제 1 장 운영체제의 개요

0 운영체제(kernel:커널)란 무엇인가 ?

- 컴퓨터 시스템의 핵심적인 소프트웨어
- 컴퓨터 하드웨어를 관리하는 시스템 소프트웨어
- 하드웨어와 응용 프로그램 간의 상호 작용을 관리하고 제어하는 시스템 소프트웨어
- 컴퓨터의 자원을 효율적으로 관리, 활용

0 주요 운영 체제로는 윈도우(Windows), 리눅스, UNIX, macOS등

(정리) 운영체제란 무엇인가 ?

- 자원의 관리 <- 운영체제의 역할

★ 자원(resource) 이란 무엇인가 ?

0 운영체제 시작 원리

- 전원 스위치 on할 경우 ROM내의 F/W(Firmware)인 boot strap loader)가 작동
- boot strap loader 역할 ?

♦ 운영체제의 또 다른 기능

- 복잡성을 숨기고 프로그램머에게 친숙한(user friendly) 사용 환경을 제공

♦ 운영체제의 수행 모드

- 커널 영역(kernel area)
- 사용자 영역(user area)

♦ 운영체제의 의미

- 좁은 의미의 운영체제 --> 커널(kernel) : 전문가들이 사용하는 용어
- 넓은 의미의 운영체제 --> 커널 + 라이버러리+명령어등 : 일반인들이 사용하는 용어

♦ 넓은 의미의 운영체제 구조(그림 1.1) : pp.14

0 운영 체제(커널)의 4가지 기능 모듈

0 운영 체제(커널)의 4가지 기능 모듈

(1) 프로세스 관리(process management) : 프로세스는 실행 중인 프로그램을 의미하며, 운영 체제는 프로세스의 생성, 스케줄링, 종료 등을 관리, 관련 자원 : CPU

(2) 메모리 관리(memory management) : 운영 체제는 시스템의 메모리(주기억장치, Main Memory:M.M)를 효율적으로 할당하고 관리하여 여러 프로세스가 메모리를 공유하고 충돌하지 않도록 한다. 관련 자원 : 주기억장치(main memory)

(3) 파일 시스템 관리(file system management) : 파일 시스템은 데이터를 저장하고 관리하는 방식을 의미하며, 운영 체제는 파일과 디렉토리를 생성, 삭제, 읽기, 쓰기 등의 작업을 관리, 관련 자원 : 보조기억장치

(4) 입출력 관리(IO management) : 운영 체제는 입력과 출력 장치를 관리하며, 데이터의 흐름을 조절하여 데이터를 읽고 쓰는 작업을 관리, 관련 자원 : 입출력장치

1.1 운영체제의 의미

1.1.1 확장된 기계

♦ 운영체제 사용자에게 추상화된 형태(data abstraction)(동작을 볼수 없도록 설계)로 제공
♦ 사용자에게 추상화된 기계 형태로 제공 --> 가상 기계(Virtual Machine) 개념

1.1.2 자원 관리자

♦ 자원을 효율적이고 공정하게 관리(Manage resources efficiently and fairly)

1.2 운영체제의 역사

♦ 운영체제는 HW와 밀접하게 연관
♦ 운영체제의 발달 역사는 HW 발달 역사와 밀접히 연관

(pp. 21-22)

	특 징
1945-1950 년	■ 1945-1952 년 ■ 운영체제의 **모든 일을 사람들이 직접 담당**:한번에 하나의 작업만 수행 ■ 운영체제라는 개념이 없음
1950-1960년	■ 1952 년- : 최초의 운영체제가 등장 (IBM701,IBM709)
1960년대	■ OS : 하드웨어(자원)를 효율적으로 이용하기 위해 개발 -> multi-tasking/multi-programming 개념이 등장 ■ **시분할 체제(time-sharing system)** 개념의 등장
1970년대	■ 1970 년대 전반기 Unix OS 보급 확산 -> 강력한 파일 시스템 제공, multi-user기능으로 보급 확산 ■ 최초 PC의 등장, 구조적 프로그램 기술이 개발된 시기
1980년대	■ 수많은 컴퓨터(초대형, 대형, 중형, 소형, 워크스테이션, PC) 등장 -> 다양한 OS 등장 ■ 윈도우 및 GUI 기술 등장 : 사용자 편리성 제공
1990 년 이후 - 2010	■ WWW을 통한 '사이버 스페이스' 개념 등장 ■ 멀티미디어 PC 등장 ■ Unix의 기능 확장 및 다종화
2010년대	■ cloud 서비스의 확산으로 가상환경에서 운영체제 사용이 증가 ■ 다양한 분야에 전자기기 사용의 증대 ➔ 이러한 기기에 탑재되는 운영체제 종류들이 다양화 ➔
2020년대2020년대~2025년	■ OS에 AI기능을 통합 ■ 기계 장치 등이 전자화 되는 추세

	➔ 모든 전자기기에는 운영체제 탑재가 필수, 실시간 운영체제(real time OS)탑재 ■ 양자 컴퓨팅(quantum computing) 기술의 발달 - 양자 컴퓨터에 맞는 운영체제 기술 개발

(정의) **일괄처리(batch processing)**
작업을 한꺼번에 모아서 처리하는 것
(장점)
(단점)

(정의) **다중 프로그래밍(Multiprogramming)**
여러 개의 프로그램이 동시에 주기억장치에 탑재

♦ Multi-programming 기법을 사용(pp. 27)

- 많은 사용자 프로그램이 동시에 주 기억 장치에 상주
- 대부분의 프로그램의 수행은 입출력 수행 시간이 전체 프로그램 수행의 80%이상을 차지

♦ 다중 프로그래밍 동작 구조(그림 1.4, pp.27)
---> 주기억장치에 n개의 프로그램이 동시에 상주

★ multimedia 컴퓨팅의 등장
- (특징) 실시간성(Deadline)을 만족
- **실시간 운영체제(Realtime O.S, RTOS)**의 사용을 통한 Deadline 만족

- (추가) **실시간 처리 방법**
(1) Soft realtime

(2) Hard realtime
(장점)
(단점)

1.3 운영체제의 구조

0 운영체제 구조는 단일체 구조, 계층시스템 구조로 분류
(1) Monolithic structure(단일체 구조)
(2) 계층 시스템 구조

0 커널의 모듈화(kernel module, 커널의 네가지 구성 요소)
- 계층적인 구조로 동작
- Pp. 36 그림

1.3.1 클라이언트/서버 모델(분산 처리 시스템)

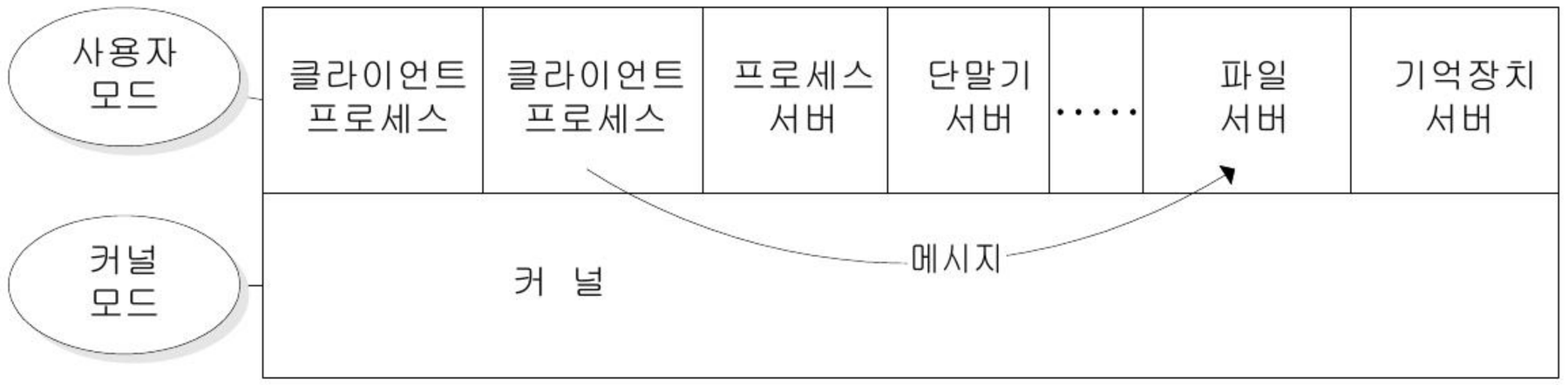
(정의) 분산 처리 시스템(distributed processing system)
여러 개의 시스템을 네트워크로 묶어서 상호 동작이 되도록 하는 시스템
예) 클라우드 시스템, 금융시스템, 웹서버를 이용한 데이터 접근등
(장점)
(단점)

0 **분산처리 시스템의 두가지 일반적인 모델**
(1) 메시지 전송(message passing) 모델
- 커널 프로세스 간에 통신 기능을 통해서 메시지 교환
(장점)
(단점)
(2) 공유 기억 장치 모델 (Shared Memory Model)
- 분산 시스템간에 공유 기억장치(shared memory system)를 이용하여 정보 교환
(장점) 빠른 수행 속도를 보장
(단점) 구현이 어렵다.

보호와 동기화 부분의 관리 어렵다.

0 커널내 프로세스 간의 클라이언트/서버 모델

- 그림 1.5



0 분산 시스템의 클라이언트/서버 모델

- 그림 1.6

(추가)

0 최근 운영체제는 커널내의 기능을 사용자 영역으로 이동시키는 추세

0 커널의 기능을 단순화시키는 추세

=====> 마이크로 커널(micro kernel)

(장점)

(단점)

0 대표적인 마이크로 커널

--> MACH O.S, QNX O.S, MINIX O.S

1.4 운영체제 관련 용어 정의

1.4.1 프로세스(process)

0 프로세스의 정의 : 실행 상태의 프로그램을 말함

0 프로그램은 그 자체가 프로세스는 아님.

0 커널은 프로세스 관리 기능을 담당

- 프로세스들의 생성 및 제거

0 커널은 프로세스들에게 CPU시간을 할당

1.4.2 재 진입(Reentrance)

0 많은 프로세스들이 동시에 커널 코드에 진입을 하여 실행할 수 있도록 커널을 설계
0 이때 커널 내의 코드가 사용 중에 내용이 변하지 않아야
한다.

1.4.3 시스템 호출(system call)
0 사용자 프로그램은 커널과 통신하면서 시스템
호출을 통하여 필요한 서비스를 커널에 요구