시 요구되는 블록만 주기억장치에 불연속적으로 할당하여 처리
- 주기억장치보다 용량이 큰 프로그램 실행하기 위해 사용
- 가상기억장치에 저장된 프로그램을 실행하려면 가상기억장치의 주소를 주기억장치의 주소로 변환하는 작업이 필요 (매핑)

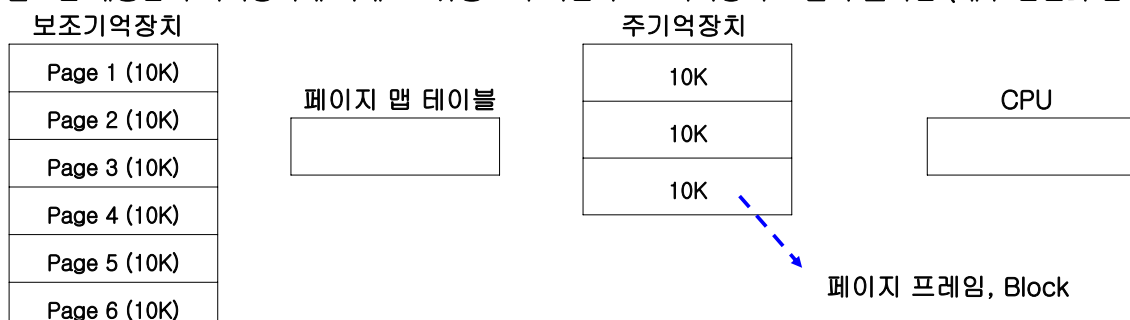
### 2. 가상기억장치 구현 기법



## [OS 6강]-기억장치 관리기법(주, 가상)

### 3. 페이징(Paging) 기법 ★

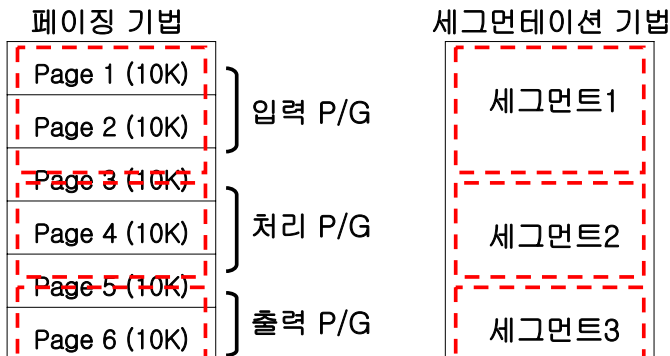
- 가상기억장치에 보관되어 있는 프로그램과 주기억장치의 영역을 동일한 크기로 나눈 후 나뉜 프로그램 (페이지)을 동일하게 나뉜 주기억장치의 영역(페이지 프레임)에 적재시켜 실행하는 기법
- 주소변환(Mapping) : 가상주소(보조기억장치)→실주소(주기억장치)
- 주소변환을 위해 페이지 맵핑 테이블(페이지 사상표)이 필요 => 기억장소 낭비
- 페이지 부재(Page Fault) : P/G 실행시 참조한 페이지가 주기억장치에 없는 현상
- 외부단편화(X), 내부단편화(O)
- \* 페이지 크기가 작을 경우 (10K → 1K)
- 페이지 수 증가 → 페이지 맵핑 테이블 커진다 → 맵핑 속도 느리고 기억 공간 낭비 발생
- 디스크 접근 횟수 증가 → 전체적인 입.출력 시간은 늘어남
- 한 개의 페이지를 주기억장치로 이동하는 시간이 줄어 듭
- 필요한 내용만 주기억장치에 적재 → 유용도가 커진다 → 기억장치 효율이 높아짐 (내부 단편화 감소)



## [OS 6강]-기억장치 관리기법(주, 가상)

### 4. 세그멘테이션(Segmentation) 기법 ★★☆☆☆

- 가상기억장치에 보관되어 있는 프로그램을 다양한 크기의 논리적인 단위로 나눈 후 주기억장치에 적재시켜 실행시키는 기법 => **메모리 절약**
- 논리적인 크기로 나눈 단위를 세그먼트라고 하며, 각 세그먼트는 고유한 이름과 크기를 갖고 있음
- 다른 세그먼트에게 할당된 영역을 침범할 수 없으며, 이를 위해 **기억장치 보호키(Storage Protection Key)**가 필요함
- 외부단편화(O), 내부단편화(X)



## [OS 6강]-기억장치 관리기법(주, 가상)

### 5. 가상기억장치의 성능에 영향을 미치는 요인 ★

#### 1) 워킹 셋 (Working Set)

- 프로세스가 일정 시간 동안 자주 참조하는 페이지들의 집합으로, 자주 참조되는 워킹 셋을 주기억장치에 상주시킴으로써 페이지 부재 및 페이지 교체 현상을 줄임

#### 2) 스래싱 (Thrashing)

- 프로세스의 처리 시간보다 페이지 교체 시간이 더 많아지는 현상 -> CPU 이용률 저하
- 페이지 오류율(page fault)이 크면 스래싱이 많이 일어난 것이다
- 다중 프로그래밍의 정도가 높을수록 스래싱의 발생 빈도는 높아진다
- 스래싱 방지 방법 : 다중 프로그래밍의 정도를 줄인다, CPU 이용률을 높인다, Working set 방법을 사용

#### 3) 구역성 (Locality, 국부성) : 참조국부성(locality of reference)

- 프로세스가 실행되는 동안 일부 페이지만 집중적으로 참조하는 성질

##### ① 시간 구역성

- 최근에 참조된 기억 장소가 가까운 장래에도 계속 참조될 가능성이 높음
- 예) Loop(반복), 스택, 부프로그램(Sub Routine), 카운팅(Counting), 집계(Totaling)에 사용되는 변수

##### ② 공간 구역성

- 하나의 기억 장소가 참조되면 그 근처의 기억 장소가 계속 참조될 가능성이 높음
- 예) **순차적 코드(수행) 실행, 배열 순회, 같은 영역에 있는 변수 참조**

## [OS 6강]-기억장치 관리기법(주, 가상)

1. 새로 들어온 프로그램과 데이터를 주기억장치 내의 어디에 놓을 것인가를 결정하기 위한 주기억장치 배치 전략에 해당하지 않는 것은?

- 가. Best Fit    나. Worst Fit  
다. First Fit    라. Last Fit

2. 그림과 같이 저장 장치가 배치되어 있을 때 13K의 작업이 공간의 할당을 요구하여 최악 적합(Worst-Fit) 전략을 사용한다면 어느 주소에 배치되는가?

- 가. b    나. d    다. f    라. h

a	OS 사용 공간
b	16K 공백
c	사용 중
d	14K 공백
e	사용 중
f	5K 공백
g	사용 중
h	30K 공백

3. 저장장치의 배치 전략 중에서 작업의 배치 결정을 가장 빨리 내릴 수 있는 방식은?

- 가. Best Fit    나. First Fit  
다. Worst Fit    라. Last Fit

4. 다음 설명이 의미하는 것은?

단일 사용자 시스템에서 프로그램의 크기는 주기억 장치의 용량보다 클 수는 없다. 그러나 사용하지 않는 프로그램의 부분을 보조기억장치로 옮겨와서 이제 더 이상 필요하지 않는 프로그램 부분이 사용하고 있던 장소를 다른 프로그램이 사용하게 하면 실제 영역보다 더 큰 프로그램의 실행이 가능하다.

- 가. 오버레이(Overlay)    나. 세그먼트(Segment)  
다. 페이지(Page)    라. 스레드(Thread)

[정답] 1.라 2.라 3.나 4.가



## [OS 6강]-기억장치 관리기법(주, 가상)

5. 기억 장치 관리에서 60K의 사용자 공간이 아래와 같이 분할되어 있다고 가정 할 때 24K, 14K, 12K, 6K의 작업을 최적 적합(Best Fit) 전략으로 각각 기억 공간에 들어온 순서대로 할당할 경우 생기는 총 내부 단편화(Internal Fragmentation)의 크기와 외부 단편화(External Fragmentation)의 크기는 얼마인가?

운영체제
25K
15K
10K
10K

- 가. 내부 단편화 4K, 외부 단편화 6K  
나. 내부 단편화 6K, 외부 단편화 8K  
다. 내부 단편화 6K, 외부 단편화 10K  
라. 내부 단편화 4K, 외부 단편화 12K

6. 주기억장치상에서 빈번하게 기억 장소가 할당되고 반납됨에 따라 기억 장소들이 조각들로 나누어지는 현상을 무엇이라고 하는가?

- 가. Compaction    나. Fragmentation  
다. Coalescing    라. Collision

7. 메모리 관리 기법 중에서 서로 떨어져 있는 여러 개의 낭비 공간을 모아서 하나의 큰 기억 공간을 만드는 작업을 무엇이라고 하는가?

- 가. Swapping    나. Coalescing  
다. Compaction    라. Paging

8. 다음 표는 고정 분할에서의 기억 장치 단편화 현상을 보이고 있다. 외부 단편화(External Fragmentation)은 총 몇 K인가?

	분할의 크기	작업의 크기
A	20K	10K
B	50K	60K
C	120K	160K
D	200K	100K
E	300K	150K

- 가. 480K    나. 430K    다. 260K    라. 170K

[정답] 5.다 6.나 7.다 8.라

## [OS 6강]-기억장치 관리기법(주, 가상)

9. 가상기억장치에 대한 설명으로 옳지 않은 것은?

- 가. 연속 배당 방식에서의 기억 장소 단편화 문제를 적극적으로 해결할 수 있다.
- 나. 기억 장치의 이용률과 다중 프로그래밍의 효율을 높일 수 있다.
- 다. 가상 기억장치의 일반적인 구현 방법에는 페이지링 기법과 세그먼테이션 기법이 있다.
- 라. 주기억장소의 물리적 공간보다 큰 프로그램은 실행 될 수 없다.

10. 기억장치 관리 기법 중 세그먼트 기법에서 사용하는 기억장소 보호 방법은?

- 가. 경계 레지스터(Boundary Register)
- 나. 기억장치 보호키(Storage Protection Key)
- 다. 기준 레지스터(Base Register)
- 라. 재배치 레지스터(Relocation Register)

11. 가상 기억장치에서 주기억장치로 페이지를 옮겨 넣을 때 주소를 조정해 주어야 하는데 이를 무엇이라 하는가?

- 가. 매핑(Mapping)                      나. 스케줄링(Scheduling)
- 다. 매칭(Matching)                    라. 로딩(Loading)

12. 세그먼테이션 기법에 대한 설명으로 옳은 것은?

- 가. 각 세그먼트의 크기는 같다.
- 나. 내부 단편화가 발생한다.
- 다. 외부 단편화가 발생한다.
- 라. 공유가 불가능하다.

13. 시간적 구역성(Temporal Locality)의 예가 아닌 것은?

- 가. 루프                                      나. 서브루틴
- 다. 프로그램의 순차적 수행            라. 스택

14. 스레싱(THRASHING) 현상의 해결 조치로 틀린 것은?

- 가. 부족한 자원을 증설한다.
- 나. 일부 프로세스를 중단시킨다.
- 다. 성능자료의 지속적 관리 및 분석으로 임계치를 예상하여 운영한다.
- 라. 다중프로그래밍의 정도를 높여준다.

[정답] 9.라 10.나 11.가 12.다 13.다 14.라



## [OS 6강]-기억장치 관리기법(주, 가상)

15. 구역성(Locality)에 대한 설명으로 옳지 않은 것은? 17. 페이지(Page) 크기에 대한 설명으로 옳은 것은?

- 가. 프로세스가 실행되는 동안 일부 페이지만 집중적으로 참조되는 경향을 말한다.
- 나. 시간 구역성은 최근에 참조된 기억 장소가 가까운 장래에도 계속 참조될 가능성이 높음을 의미한다.
- 다. 공간 구역성은 하나의 기억 장소가 참조되면 그 근처의 기억 장소가 계속 참조되는 경향이 있음을 의미한다.
- 라. 프로세스가 효율적으로 실행되기 위해 프로세스에 의해 자주 참조되는 페이지들의 집합을 말한다.

16. Denning이 제안한 프로그램의 움직임에 관한 모델로 프로세스를 효과적으로 실행하기 위하여 주기억장치에 유지되어야 하는 페이지들의 집합을 의미하는 것은?

- 가. Locality                              나. Working set
- 다. Overlay                                라. Mapping

- 가. 페이지크기가 작을 경우, 동일한 크기의 프로그램에 더 많은 수의 페이지가 필요하게 되어 주소 변환에 필요한 페이지 사상표의 공간은 더 작게 요구 된다.
- 나. 페이지 크기가 작을 경우, 페이지 단편화를 감소시키고 특정한 참조 지역성만을 포함하기 때문에 기억장치 효율은 좋을 수 있다.
- 다. 페이지 크기가 클 경우 페이지 단편화로 인해 많은 기억 공간을 낭비하고 페이지 사상표의 크기도 늘어난다.
- 라. 페이지 크기가 클 경우, 디스크와 기억장치 간에 대량의 바이트 단위로 페이지가 이동하기 때문에 디스크 접근시간 부담이 증가되어 페이지 이동 호출이 나빠진다.

18. 하나의 프로세스가 작업 수행 과정에서 수행하는 기억 장치 접근에서 지나치게 페이지 폴트가 발생하여 프로세스 수행에 소요되는 시간보다 페이지 이동에 소요되는 시간이 더 커지는 현상은?

- 가. 스레싱(Thrashing)                      나. 워킹세트(Working set)
- 다. 세마포어(Semaphore)                라. 교환(Swapping)

[정답] 15.라 16.나 17.나 18.가

