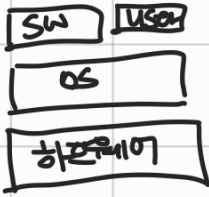


* 운영체제



⇒ 사용자 및 다른 모든 소프트웨어와 하드웨어를 연결하는 소프트웨어 계층

* 목적

① 컴퓨터 시스템을 편리하게 사용할 수 있는 환경을 제공

- OS는 동시 사용자/프로그램들이 독자적으로 컴퓨터에서 수행하는 환경을 제공

② 자원의 효율적인 관리

- CPU, I/O 장치 등의 효율적으로 관리

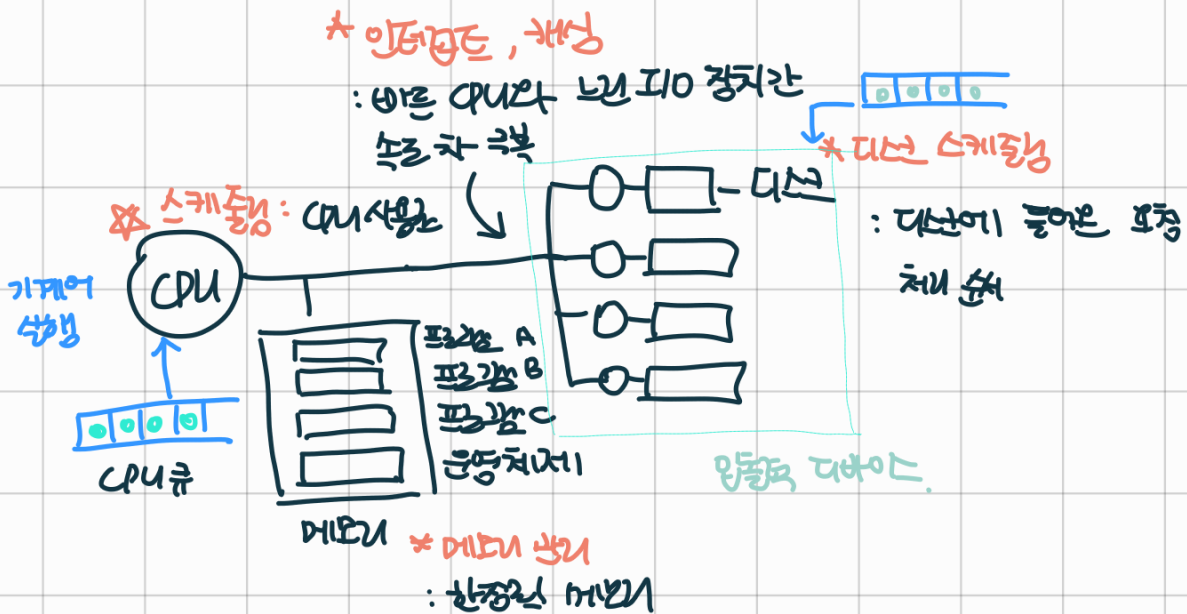
· 주어진 자원을 최대한의 성능으로 활용

· 특정 사용자/프로그램의 자원 불이익이 발생하지 않도록

⇒ 형평성

- 사용자 및 OS자신의 안전

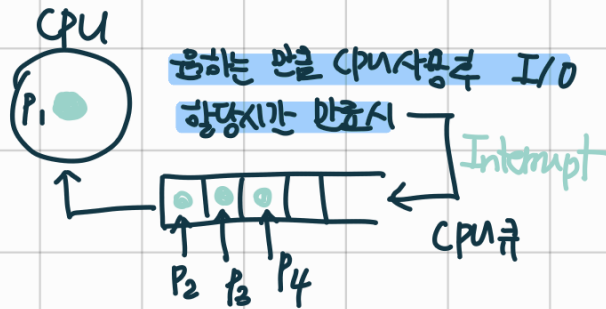
* 구조



CPU 스케줄링

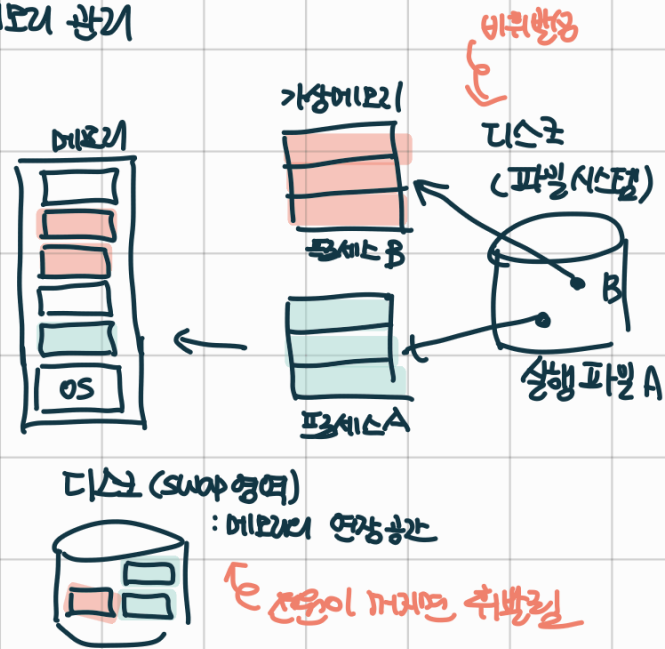
- FCFS (First come First Served)
- SJF (Shortest Job First)
 - : CPU 사용시간이 제일 짧은 프로세스 먼저 실행
 - ⇒ minimum average waiting time 보장
- 문제점: Starvation 발생
 - ↳ 왜 발생하나?
 - 계속해서 본인보다 짧은 CPU를 차지하는 경우 영원히 못받음

- RR (Round Robin)
 - : 각 프로세스는 동일한 CPU 할당 시간을 가진다



- 여러 프로세스가 CPU 큐에 있는 경우
 - 어떤 프로세스도 $(n-1) \times$ 할당시간 만큼 기다려야 함
 - 대기 시간이 프로세스의 CPU 사용시간에 비례함

메모리 관리



- 메모리가 꽉 찼을 경우?
 - ⇒ 메모리에 사용할 자원을 찾아야 함
 - ↳ 과거를 봐야 함

LRU VS LFU

- LRU : 가장 오래 전에 참조 페이지 삭제
 - ↳ 문제점: 자주 사용 X, 최근에 이용한 경우
- LFU : 참조 횟수가 가장 적은 페이지 삭제
 - ⇒ 과거에 가장 적게 사용한 자원
 - ↳ 문제점: 시간에 따라 필요한 자원의 우선순위가 달라짐

디스크 스케줄링 (HDD: 하드 디스크)

· 디스크 접근 시간 (Access time)의 구성

- 탐색시간 (Seek time)

: 헤드를 해당 트랙의 움직여야 하는 시간

- 회전시간 (Rotational time)

: 헤드가 원하는 섹터에 도착하기 까지 걸리는 시간

- 전송시간 (Transfer time)

: 실제 데이터 전송시간

· 디스크 스케줄링

- seek time 을 최소화 하는 것이 목표

- seek time \approx seek distance

· FCFS \rightarrow 사용 X

· SSTF (Shortest Seek time first)

: seek time 이 가장 짧은 것 부터 실행

\hookrightarrow 문제: starvation. 발생

· SCAN \rightarrow 가장 많이 사용

: 헤드가 디스크의 한쪽 끝에서 다른쪽 끝으로 이동해다

가는 모든 요청 처리 ex)



\Rightarrow 헤드도 이동거리가 짧아질

저장 장치 계층구조나 캐싱 (caching)

