
Operating System

컴퓨터개론

(Introduction to Computer Systems)

GEN1030

운영체제 개요

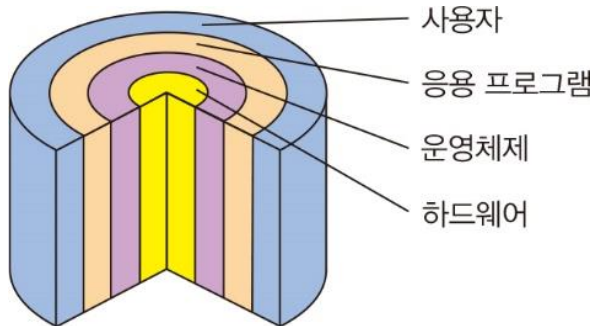
운영체제란?

- Operating System (OS)

- 주기억장치 내에 상주하면서 컴퓨터의 효율적인 운영을 담당하는 시스템 소프트웨어

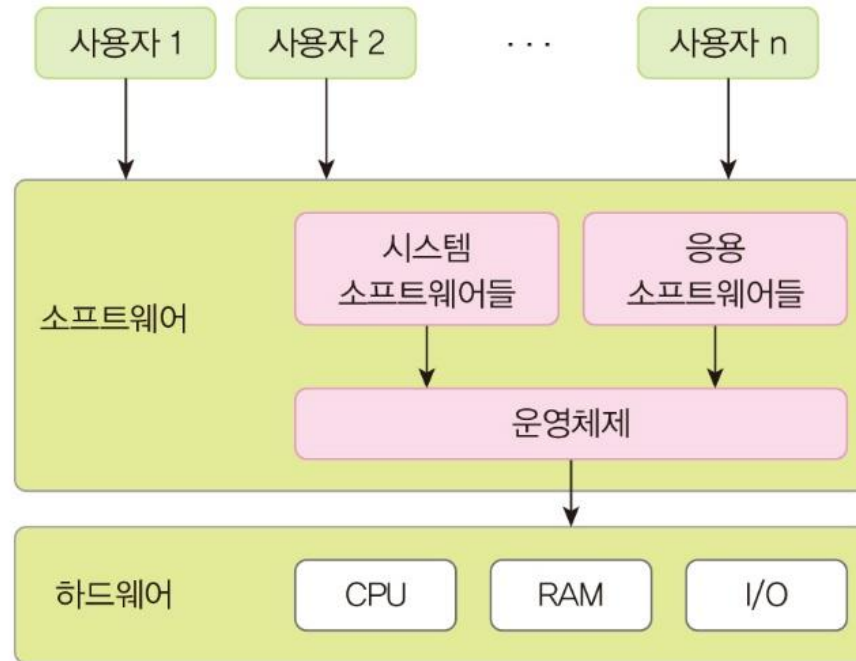
- 시스템의 자원(resource)인 CPU, 주기억장치, 보조기억장치, 입출력 장치, 네트워크 등을 효율적으로 관리하고 운영함으로써 인간과 컴퓨터 간의 인터페이스(interface) 역할을 담당

Interface? a point of connection, boundary, or interaction between two distinct entities



운영체제란?

- 컴퓨터 내의 다른 프로그램들을 관리
- 하드웨어와 소프트웨어 자원 관리하고 제어



운영체제 목적과 기능

- 컴퓨터 시스템의 자원을 편리하게 사용할 수 있는 환경을 제공하고 컴퓨터 시스템의 자원들을 효율적으로 관리하여 시스템의 성능을 최적화
 - 사용자에게 편리한 인터페이스 제공
 - 하드웨어와 소프트웨어 자원 관리 및 제어
 - 수행 중인 프로그램들의 효율적인 운영을 지원
 - 작업 처리 과정 중에 데이터 공유
 - 오류가 발생 시 오류 처리

초기화	자원 관리	기타
<ul style="list-style-type: none"> • 컴퓨터 시스템 초기화 설정 기능 	<ul style="list-style-type: none"> • 중앙처리장치 • 저장장치 • 입출력장치 • 주기억장치 • 네트워크장치 • 파일시스템 	<ul style="list-style-type: none"> • 사용자와 컴퓨터 간의 편리한 인터페이스 기능 • 오류 검사 및 복구 기능 • 사용자 계정관리 • 자원 공유 및 보안

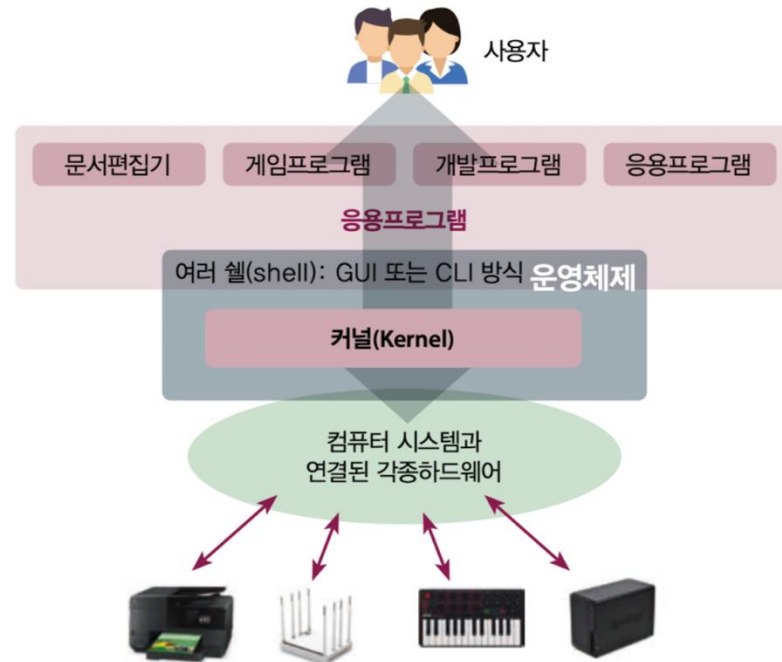
운영체제 구성

- 커널(Kernel)

- 자원을 관리하는 모듈의 집합으로 운영체제 기능의 핵심적인 부분을 모아 놓은 부분
- 메모리 관리 및 스케줄링 처리 등의 기능을 담당함
- 사용자는 직접 커널의 기능을 제어할 수 없으며 셸(shell)에 의뢰함
- 꼭 필요한 부분이므로 항상 메모리에 적재되어 있음

- 셸(Shell)

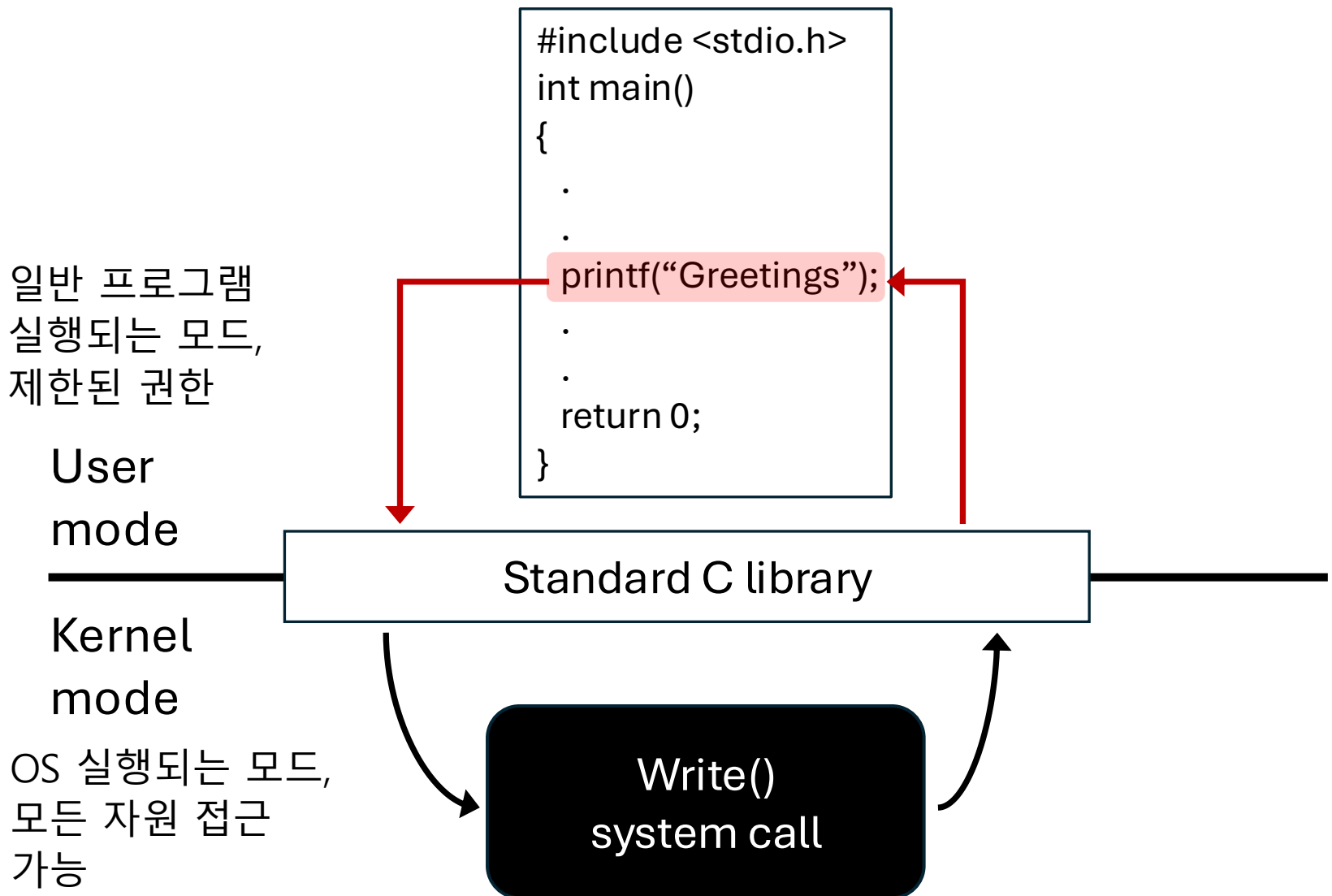
- 사용자 명령을 받아 해석하고 커널에 전달하는 프로그램
- CLI or GUI



System Call

- 응용 프로그램이 OS가 관리하는 하드웨어 자원에 접근하거나 서비스를 이용하기 위해 OS에 요청을 보내는 유일한 통로
 - 응용 프로그램이 직접 하드웨어 제어하면 시스템 전체가 위험해질 수 있음
- 작동 원리
 - 응용 프로그램이 특정 기능을 위해 System call 발생
 - Ex) 파일 읽기/쓰기, 프로세스 생성/종료, 네트워크 통신
 - CPU가 User mode에서 Kernel mode로 전환
 - User mode: 일반 프로그램 실행되는 모드, 제한된 권한
 - Kernel mode: OS 실행되는 모드, 모든 자원 접근 가능
 - OS가 요청된 작업 수행 후 다시 User mode로 복귀하여 결과 전달

System Call



운영체제 종류

- 운용체제 종류

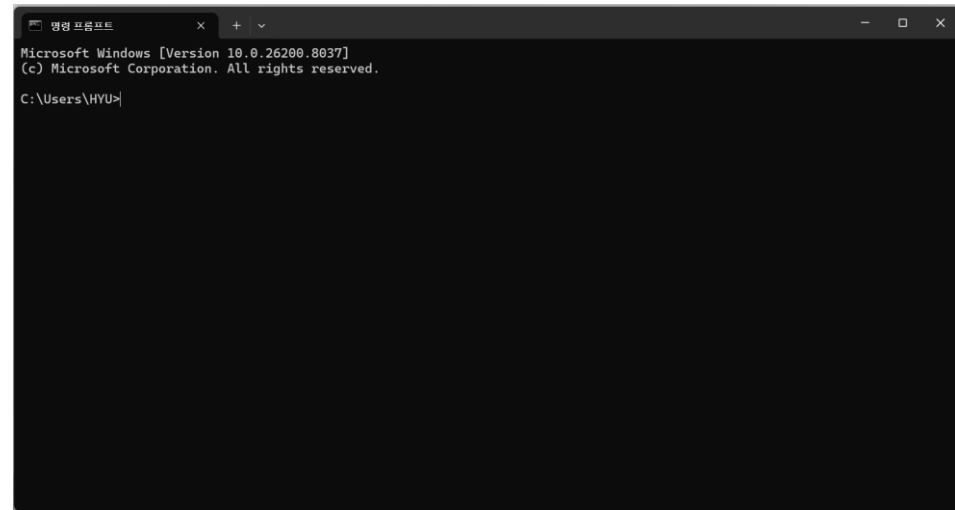
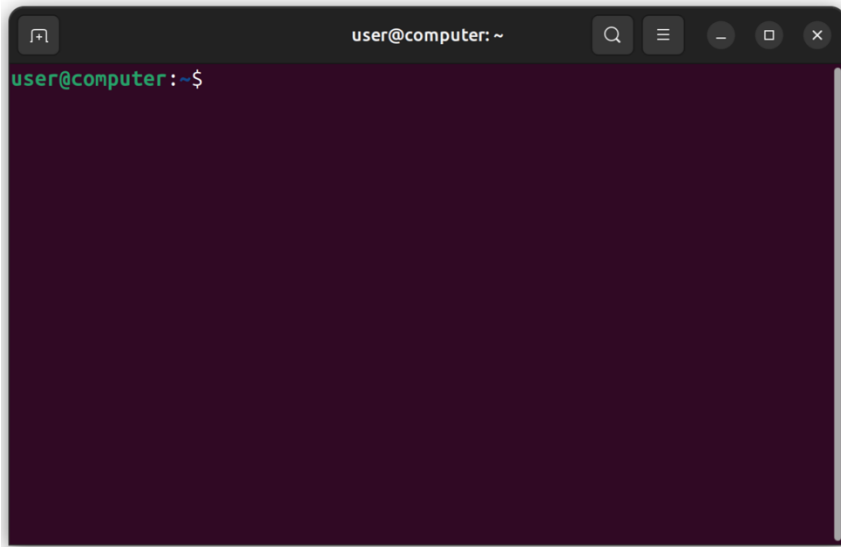
개인용 컴퓨터	중대형 컴퓨터	스마트폰 및 기기
<ul style="list-style-type: none">• MS-DOS• Windows 계열• OS/2• Linux 계열• 맥OS	<ul style="list-style-type: none">• Unix 계열• Windows Server 계열• VMS	<ul style="list-style-type: none">• iOS• Android• Windows

사용자 인터페이스 방식

- 사용자가 OS를 조작하는 방식
 - 시스템과 사용자 사이의 상호작용이 일어나는 접점
- 명령행 인터페이스 방식
 - Command Line Interface (CLI)
- 그래픽 사용자 인터페이스 방식
 - Graphic User Interface (GUI)

사용자 인터페이스 방식

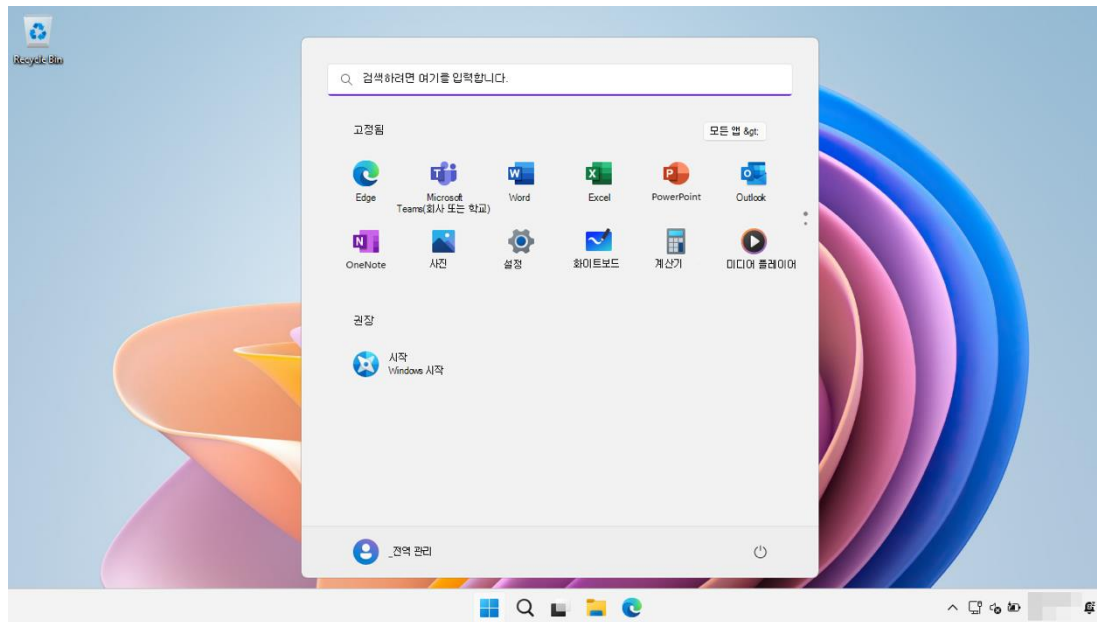
- 명령행 인터페이스 방식
 - Command Line Interface (CLI)
 - 운영체제가 제공하는 기능을 키보드 입력을 통해 사용하는 방식



- 그래픽 처리가 필요 없어 시스템 자원 사용이 낮은편
- MS-DOS, Linux 터미널(Terminal), Windows의 명령 프롬프트(Command Prompt)

사용자 인터페이스 방식

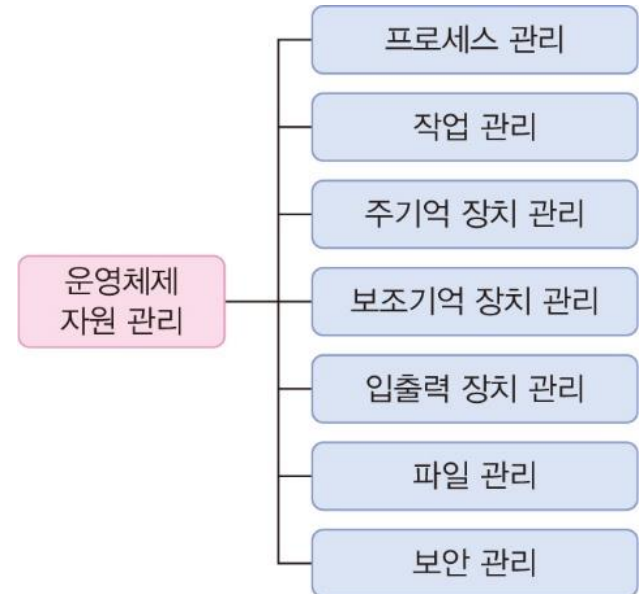
- 그래픽 사용자 인터페이스 방식
 - Graphic User Interface (GUI)
 - 운영체제가 제공하는 기능을 아이콘/메뉴로 보여주고 사용자가 마우스로 선택해서 작업 수행하는 방식
 - CLI 보다 직관적
 - CLI 보다 리소스 소모가 큼



운영체제 역할 - 관리

운영체제의 컴퓨터 관리

- 운영체제의 가장 중요한 기능은 시스템의 여러 자원을 관리하는 것
- **프로세스(process) 관리**: 프로세스의 생성과 삭제, 중지와 계속 등을 관리
- **작업(job) 관리**: 작업과 관련된 순서, 우선순위, 프로세스 할당 등 관리
- **주기억장치 관리**: 주기억장치의 할당과 회수를 담당함
- **보조기억장치 관리**: 보조기억 장치의 효율적인 사용을 관리
- **입출력 장치 관리**: 입출력 장치들을 관리함
- **파일 관리**: 파일의 관리를 담당함
- **보안 관리**: 보안과 관련 된 사항들을 관리함



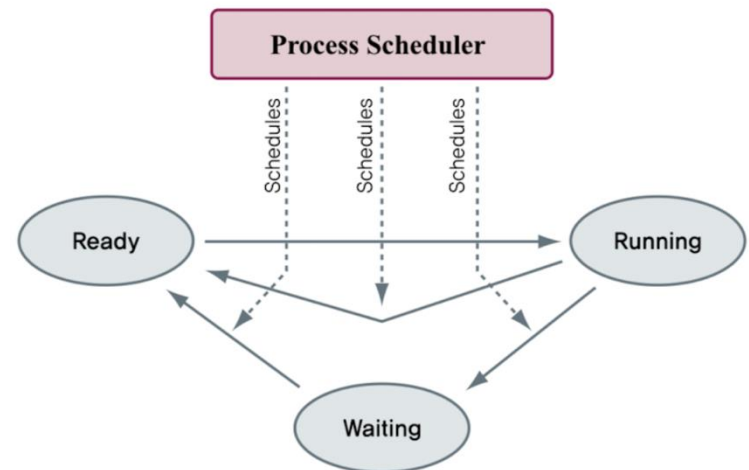
프로세스 관리

- 프로세스(Process)
 - 컴퓨터 내부에서 현재 실행 중이거나 곧 실행이 가능한 프로그램
 - 시작했지만 아직 종료되지 않은 프로그램으로 주기억장치에 적재되어 있는 프로그램
- 프로세스 관리자(Process Manager)
 - 프로세스의 생성과 삭제, 중지와 계속, 동기화 등의 기능 수행

프로세스 관리

- 프로세스 상태
 - 준비(Ready)
 - 필요한 자원(주기억장치 등)을 할당 받은 상태에서 프로세서(CPU)를 할당 받기 위해 기다리는 상태
 - 실행(Running)
 - 프로세서에 의해 실행중인 상태
 - 운영체제에 의해 다시 준비, 대기 상태로 갈 수 있음
 - 대기(Waiting)
 - 프로세서가 필요한 자원 요청한 후 할당 받을 때까지 기다리는 상태

- 프로세스 스케줄러(Scheduler)
 - 프로세스의 상태를 다른 상태로 이동시키는 모듈
 - 프로세스 관리자의 주요 구성요소



Program vs. Process

Program

- 보조기억장치에 저장된 실행 파일 (Static)
- 아직 실행되지 않은 상태
- Ex) chrome.exe

Process

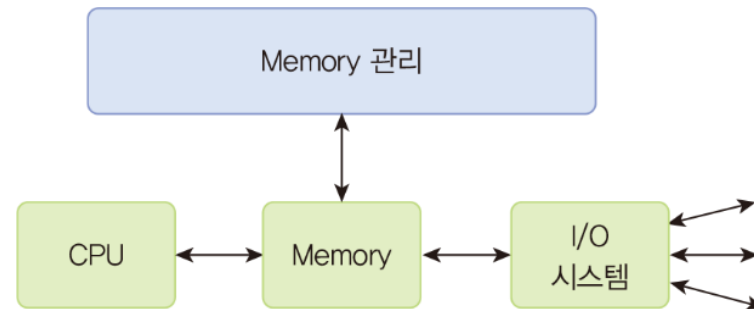
- 실행 중인 프로그램 (Dynamic)
- CPU, Memory 등 자원 할당 받음
- 하나의 program → 여러 process 가능

기억장치 관리

- OS는 기억장치를 효율적으로 사용할 수 있도록 관리

- 관리 대상

- 주기억장치
- 보조기억장치

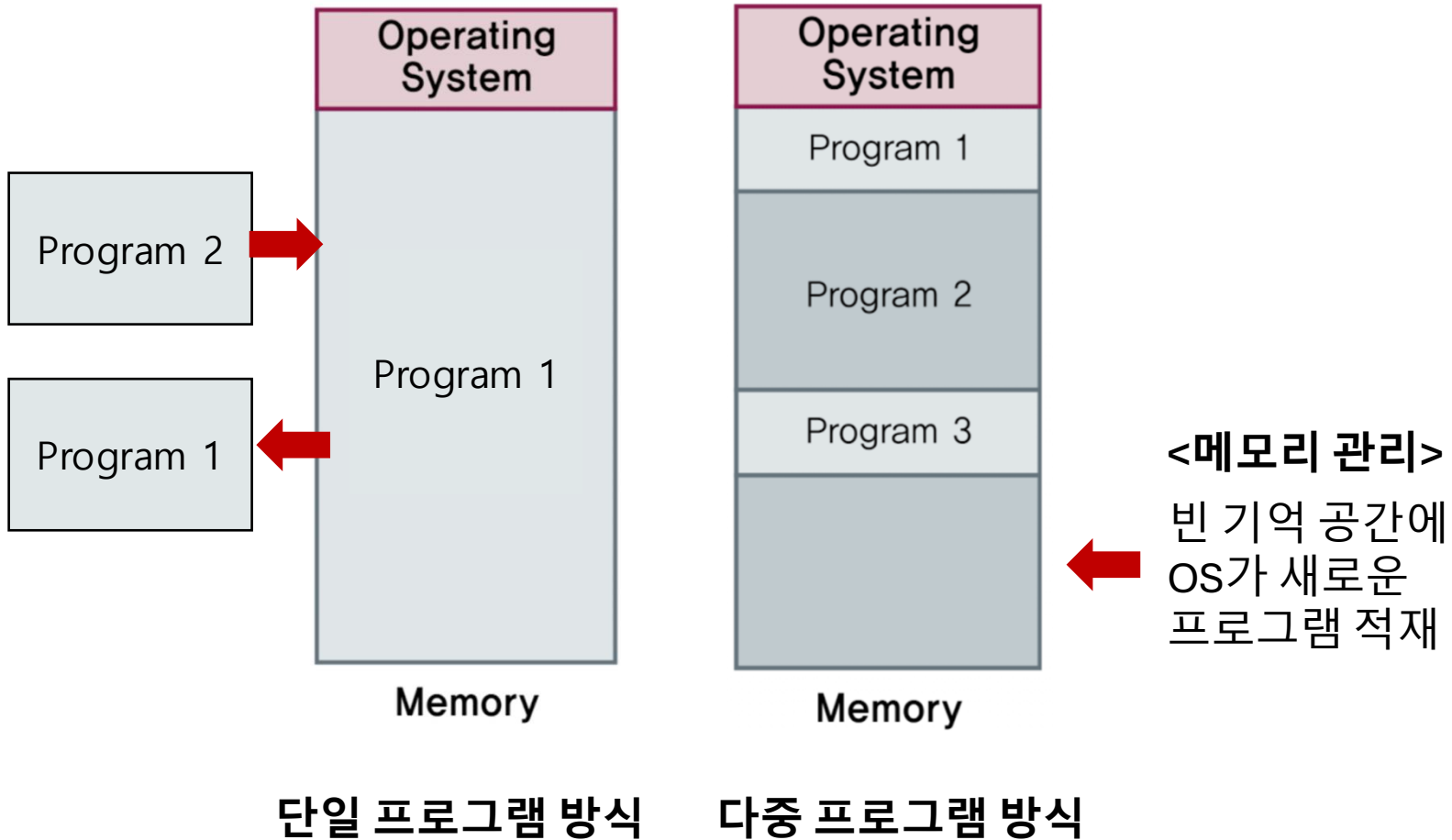


- 주기억장치와 보조기억장치 사이의 정보 교환 관리
- 어느 프로세스를 주기억장치에 적재할지 결정해 적절한 주기억장치에 프로세스를 적재
- 프로세스가 더 이상 주기억장치를 필요로 하지 않으면 다시 주기억장치를 회수

기억장치 관리

- 단일 프로그램 방식(Single-programming)
 - 하나의 프로그램만 메모리에 적재 및 실행
 - 다른 프로그램 실행 시 이전의 프로그램 내리고 새로운 프로그램 적재
 - CPU가 쉬는 시간이 많을 수 있음
 - ex) 입출력 요청을 기다리는 동안 - 파일 읽기, 키보드 입력 대기, 네트워크 응답 대기 등
- 다중 프로그램 방식(Multiprogramming)
 - 메모리에 여러 개의 프로그램 동시에 적재
 - 한 프로그램이 기다리는 동안 다른 프로그램 실행
→ CPU 활용률 증가

기억장치 관리

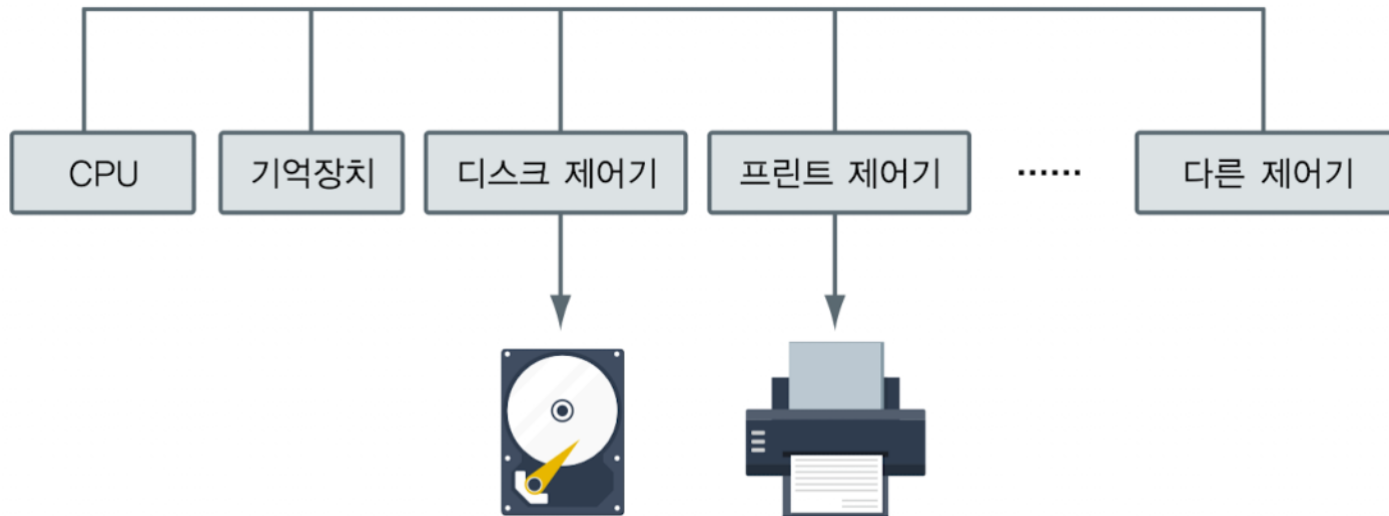


장치 관리

- 장치 관리자(Device Manager)
 - 입출력 장치의 할당과 해제 등의 기능을 관리
- 입출력 처리는 인터럽트를 이용하여 수행
 - **인터럽트(interrupt)**: 컴퓨터 장치에서 어떠한 이벤트(e.g., I/O 완료)가 발생했음을 CPU에게 알리는 신호
 - 실행 중인 프로그램 흐름과 무관하게 발생 가능
- 실행 중인 프로세스가 입출력 요구
 - 프로세스 실행 잠시 중단(waiting)
 - CPU는 다른 프로세스 실행 가능
 - 해당하는 장치를 할당 및 입출력 수행
 - 입출력 완료 및 인터럽트 발생
 - 입출력 장치 자원 회수
 - 중단된 시점부터 프로세스 실행 재개

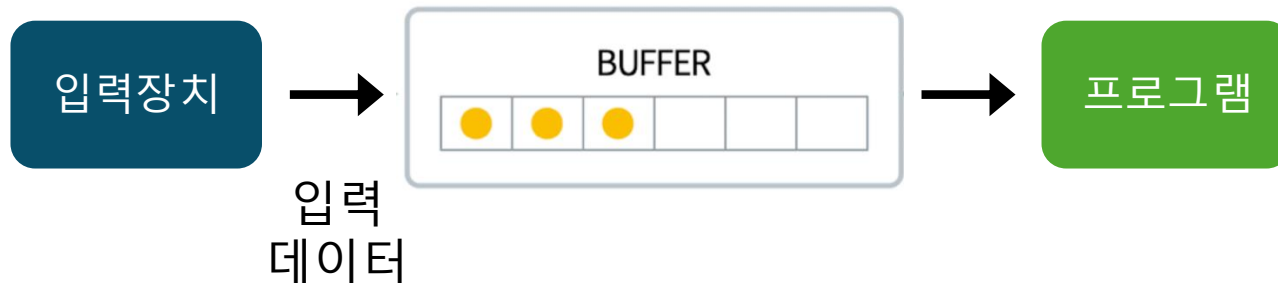
장치 관리

- 장치 제어기(Device Driver)
 - 장치 관리자와 입출력장치 사이의 인터페이스 담당
 - 장치 관리자가 모든 기기의 내부 구조를 알 필요 없이, 표준화된 명령을 내릴 수 있게 함 (추상화, abstraction)
 - 장치마다 다른 드라이버 필요



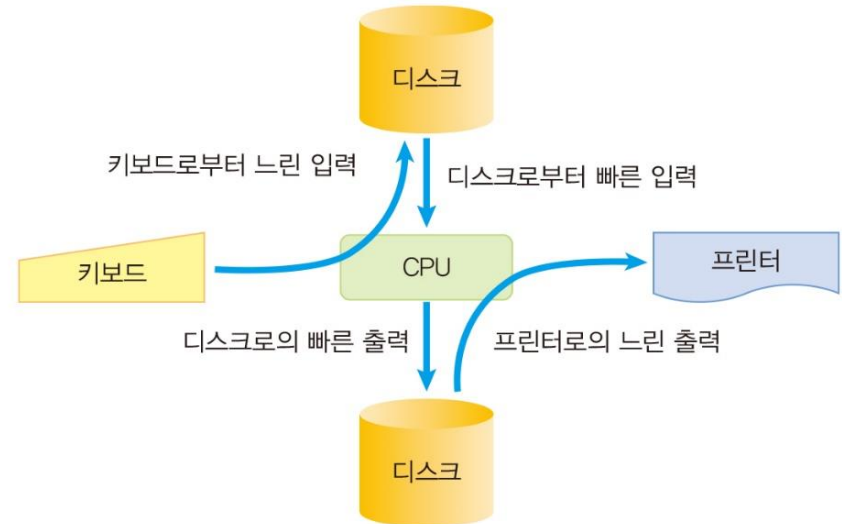
입출력 프로그래밍

- 입출력 관리에서 중요한 고려 사항
 - CPU와의 속도 차이
- 버퍼링(Buffering)
 - 입출력 장치는 기계적 요인으로 인해 중앙처리장치보다 훨씬 느린 속도로 작동
 - 주기억장치 일부를 임시 저장 공간(버퍼)로 뒤서 처리 효율을 올림



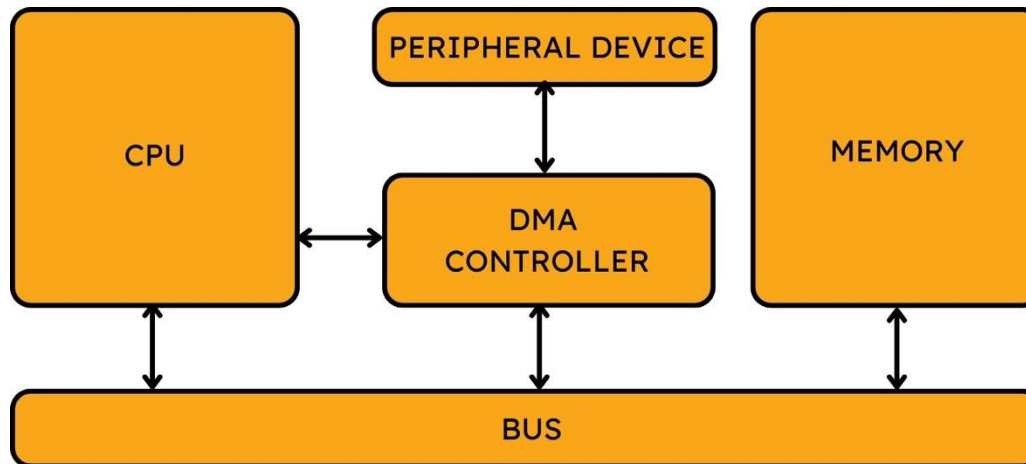
입출력 프로그래밍

- 입출력 관리에서 중요한 고려 사항
 - CPU와의 속도 차이
- 스푼링(Spooling)
 - Simultaneous Peripheral Operation On-Line (SPOOL-ing)
 - 입출력 작업을 보조기억장치에 임시로 모아두었다 처리
 - OS는 처리한 결과를 보조기억장치에 저장하고 즉시 다른 일 처리 가능 (입출력장치의 처리를 기다릴 필요 없음)
 - 입출력장치는 준비되면 모아진 작업을 순차적으로 처리



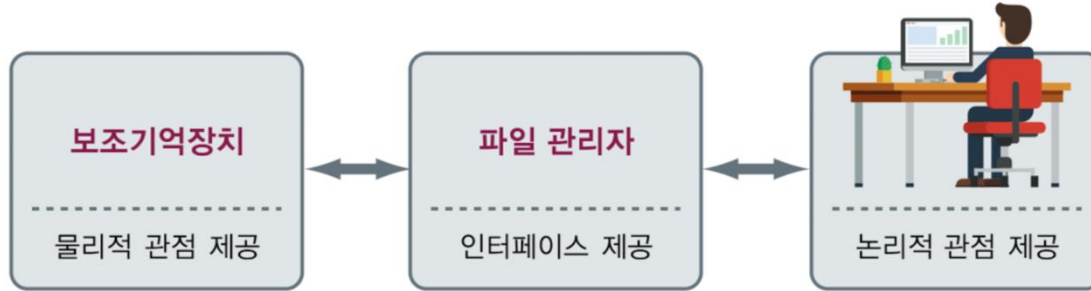
Direct Memory Access (DMA)

- CPU 개입 없이 입출력장치와 메모리 사이 직접 데이터 전송
 - CPU는 다른 작업 수행 가능
- DMA 컨트롤러(Controller)가 데이터 전송 담당
- OS가 초기 설정만 수행
 - 전송할 데이터 위치, 전송 크기, 입출력장치 정보 등



파일 관리

- 파일 관리자(File Manager)
 - 보조기억장치에 저장되는 파일 관리
 - 사용자와 보조기억장치 사이의 인터페이스 제공
 - 사용자의 논리적 관점과 하드웨어의 물리적 관점 연결



- 기능
 - 파일 접근 제어
 - 파일 생성, 삭제 수정 관리
 - 파일의 저장 위치 관리(파일 이름과 저장 위치 매핑)
 - 파일 공유 기능 제공(여러 사용자가 공유하기 위한)
 - 폴더 구조 관리

보안 기능

- 여러 프로그램이 동시에 실행 됨
 - 서로 간섭하면 시스템이 제대로 작동하지 않을 수 있음
- OS가 제공하는 보안 기능
 - Process isolation
 - 각 프로세스는 독립된 메모리 공간 사용
 - 다른 프로세스 접근 불가
 - Access control
 - 파일/자원 접근 권한 관리
 - Authentication
 - 사용자 계정 기반 접근 제어

Summary

- Kernel
- Shell
- CLI and GUI
- Management
 - Process
 - Memory
 - Device
 - File
 - ...