**메모리 관리 기법**

- 각 프로세스는 독립된 메모리 공간을 가지나, 운영체제 혹은 다른 프로세스의 메모리 공간에 접근할 수 없다

- 단, 운영체제는 모든 메모리 공간의 접근에 제약을 받지 않는다.

**◼ 연속 메모리 관리**

- 프로그램 전체가 하나의 공간에 연속적으로 할당되어야 함

|  |  |
| --- | --- |
| 고정 분할 기법 | 주기억장치가 고정된 파티션으로 분할  → 내부 단편화 발생 |
| 동적 분할 기법 | 파티션들이 동적으로 생성되며 자신의 크기와 같은 파티션에 적재  → 외부 단편화 발생 |

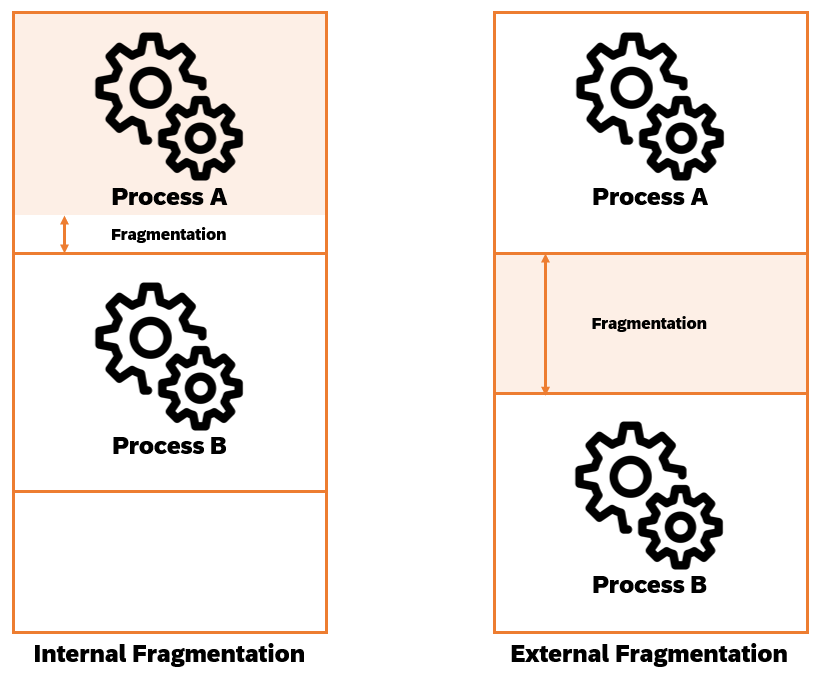
**◼ 불연속 메모리 관리**

- 프로그램의 일부가 서로 다른 주소 공간에 할당될 수 있음

|  |  |
| --- | --- |
| 페이지(Page) | 고정된 크기의 작은 프로세스 조각 |
| 프레임(Frame) | 페이지 크기와 같은 주기억장치 메모리 조각 |
| 세그먼트(Segment) | 서로 다른 크기를 가진 논리적 블록이 연속된 공간에 배치 |

**단편화(Fregmentation)**

- 프로세스가 사용하는 메모리 간에 사용하지 못하는 자유 공간들



**◼ 내부 단편화**

- 프로세스가 사용하는 메모리 공간 중 사용하지 않는 메모리 영역

**◼ 외부 단편화**

- 프로세스들 사이에 끼어 사용하지 못하는 메모리 영역

**스와핑(Swapping)**

- CPU 사용 시간이 끝난 프로세스의 메모리를 보조 기억장치[[1]](#footnote-1)로 내보내고 다른 프로세스의 메모리를 불러들이는 메모리 관리 기법

**◼ Swap-in**

- 보조 기억장치에서 주기억장치로 불러오는 과정

**◼ Swap-out**

- 주기억장치에서 보조 기억장치로 내보내는 과정

\* Swapping 시에는 디스크 전송 시간이 많이 소요되므로 메모리 공간이 부족할 때에 Swapping을 수행한다

1. 하드디스크 등 [↑](#footnote-ref-1)