

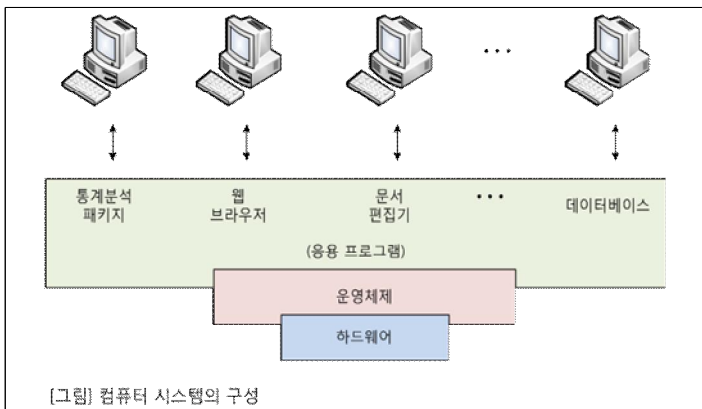
1주차 1차시 운영체제의 개념과 기능

【학습목표】

1. 운영체제의 개념 및 기능을 설명할 수 있다.

학습내용1 : 컴퓨터 시스템의 개념

1. 컴퓨터 시스템



2. 하드웨어

물리적인 컴퓨터 자원

예시 : 중앙처리장치(CPU), 메모리, 입출력장치

3. 운영체제

응용 프로그램의 하드웨어 사용을 제어 및 조정
-사용자에게 컴퓨터 사용을 위한 인터페이스 제공

예시 : 유닉스/리눅스, 윈도우 XP, Mac OS X

4. 응용 프로그램

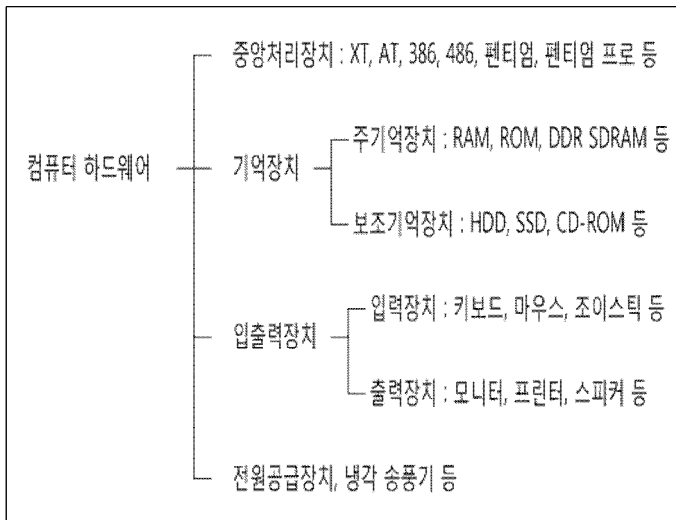
사용자들의 문제를 해결하기 위한 소프트웨어

예시 : 엑셀, 워드프로세서, 웹 브라우저, 데이터베이스

5. 사용자

사람, 기계, 다른 컴퓨터 시스템

6. 컴퓨터 하드웨어 구성 요소



7. 컴퓨터 시스템의 기능

① 프로그램 실행

- CPU가 주기억장치로부터 프로그램 코드를 읽어서 실행

② 데이터 저장

- 프로그램 실행 결과로 얻어진 데이터를 주기억장치에 저장

③ 데이터 이동

- 보조기억장치에 저장되어 있는 프로그램과 데이터 블록을 주기억장치로 이동

④ 데이터 입력/출력

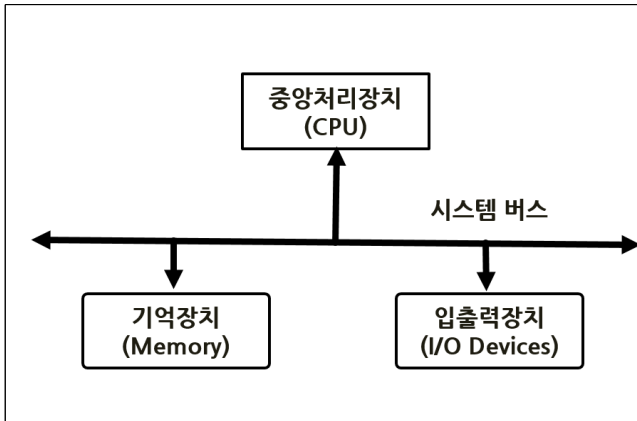
- 사용자의 키보드를 통한 명령 또는 데이터 입력
- CPU가 처리한 결과값이나 기억장치의 내용을 출력 장치에 출력

⑤ 제어

- 프로그램이 스케줄에 맞춰 실행되도록 제어
- 필요에 따라 프로그램의 실행 순서를 변경하도록 조정

8. 컴퓨터 시스템의 기본 구조

- [그림] 컴퓨터 시스템의 기본 구조



9. 중앙처리장치

- 프로세서(Processor)라고도 함
- 프로그램 실행과 데이터 처리라는 중추적 기능 수행을 담당

10. 주기억장치

CPU 가까이 위치하며, 반도체 기억장치 칩들로 구성
보조 기억장치에 비해 고속 액세스 가능
저장 용량의 한계로 가격이 높고 면적을 많이 차지 함
RAM(Random Access Memory)

- 휘발성 메모리의 특성을 지님
- 영구 저장 능력이 없기 때문에 일시적인 저장장치로만 활용

11. 보조기억장치

2차 기억장치
기계적인 장치가 포함되기 때문에 저속 액세스 처리
저장 밀도가 높고 비트당 비용이 증가
영구 저장 능력을 가진 저장 장치
- HDD, SSD, CD-ROM 등

12. 입출력장치

입력장치(Input Device)
출력장치(Output Device)
각 장치와 CPU간의 데이터 이동을 위한 별도의 제어기가 존재함
사용자와 컴퓨터의 데이터를 위한 도구로 활용

13. 컴퓨터 정보

1과 0으로 구성된 2진수 비트(1과 0)들로 표현된 프로그램 코드와 데이터

14. 프로그램 코드

기계어
어셈블리어
고급 언어

① 기계어

컴퓨터 하드웨어가 이해할 수 있는 언어
2진 비트로 구성

② 어셈블리어

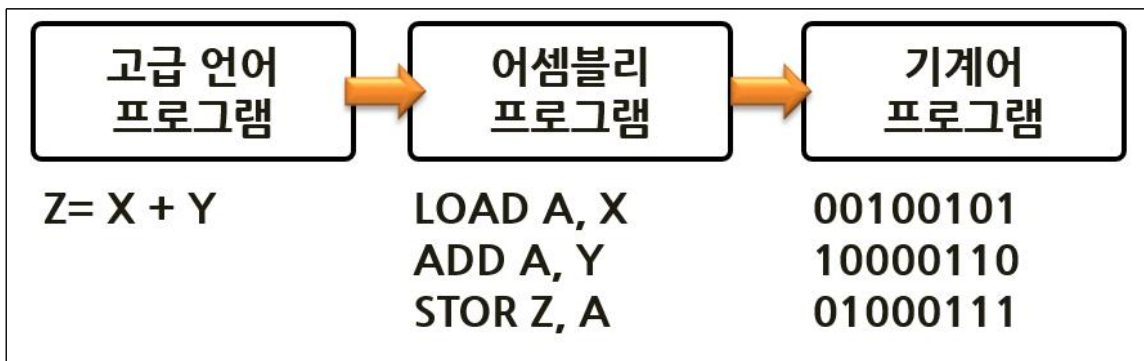
고급 언어와 기계어 사이의 중간 언어
저급 언어(Low-Level Language)로 분류
기계어 코드를 사람이 이해할 수 있는 심볼로 표현

③ 고급 언어

영문자와 숫자로 구성된 사람이 이해하기 쉬운 언어
예시 : C, C++, JAVA, Python 등

15. 프로그램 언어의 변환 과정

$Z = X + Y$



① LOAD A, X

- 기억장치 X번지의 내용을 읽어 레지스터 A에 보관

② ADD A, Y

- 기억장치 Y번지의 내용을 읽어 레지스터 A에 보관된 값과 더한 후 결과를 A에 보관

③ STOR Z, A

- 레지스터 A에 있는 연산 결과를 기억장치 Z번지에 저장

16. 프로그램 언어 번역 소프트웨어

①컴파일러

- 고급 언어 프로그램을 기계어 프로그램으로 번역하는 소프트웨어

②컴파일러

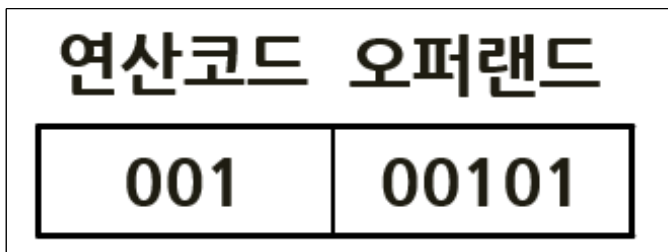
어셈블리어 프로그램을 기계어 프로그램으로 번역하는 소프트웨어

니모닉스를 사용

- 어셈블리 명령어가 지정하는 동작을 개략적으로 짐작할 수 있도록 하기 위해서 사용된 기호

예시 : LOAD, ADD, STOR 등

17. 기계 명령어 형식



①연산코드

CPU가 수행할 연산을 지정해 주는 비트들

비트수가 3이면 지정할 수 있는 연산의 최대 수는 8개($2^3=8$)

②오퍼랜드

저장되거나 연산에 사용될 데이터가 저장된 기억장치의 주소

비트수가 5라면 주소 지정할 수 있는 기억 장소의 최대 수는 32개 ($2^5=32$)

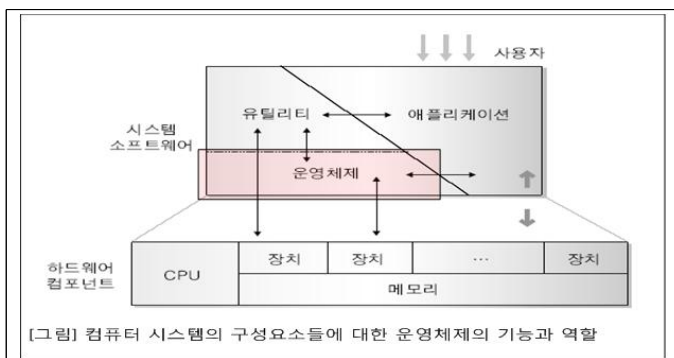
학습내용2 : 운영체제의 개념

1. 운영체제의 정의

* 일반적인 관점에서 바라본 운영체제

컴퓨터 하드웨어와 컴퓨터 사용자 간의 매개체 역할을 하는 시스템 소프트웨어

사용자가 프로그램을 수행할 수 있는 환경을 제공



- 지원관리자 측면에서 본 운영체제

컴퓨터 시스템의 데이터 이동, 저장, 처리와 이들과 관련된 필요한 자원을 제어하는 역할을 수행하는 소프트웨어

① 조정자 역할

시스템을 운영하여 이들 자원을 적절하게 사용할 수 있도록 제어하는 역할

② 자원 할당자

컴퓨터 시스템을 공정하고 효율적으로 운영하기 위한 역할

③ 입출력 장치와 사용자 프로그램 제어

프로그램의 수행을 제어하며 특히 입출력장치를 동작하고 통제하는 역할

2. 운영체제의 목적

① 사용자에게 편리한 환경 제공

② 시스템 성능 향상

처리능력의 향상

- 시간당 작업처리량 향상

신뢰도의 향상

- 실패 없이 주어진 기능을 수행할 수 있는 능력 향상

응답시간의 단축

- 사용자가 시스템에 작업을 의뢰하고 나서 반응을 얻을 때까지의 시간 단축

사용가능도의 향상

- 고장과 오류가 발생해도 영향을 받지 않거나 최소한의 영향으로 운영 가능

3. 운영체제의 발전 과정

구분	시대	특징
제0세대	1940년대	<ul style="list-style-type: none"> • 운영체제 없음 • 기계어 사용
제1세대	1950년대	<ul style="list-style-type: none"> • IBM 701(운영체제 효시) • 작업간의 원활한 변환 • 버퍼링, 스펀링, 일괄처리 시스템
제2세대	1960년대 초기	<ul style="list-style-type: none"> • 고급언어로 운영체제 제작 • 장치 독립성 • 다중 프로그래밍, 다중 처리, 시분할 시스템

구분	시대	특징
제3세대	1960년대 중반 ~ 1970년대 중반	<ul style="list-style-type: none"> • IBM 360 시리즈, 유닉스 • 범용 시스템 • 다중 모드 시스템
제 4세대	1970년대 중반 ~ 현재	<ul style="list-style-type: none"> • 네트워크 시스템 • 가상기계, 분산 데이터 처리 • 개인 컴퓨터

* 제 1세대(1950년대)

운영체제의 효시

1952년 GM사에서 IBM 701의 운영체제 개발

초기 운영체제

- 설치 시간(Setup Time)과 철거 시간(Teardown Time) 감소에 관심
- 유휴시간(Idle Time) 최소화 : 일괄처리(Batch Processing) 작업
- 하드웨어 장치 독립성 : 시스템 파일 이름의 개념 출현

시간 지연 해결방안

- 버퍼링(Buffring)
- 스푼링(Spooling)

* 제 2세대(1960년대 초기)

처리량(Throughput) 향상에 관심

장치 독립성(Device Independence)을 통한 하드웨어 관리 편리성 개선

특징

- ① 다중 프로그래밍(Multiprogramming) 시스템
여러 사용자 프로그램들이 동시에 메인 메모리에 공존
- ② 다중처리(Multiprocessing) 시스템
다중 프로세서 사용을 통한 시스템의 처리 능력 향상
- ③ 시분할 시스템(Time Sharing System)
사용자들이 터미널을 통하여 컴퓨터와 접촉하여 대화형태로 작동
- ④ 실시간 시스템(Real Time System)
활용도보다 응답 시간 단축을 위해 많은 비용 투자
예시 : 미국 항공 회사의 SABRE 온라인 예약 시스템

* 제3세대(1960년대 중반 ~ 1970년대 중반)

1964년 IBM시스템/360 계열

범용 시스템으로 설계

- 단점

- = 기능이 필요 없는 분야의 사용자들 존재 : 실행 시간 과부하
- = 시스템 이해를 위한 매뉴얼 필요
- = 오류 발생시 수정에 많은 시간 필요
- = 시스템 유지에 많은 비용 부담

다중모드 시스템 - 일괄처리, 시분할처리, 실시간 처리, 다중처리 가능

- 단점

- = 사용자와 하드웨어 간의 데이터 처리를 위해 소프트웨어 계층 도입
- = 사용자는 소프트웨어에 대한 이해가 필수적으로 요구
- = 작업 제어 언어(Job Control Language)의 이해필요

* 제4세대(1970년대 중반 ~ 현재)

컴퓨터 네트워크와 온라인 처리 보편화

- 개인용 컴퓨터는 데이터 통신 인터페이스를 갖추어 터미널로 사용
- 지역적으로 멀리 떨어져 있는 여러 시스템들과 통신 가능

최근의 중요 관심사로 정보 보호가 대두

**사용자 친화 시스템(User Friendly System)을 위한
GUI (Graphic User Interface)**

- 1960 ~ 1970 : 텍스트(명령) 중심의 시스템
- 1980 : 사용자들에게 편리한 메뉴 지향적인 시스템
- 1990 : 도형(아이콘)과 마우스에 바탕을 둔 시스템

가상기계(Virtual Machine)

- 운영체제에 의해 만들어진 가상적 기계 관점

데이터베이스 시스템(Database System)

- 일정한 통제하에서 정보 제공 받을 권리가 있는 사람에게 정보 제공

분산 데이터 처리(Distributed Data Processing)

학습내용3 : 운영체제의 기능

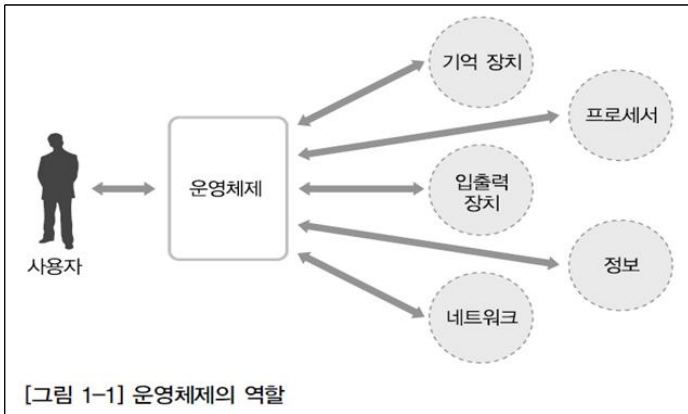
1. 운영체제 역할

컴퓨터 자원의 운영을 위해 수행되는 가장 중요한 소프트웨어

*자원 할당자

컴퓨터 시스템의 문제를 해결하기 위해 필요한 자원을 운영

중앙처리장치 점유시간, 기억장치 공간, 파일 저장 공간, 입출력장치 운영



2. 운영체제의 구성

* 사용자 명령 인터페이스

* 서브시스템 관리자

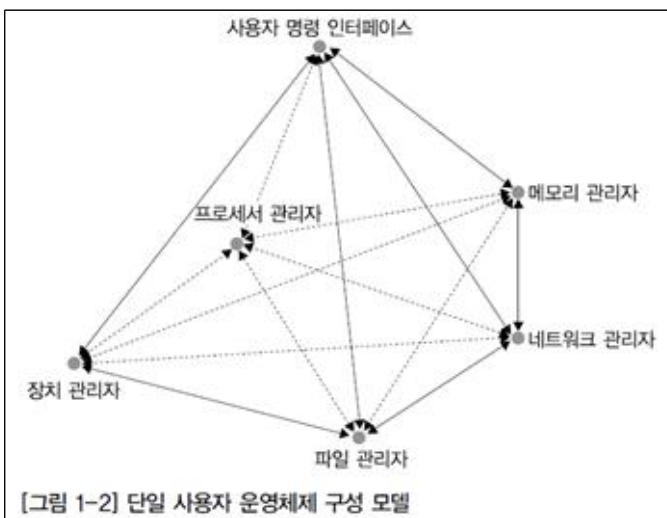
메모리 관리자

프로세서 관리자

장치 관리자

파일 관리자

네트워크 관리자



3. 사용자 명령 인터페이스

사용자 명령
인터페이스



사용자와 시스템의 대화 수단

DOS, 유닉스/리눅스 : 셸
MS 윈도우 : 아이콘, 메뉴

4. 메모리 관리자

- ① 프로그램의 메모리 요청의 적합성 점검
- ② 프로그램에 메모리 할당
- ③ 프로그램에 할당된 메모리 보호
- ④ 프로그램 종료 시 할당된 메모리 회수

5. 프로세서 관리자

- ① 명령어들 실행을 위한 작업 스케줄링
- ② 사용자의 작업 요청 수용 또는 거부

6. 장치 관리자

시스템 내의 모든 장치를 프로그램에 할당 또는 회수

7. 파일 관리자

- ① 사용자별로 파일 접근 권한을 부여
- ② 접근 권한에 따라 파일을 할당 또는 해제

8. 네트워크 관리자

네트워크에서 접근 가능한 자원을 관리

학습내용4 : 운영체제의 종류와 특징

1. DOS(Disk Operation System)

처음으로 개인용 컴퓨터에서 사용된 운영체제
1980년대와 90년 초까지 대부분의 PC 에서 사용

종류

- MS-DOS, PC-DOS, Dr-DOS 등
- MS-DOS(Microsoft DOS)가 일반적으로 많이 사용

특징

- 텍스트 기반의 사용자 환경(CLI)
- 사용자들은 필요한 명령어들을 숙지하여 사용
- 멀티태스킹이 불가능

2. OS/2(Operating System/2)

마이크로소프트와 IBM이 공동으로 제작한 컴퓨터 운영 체제
초기 80286컴퓨터를 위하여 설계되었으며 80386, 80486 에서 사용할 수 있도록 업그레이드 함

*장점

비슷한 시기의 일반 개인용 컴퓨터 운영체제보다 안정성이 뛰어남
응용 프로그램에 의한 시스템 다운 현상이 없음
파일 손실 가능성과 디스크의 단편화 현상이 적음
GUI 방식의 사용자 인터페이스인 WPS(Work Place Shell)을 사용
직관적으로 화면 관리가 가능
폴더 개념의 파일 관리

*단점

OS/2를 기반의 소프트웨어 개발자 부재
OS/2의 초기 버전이 MS-DOS 프로그램을 제대로 실행하지 못했기 때문에 OS/2의 수요가 거의 없었음

3. 윈도우

마이크로소프트에서 만든 GUI 환경의 운영체제 제품명

1985년 Windows 1.0을 발표한 후 꾸준히 버전 업그레이드
1990년 Windows 3.0부터 대중들에게 널리 알려지기 시작
1995년 Windows 95 가 출시- 통합된 TCP/IP, 전화 접속 네트워킹, 긴 파일 이름 지원 등이 포함

현재

- 멀티미디어 기능들이 대폭 보강된 Windows XP와 Windows 7이 널리 사용
- Windows 8.1 버전까지 출시 됨

4. 유닉스

1969년에 AT&T의 벨 연구소에서 개발한 운영체제로 처음에는 중형 컴퓨터에 사용하도록 고안
여러 가지 유틸리티가 공개되면서 일반 사용자들에게까지 확산

- 다중 사용자, 다중 작업 처리 가능
- 프로그램 개발이 쉬운 운영체제
- 일반 통신 서비스 프로그램은 대부분 유닉스를 기반으로 하고 있음

5. Linux

1991년 핀란드의 대학생이었던 리누스 토발즈에 의해 만들어진 운영체제
개인 컴퓨터용 유닉스에 해당

- 프로그램 소스코드가 공개돼 있어 프로그래머가 원하는 대로 특정기능 추가 가능
- 다양한 플랫폼에 사용 가능
- 무료

Linux의 종류

우분투, 레드햇, 데비안, 슬랙웨어 등

6. Mac OS

애플 컴퓨터의 매킨토시 계열 개인용 컴퓨터나 워크스테이션용 운영체제로 개인용 컴퓨터에 GUI를 처음으로 도입

1984년에 처음으로 세상에 선을 보인 이후로 거듭 발전

현재- 전통적인 클래식 시리즈(System 1.0 ~ Mac OS 9)와 새로운 Mac OS X 이후의 것으로 구분

특징

- 문서편집이나 그래픽분야에서 많은 사랑을 받고 있음
- 애플의 스마트폰인 아이폰을 위한 iOS 존재

7. 솔라리스

유닉스 계열의 운영체제 중 하나

리눅스보다 먼저 출시가 된 상용 운영체제

특징

- 처음에는 SUN에서 제작한 스팍(Sparc) CPU를 사용한 기종에서만 사용되는 전용 운영체제로 전문가들이 주로 사용
- 인텔 아키텍처의 대량 보급으로 인하여 인텔용 솔라리스도 출시 됨

【학습정리】

1. 운영체제란 사용자가 컴퓨터 시스템을 손쉽게 사용하고, 시스템 자원을 효율적으로 관리할 수 있도록 하는 프로그램 집합이다.
2. 시스템은 메모리 관리자, 프로세서 관리자, 장치 관리자, 파일 관리자, 네트워크 관리자로 구성되어 있다.