

프로그래밍의 기초

컴퓨터 하드웨어의 구성

컴퓨터 하드웨어의 구성

■ 컴퓨터 시스템

- 데이터를 처리하는 물리적인 기계장치인 하드웨어^{hardware}와 어떤 작업을 지시하는 명령어로 작성한 프로그램인 소프트웨어^{software}로 구성

■ 컴퓨터 하드웨어

- 하드웨어는 프로세서, 메모리(기억장치), 주변장치로 구성되고, 이들은 시스템 버스로 연결

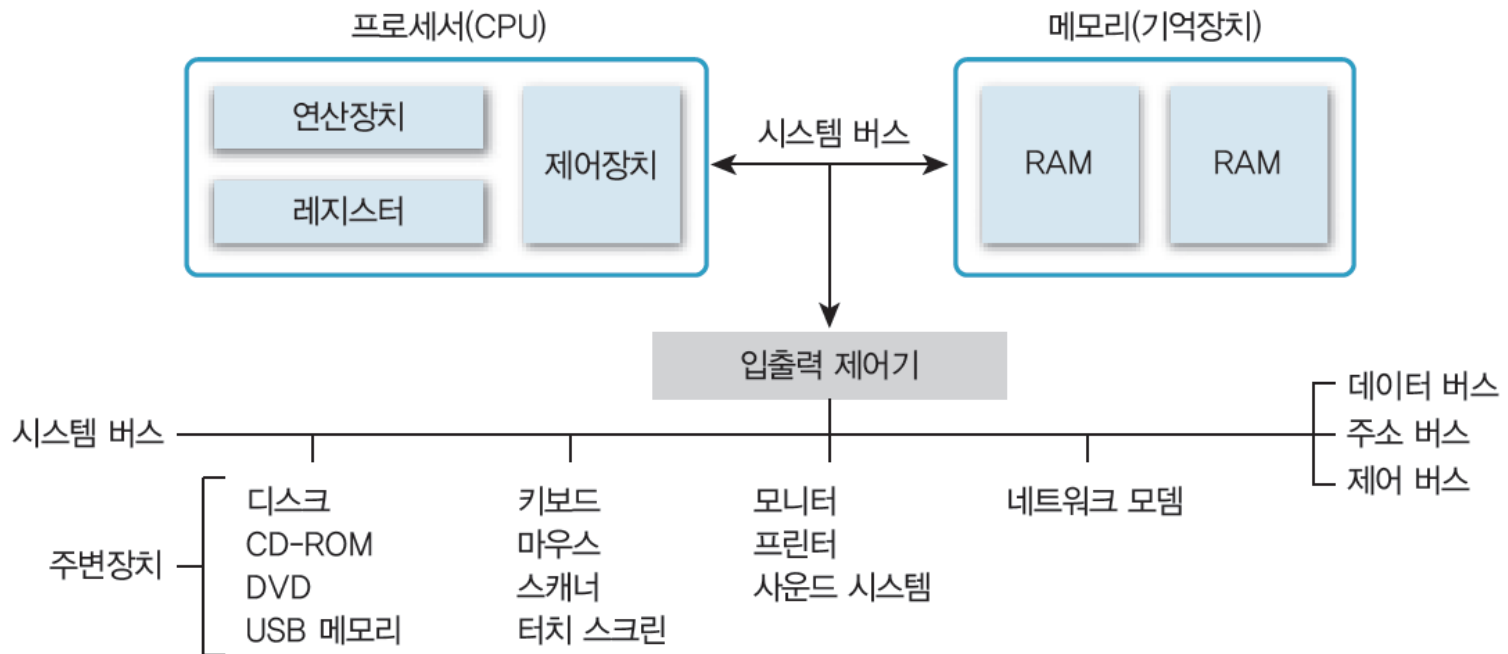


그림 1-1 컴퓨터 하드웨어의 구성

운영체제 (Operation System)

운영체제의 개념과 발전 목적

■ 운영체제의 개념

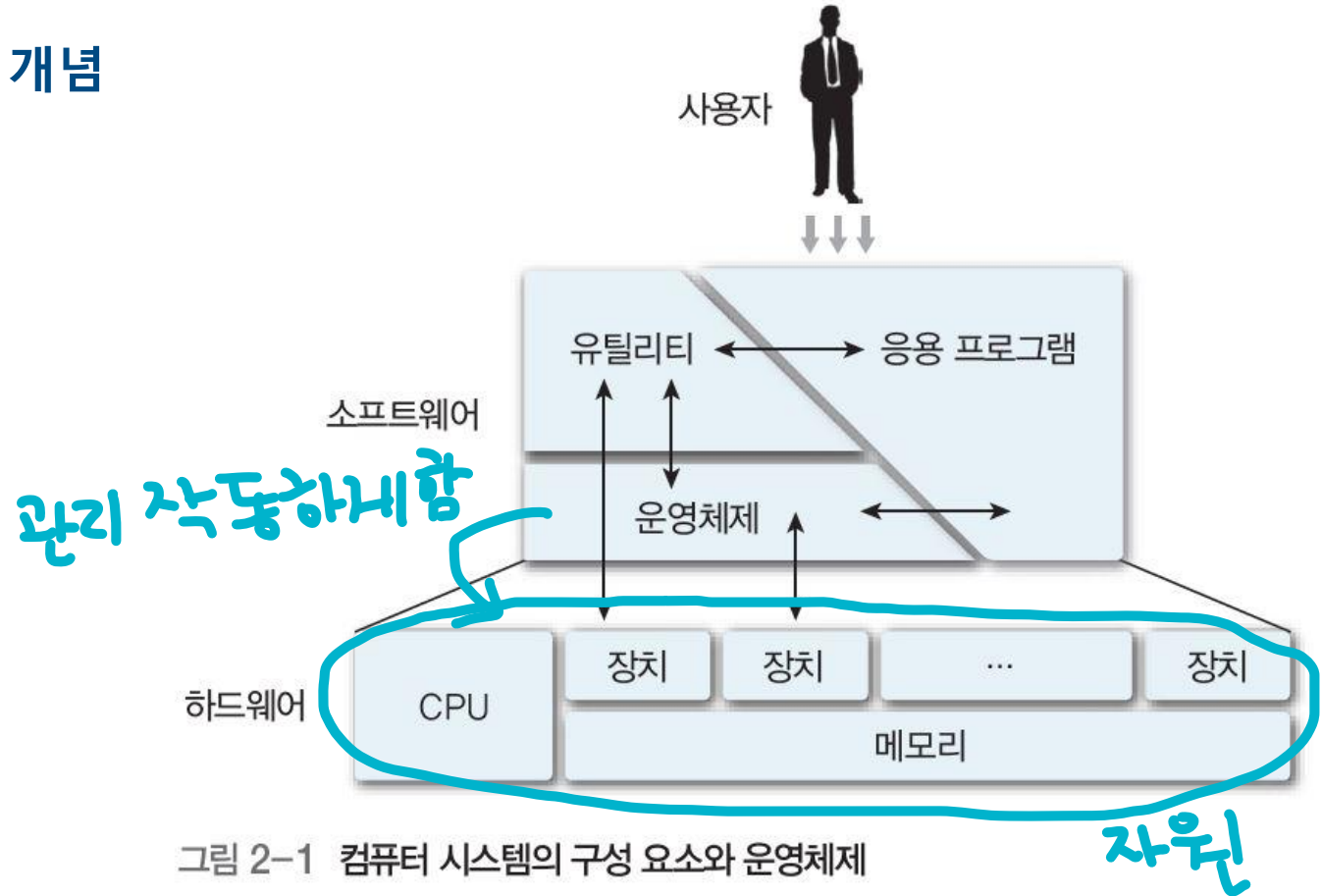
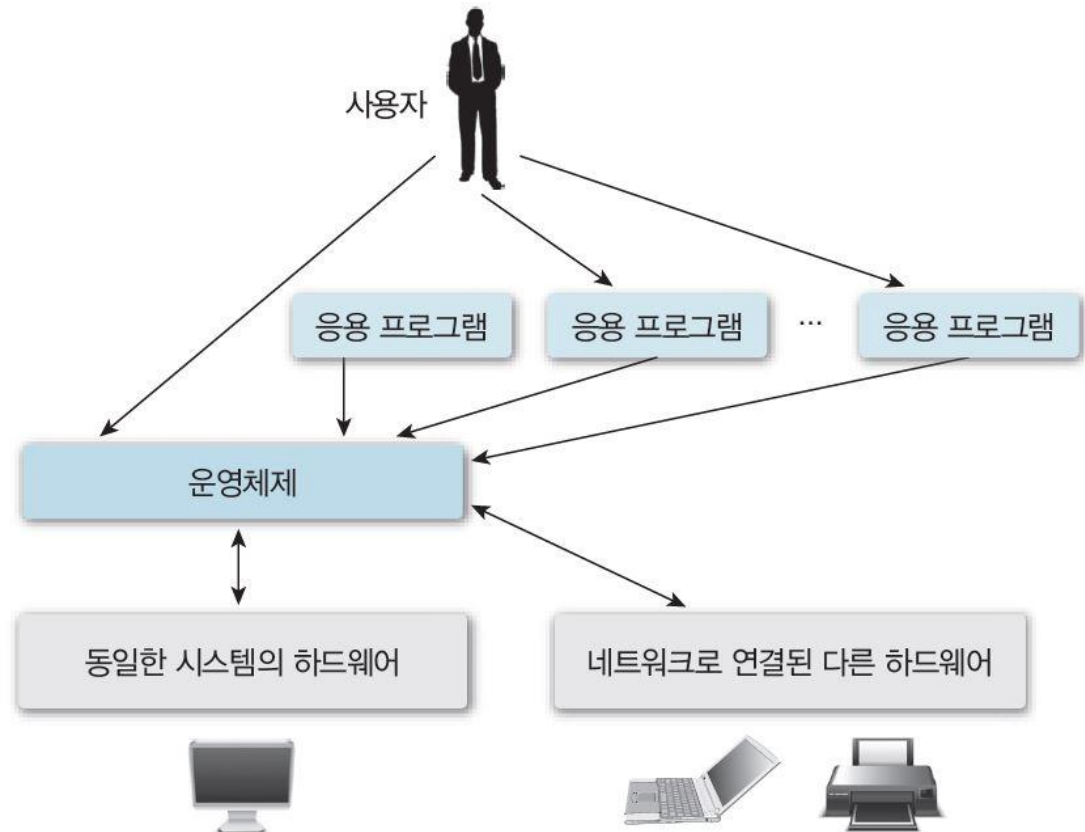


그림 2-1 컴퓨터 시스템의 구성 요소와 운영체제

- **사용자** : 컴퓨터를 사용하는 사람이나 장치, 다른 컴퓨터 등을 의미
- **소프트웨어** : 컴퓨터의 기능 수행에 필요한 모든 프로그램
- **하드웨어** : 기본 연산 자원을 제공하는 프로세서(CPU, 중앙처리장치), 메모리, 주변장치 등

운영체제의 개념

- 컴퓨터 자원 관리 면에서
운영체제의 정의



- 조정자** : 운영 요소 사용을 제어하면서 **사용자와 응용 프로그램 간에 통신**할 수 있게 함.
작업을 할 수 있는 환경만 제공
- 자원 할당자 혹은 관리자** : 각 응용 프로그램에 필요한 자원 할당, 자원 할당 방법 결정
- 응용 프로그램과 입출력장치 제어자** : 다양한 입출력장치와 응용 프로그램 제어

운영체제의 개념

■ 운영체제의 정의와 역할

■ 정의

- 사용자와 하드웨어 사이의 중간 매개체로 **응용 프로그램의 실행을 제어**하고,
- **자원을 할당 및 관리**하며,
- **입출력 제어 및 데이터 관리**와 같은 서비스를 제공하는 소프트웨어

■ 역할

- 하드웨어 및 사용자, 응용 프로그램, 시스템 프로그램 사이에서 인터페이스를 제공
- 프로세서, 메모리, 입출력장치, 통신장치 등 컴퓨터 자원을 효과적으로 활용하려고 조정·관리
- 메일 전송, 파일 시스템 검사, 서버 작업 등 높은 수준의 서비스를 처리하는 응용 프로그램을 제어
- 다양한 사용자에게서 컴퓨터 시스템을 보호하려고 입출력을 제어하며 데이터를 관리

운영체제의 발전 과정

■ 2000년대 이후 : 모바일 및 임베디드, 가상화 및 클라우드 컴퓨팅

- 21세기에 접어들어 스마트폰이나 태블릿 같은 모바일 기기 대중화
- 모바일 운영체제 Mobile Operating System : 모바일 장치나 정보 기기 제어 운영체제
 - 스마트폰용 : 구글(안드로이드^{android}), 애플(iOS), ~~마이크로소프트의 윈도우~~ 등이 대표적이다.
자바
포토진 → 리눅스
- 사물 인터넷 IoT, Internet of Things 기술 등장 : 각종 사물에 컴퓨터칩과 통신 기능 내장 인터넷에 연결
 - 초기에는 시스코 등 네트워크 기업, 나중에는 인텔, 퀄컴을 비롯한 반도체 칩 판매사가 사물인터넷을 전파하다가 이제는 애플이나 삼성전자 같은 제조사가 사물인터넷과 관련된 제품의 청사진과 플랫폼 구상 중
 - 특히 구글이나 아마존 등은 인터넷과 클라우드 기반 플랫폼에서 우위를 바탕으로 사물 인터넷 시장 선도하려고 하루가 멀다 하고 다양한 관련 기술과 서비스 발표.

최근 운영체제의 특징

■ 최근 운영 체제의 특징

■ 멀티미디어 자원의 관리

: 기존의 텍스트 위주의 자원에서 동영상, 사운드, 애니메이션 등의 여러 형태의 미디어 파일을 처리함

■ Windows 및 Graphics User Interface(GUI)의 일반화

: Windows 메뉴와 마우스 클릭으로 컴퓨터와 인간의 상호 작용이 가능함

■ 보안 기술 강화

: 인터넷 등의 통신망의 발달로 인한 접근제어 기술과 보안기술이 강화됨

■ 객체지향 기술 이용

: GUI, OLE에서 이용된 개념으로 모든 종류의 정보 객체

– 아이콘, 바탕화면, 문서, 동영상, 소리, 3D, 계속 갱신되는 뉴스, 컨트롤 등 대략 600개 이상의 종류-
를 관리, 실행하기 위한 기술임

운영 체제의 개념

■ 최근 운영 체제의 특징

■ UNIX의 기능 확장 및 대중화

UNIX의 기능이 다양하게 확장되었고 이러한 기능들은 Windows, Mac OS 등에서 이용되고 있으며, UNIX 자체도 다양한 버전이 출시됨

■ Windows의 대중화

MICROSOFT사의 OS인 Windows제품군이 대중화되어 PC들 사이에 호환성에서 발전을 이룸

■ Linux의 확대

PC용 UNIX라 할 수 있는 Linux가 open source 개념으로 급속하게 확산됨

운영체제 종류

■ DOS(Disk Operating System)



- 대표적인 단일 사용자, 단일 태스크의 운영체제임
- MS의 MS-DOS, IBM의 PC-DOS 등이 대표적임
- 지금은 거의 사용하지 않지만 현재 쓰고 있는 Windows 시리즈에 많은 영향을 줌

■ DOS의 특징

- 초기의 IBM PC를 위해 만들어진 운영체제 임
- 텍스트 기반의 명령어를 직접 입력하는 방식으로 작동됨
- 저 용량 PC를 기준으로 만들어졌기 때문에 속도가 다른 운영체제에 비해 빠른 장점을 가짐
- 하지만 사용자가 명령어를 일일이 암기해야 하는 단점이 있음

```
C:\WINDOWS\system32\cmd.exe

C:\Users\jin>help
특정 명령어에 대한 자세한 내용이 필요하면 HELP 명령어 이름을 입력하십시오.
ASSOC      파일 확장명 연결을 보여주거나 수정합니다.
ATTRIB     파일 속성을 표시하거나 바꿉니다.
BREAK      확장된 CTRL+C 검사를 설정하거나 지웁니다.
BCDEDIT     부팅 로딩을 제어하기 위해 부팅 데이터베이스에서 속성을 설정합니다.
CACLS      파일의 액세스 컨트롤 목록(ACL)을 표시하거나 수정합니다.
CALL       한 일괄 프로그램에서 다른 일괄 프로그램을 호출합니다.
CD         현재 디렉터리 이름을 보여주거나 바꿉니다.
CHCP       활성화된 코드 페이지의 번호를 표시하거나 설정합니다.
CHDIR      현재 디렉터리 이름을 보여주거나 바꿉니다.
CHKDSK     디스크를 검사하고 상태 보고서를 표시합니다.
CHKNTFS    부팅하는 동안 디스크 확인을 화면에 표시하거나 변경합니다.
CLS        화면을 지웁니다.
CMD        Windows 명령 인터프리터의 새 인스턴스를 시작합니다.
COLOR      콘솔의 기본색과 배경색을 설정합니다.
COMP       두 개 또는 여러 개의 파일을 비교합니다.
COMPACT    NTFS 분할 영역에 있는 파일의 압축을 표시하거나 변경합니다.
CONVERT    FAT 볼륨을 NTFS로 변환합니다. 현재 드라이브는
           변환할 수 없습니다.
COPY       하나 이상의 파일을 다른 위치로 복사합니다.
DATE       날짜를 보여주거나 설정합니다.
DEL        하나 이상의 파일을 지웁니다.
DIR        디렉터리에 있는 파일과 하위 디렉터리 목록을 보여줍니다.
DISKPART   디스크 파티션 속성을 표시하거나 구성합니다.
DOSKEY     명령줄을 편집하고, Windows 명령을 다시 호출하고,
           매크로를 만듭니다.
DRIVERQUERY 현재 장치 드라이버 상태와 속성을 표시합니다.
ECHO       메시지를 표시하거나 ECHO를 켜거나 끕니다.
```

Windows 7

■ Windows 7

- Windows 7은 빠르고 안정적인 성능과 함께 사용자가 보다 편리한 방법으로 PC를 사용하여 작업할 수 있도록 설계됨



Windows 7의 바탕 화면

■ Windows 7 주요 특징

- 속도가 빠르고 최소 설치 요구사항이 가벼워 짐
- 네트워킹 기능이 단순화되고 향상됨
- BitLocker 기본탑재로 보안기능 강화됨
- 가상화 기능으로 호환성 강화됨
- 사용자 인터페이스 편리, 장애 우들을 위한 기능 향상됨
- 미디어 센터의 타임머신 기능을 탑재하여 멀티미디어 기능이 크게 향상 됨

Windows 8과 Window 10

■ Windows 10

- 마이크로소프트는 2014년 10월 Windows 10의 시험판을 발표했음.
- 2015년 7월 출시.



Windows 10

Windows 8과 Window 10

■ Windows 10의 특징

- Windows 7과 Windows 8의 시작 버튼을 혼합시킨 새로운 형태의 시작 버튼
- 창 화면으로 작동되는 메트로 앱
- 새로운 데스크 뷰 버튼과 멀티 데스크탑
- 스냅 기능과 항상 빠른 파일 검색

구조	IA-32(32비트)	x86-64(64비트)
프로세서	1GHz 이상	
메모리(RAM)	1GB	2GB
그래픽카드	DirectX 9 그래픽 장치(WDDM 드라이버 포함)	
HDD	16GB 이상의 공간	
기타	Microsoft 계정과 인터넷	

Windows 10의 하드웨어 사양

윈도우 운영체제 기본 명령어

■ 윈도우 단축키

윈도우 단축키	단축키 기능	설명
윈도우 키 + E	윈도우 탐색기 실행	'내 컴퓨터' 와 동일
윈도우 키 + F	윈도우 검색창 실행	'시작 - 검색' 과 동일
윈도우 키 + R	윈도우 실행창 실행	'시작 - 실행' 과 동일
윈도우 키 + D	모든 창 최소화	다시 단축키 누르면 창이 원래대로 돌아옴
윈도우 키 + M	모든 창을 최소화	다시 단축키를 눌러도 창이 최대화되지 않음
윈도우 키 + Shift + M	최소화한 창을 최대화	'윈도우 키 + M' 단축키의 반대 동작
윈도우 키 + L	컴퓨터 잠금	잠금 해제시 사용자 비밀번호 필요
윈도우 키 + Pause/Break	시스템 등록정보 창을 띄움	'시작 - 제어판 - 시스템' 과 동일

Unix / Linux

동기

■ 유닉스/리눅스 운영체제

- 1970년대 초에 AT&T 벨연구소에서 개발된 이후로 지속적으로 발전
- 스마트폰, PC, 서버 시스템, 슈퍼컴퓨터에까지 사용되고 있음
- 소프트웨어 경쟁력의 핵심이 되고 있다.

■ 유닉스/리눅스 기반 운영체제

- 안드로이드(Android) OS
- iOS
- 맥(Mac) OS X
- 4. 리눅스(Linux)
- 5. BSD 유닉스(Unix)
- 6. 시스템 V
- 7. Sun 솔라리스(Solaris)
- 8. IBM AIX
- 9. HP HP-UX
- 10. Cray 유니코스(Unicos)

유닉스의 특징

■ 다중 사용자, 다중 프로세스 운영체제

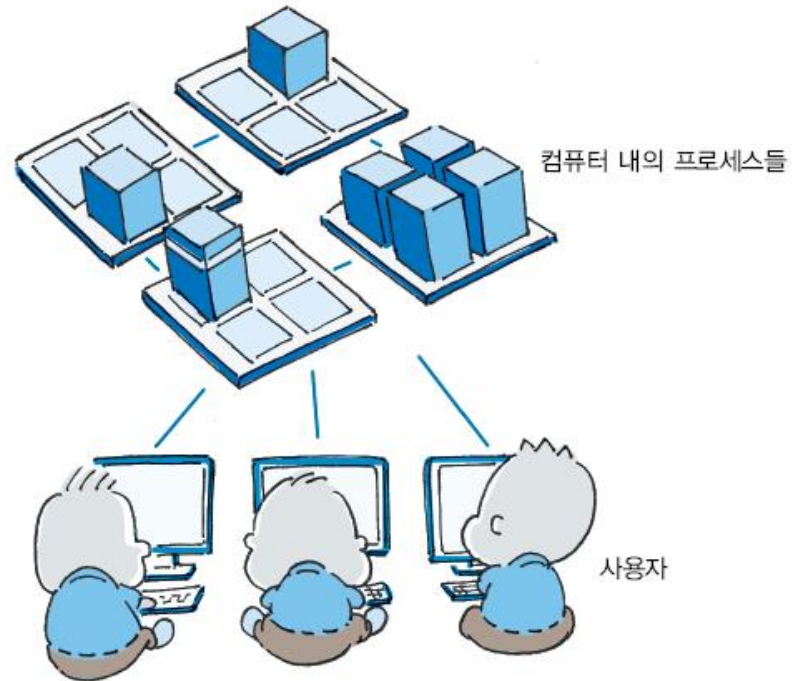
- 여러 사용자가 동시에 사용 가능
- 여러 프로그램이 동시에 실행
- 관리자 슈퍼유저가 있음.

■ 셸 프로그래밍

- 명령어나 유틸리티 등을 사용하여 작성한 프로그램

■ 훌륭한 네트워킹

- 유닉스에서부터 네트워킹이 시작
- ftp, telnet, WWW, X-window 등



리눅스

- PC를 위한 효율적인 유닉스 시스템

- 1991년 헬싱키 대학의 Linus B. Torvalds에 의해 개발됨

보안취약

- 소스코드가 공개

- 인터넷 상에서 지원자들에 의해서 기능 추가 및 확장됨
- 공용 도메인 상의 무료 OS

기술지원 한계

- 다양한 플랫폼에 포팅 가능

- PC, 워크스테이션, 서버 등

- GNU 소프트웨어와 함께 배포

- GNU/Linux 운영체제



리눅스 장점

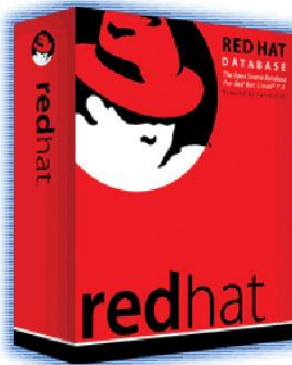
- 풍부하고 다양한 하드웨어를 효과적으로 지원
 - 대부분의 하드웨어를 지원하는 추세임
 - PC, 워크스테이션, 서버 등
- 놀라운 성능 및 안정성
 - Pentium으로도 충분히 빠르며 안전하게 수행
- 인터넷에 맞는 강력한 네트워크 구축
- 다양한 응용 프로그램 개발됨
- 무료 배포판

- 레드햇(RedHat): 상업용
- 우분투(Ubuntu)
- 페도라(Fedora)

CentOS

무요형

라즈비안, 아마존



리눅스 단점

- 책임지고 개발하는 사람들이 적음
 - 현재도 개발되고 있는 운영체제이며, 업그레이드 등의 사후 관리가 어려움
- 테스트 환경
- Linux는 컴퓨터에 대한 많은 지식을 요구함. 또한 자발적 정신으로 만들어지는 성격을 가지므로 단순한 문제는 사용자 스스로 해결해야 함

맥 OS(Mac OS)

● 1984년 애플 매킨토시 컴퓨터용 운영체제로 개발

- 개인용 컴퓨터에 GUI를 처음으로 도입

● 맥 OS X

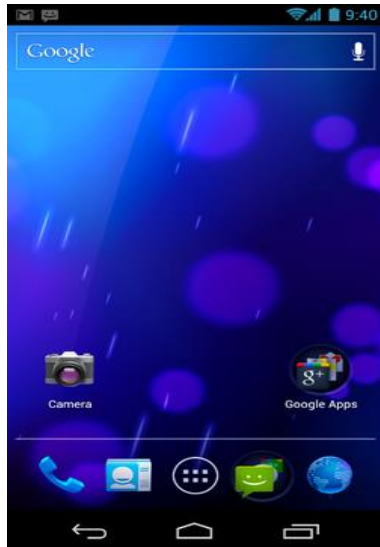
- 2002년에 NeXTSTEP 운영체제와 BSD 유닉스를 기반으로 개발
- 문서편집, 그래픽, 멀티미디어 등의 분야에서 많이 사용됨



모바일 기기용 운영체제

■ 안드로이드(Android)

- 리눅스 기반 모바일 기기용
- 주로 스마트폰, 태블릿 PC 등
- 개방형 운영체제로 소스 코드 등 공개



■ iOS

- 맥 OS X를 기반으로 개발된 모바일 기기용 운영체제
- 애플사의 iPhone, iPad, iPod



프로그래밍

프로그래밍

■ 컴퓨터 프로그램

- 어떤 목표를 이루기 위해 컴퓨터에 제시하는 계획

■ 코드 (Code)

- 암호 혹은 부호
 - 모스 부호 (Morse Code)
- "코드 짜기"
 - 컴퓨터가 이해하기 쉬운 암호(언어)를 작성
 - 프로그래머 ≡ 코더 (Coder)

스개박 코딩

■ 컴퓨터 프로그래밍(프로그래밍)

- 알고리즘을 특정한 프로그래밍 언어를 이용해서 구체적인 컴퓨터 프로그램으로 구현하는 기술

■ 빌드 (Build)

- 작성한 코드를 컴퓨터가 이해할 수 있는 형태로 번역하는 작업

■ 알고리즘 algorithm : 과제를 수행하기 위해 실행해야 하는 단계의 목록

“ 우리는 이미 알고리즘을 경험 했다 ”

알고리즘

■ 요리 레시피

된장찌개

주재료 바지락(1줌), 두부(1/4모)

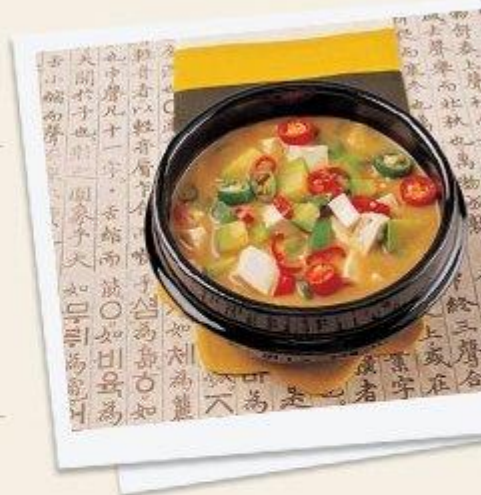
부재료 애호박(1줌), 양파(1줌) 표고버섯, 청양고추, 붉은고추,
국멸치(5마리), 다시마(사방10cm1장)

칼뜨름 쌀을 씻어 첫 번째 물은 재빨리 버리고 세번째 쌀을
씻어낸 물(5컵)

양념 된장(4), 고추장(0.5), 고춧가루(0.5)

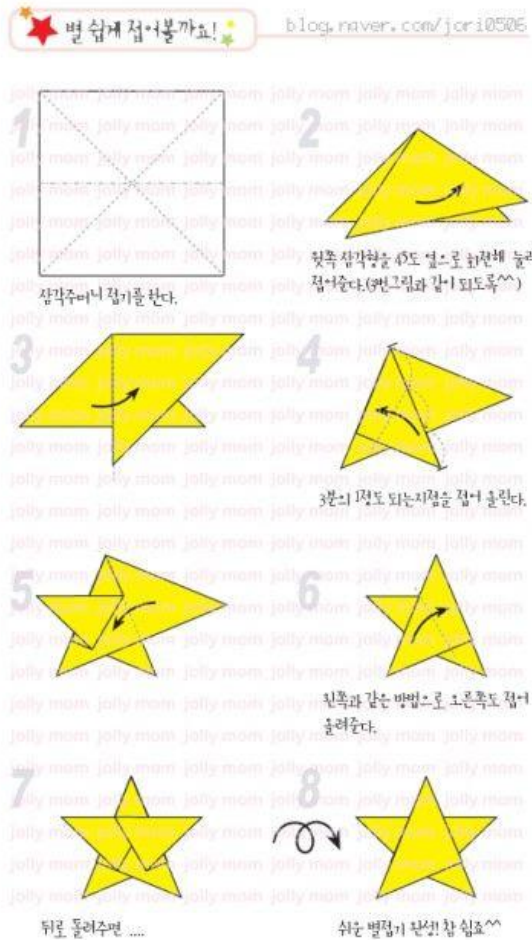
만드는 법

- 01 칼뜨름(5컵)에 다시마와 국멸치를 넣고 끓이다가 끓기
직전에 다시마만 건져내고 5분뒤 국멸치도 건져냅니다.
- 02 된장(4), 고추장(0.5)을 체에 내려 곱게 푼다.
- 03 바지락, 애호박, 양파, 표고버섯, 청양고추, 붉은고추 순으로
넣어 끓입니다.
- 04 거품은 걷어내고 두부(1/4모) 넣고 고춧가루(0.5)뿌려 마무리 합니다.



알고리즘

■ 종이접기



알고리즘

■ 레고



객체

알고리즘

- 알고리즘은 우리가 일을 처리하는 데 필요한 단계의 목록일 뿐이다.
- 어떠한 지시에 따라 행동할 때마다 알고리즘을 보게 된다.
- 게임, 레시피, 공예, 프라모델 등 모두 알고리즘의 물리적인 표현의 형태이다.
- 우리는 알고리즘을 따라 자연스럽게 컴퓨터를 사용하고 있다.

메일 보내기 : naver.com에 접속 -> 로그인 -> 메일 쓰기 -> 보내기 버튼

"결국 알고리즘은 모든 코드들의 핵심이다."

프로그래밍에서 다루는 것

- 데이터 (DATA)

- 숫자
- 문자
- 문자열
- Boolean : true or false

- 변수

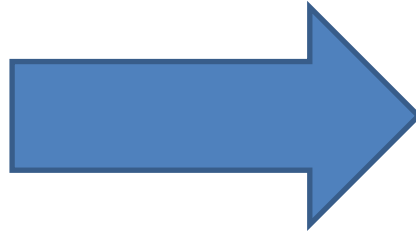
- 연산

- 비교

- 반복

- 기능 function and Method

알고리즘



프로그램

프로그래밍

■ 프로그래밍 언어의 종류

- 다양한 컴퓨터 프로그래밍 언어 존재
- 저급 언어
 - 컴퓨터가 이해하기 쉬운 언어
 - 기계어
- 고급 언어
 - 인간이 이해하기 쉬운 언어
 - 기계어로 바꾸는 과정 필요
 - JAVA
 - C++
 - C#

컴퓨터 사고력

■ 컴퓨터 사고력은

- 특별한 사고 패턴과 절차를 사용해서 문제를 찾아내고
- 해결하거나 컴퓨팅을 위한 프로그램을 고안하는 것
- 특히 문제 분해, 패턴 매칭, 추상화, 자동화를 말한다.

■ 컴퓨팅 사고력

- 문제 분해
- 패턴 매칭
- 추상화
- 알고리즘

프로그래밍 학습 방법

- 프로그래밍 언어의 단순 암기로는 해결할 수 없다.
- 문제를 찾아내고, 분해하고, 해결책을 찾고, 그 해결책의 응용까지 찾아낼 수 있어야 한다.
- 프로그램 오류에 대한 대처
 - 프로그래밍에서 발생하는 실패(오류 또는 다른 결과물)는 두려워 할 필요 없다.
 - 이 실패를 어떻게 다루느냐가 중요하다.
 - 실패로 생기는 불편한 마음은 당연하다.
 - 포기하지 않고 해결한다.
 - 프로그래밍과 오류는 절친이다.
 - 오류가 프로그램이 끝났다는 것이 아니다.
 - 실패는 새로운 길을 알려주는 지표이며,
 - 좌절은 우리가 어려운 일을 하고 있다는 증거이다.

정리합시다.

- 알고리즘

문제를 해결하기 위한 방법들의 과정 표현

- 프로그래밍

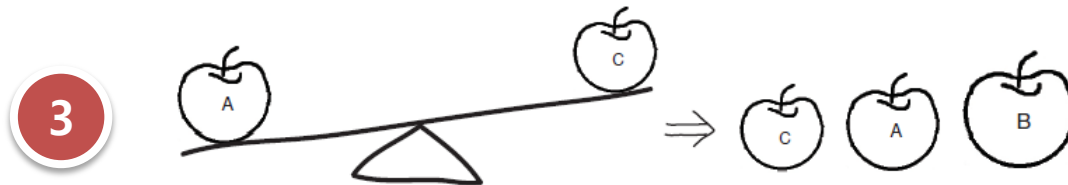
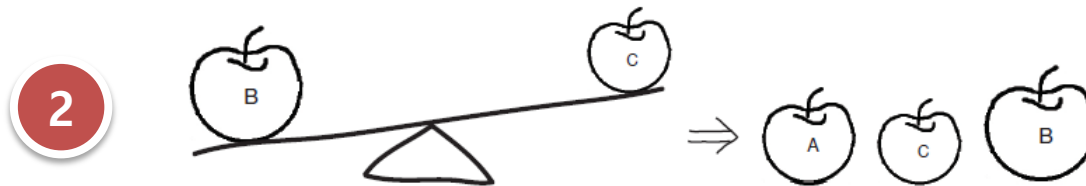
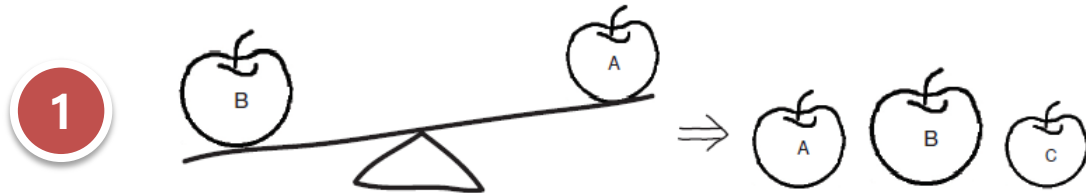
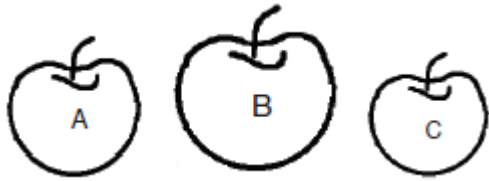
알고리즘을 특정한 프로그래밍 언어를 이용해서 구체적인 컴퓨터 프로그램으로 구현하는 기술

- 컴퓨터 사고력은

특별한 사고 패턴과 절차를 사용해서 문제를 찾아내고
해결하거나 컴퓨팅을 위한 프로그램을 고안하는 것
특히 문제 분해, 패턴 매칭, 추상화, 자동화를 말한다.

기초 알고리즘 연습

- 아래 그림은 사과의 크기 순서에 맞게 정렬하는 방법을 나열한 것 입니다.
- 순서에 맞게 해결 방법을 기술하세요.



기초 알고리즘 연습

- 두 수의 차 구하기 - 큰 수에서 작은 수
- 해결할 수 있는 방법을 순차적으로 기술하세요.

기초 알고리즘 연습

- 짝수와 홀수 판별하기
- 해결할 수 있는 방법을 순차적으로 기술하세요.

기초 알고리즘 연습

- 두 수중 큰 수 찾기
- 해결할 수 있는 방법을 순차적으로 기술하세요.

기초 알고리즘 연습

- 세 수중 가장 큰 수 찾기 – 두 가지 방법
- 해결할 수 있는 방법을 순차적으로 기술하세요.

기초 알고리즘 연습

- 최대값 찾기
- 해결할 수 있는 방법을 순차적으로 기술하세요.

기초 알고리즘 연습

- 두 변수 값 교환하기
- 해결할 수 있는 방법을 순차적으로 기술하세요.

기초 알고리즘 연습

- 작은 수에서 큰 수까지의 합 구하기
- 해결할 수 있는 방법을 순차적으로 기술하세요.

기초 알고리즘 연습

- 특정 숫자 까지의 3의배수 합 구하기
- 해결할 수 있는 방법을 순차적으로 기술하세요.

기초 알고리즘 연습

- 1, -2, 3, -4, ... , -100 합 구하기
- 해결할 수 있는 방법을 순차적으로 기술하세요.