


들어가기

CONTENTS 학습목표

- 운영체제란 무엇인지 설명할 수 있다.
- 운영체제의 사용자 인터페이스 방식을 나열할 수 있다.
- 운영체제 자원관리를 구분하여 설명할 수 있다.
- 운영체제의 종류와 각 종류별 특징을 설명할 수 있다.





복습하기

LEARNING 지난 주차 복습

04주차 학습내용. 프로그래밍 언어

- 1 프로그래밍 언어 개요
- 2 프로그램 구현
- 3 고급 프로그래밍 언어 종류
- 4 프로그래밍 언어와 구성 요소
- 5 객체지향 프로그래밍



지난 주차 **복습**

복습하기

프로그래밍 언어 개요

- 기계어
 - 0과 1로 표현되는 프로그래밍 언어
- 저급 언어와 고급 언어
 - 저급 언어 : 컴퓨터가 이해하기 쉬운 언어
 - 고급 언어 : 사람이 이해하기 쉽고 친근한 언어

지난 주차 **복습**

복습하기

프로그램 구현

인터프리터	특징	컴파일러
실행되는 줄(라인) 단위 번역	번역 방법	프로그램 전체 번역
번역 과정이 비교적 간단하고 대화형 언어에 편리함	장점	한 번 컴파일 한 후에 매년 빠른 시간 내 전체 실행 가능
실행할 때마다 매번 기계어로 바꾸는 과정을 다시 수행해야 하기에 항상 인터프리터가 필요함	단점	프로그램의 일부를 수정하는 경우에도 전체 프로그램을 다시 컴파일 해야 함
즉시 실행	출력물	목적 코드
BASIC 등	언어 종류	FORTTRAN, COBOL, C 등

LEARNING

복습하기

지난 주차 **복습**

- 고급 프로그래밍 언어 종류
 - 베이직, 파스칼, C, C++, 자바 등
- 프로그래밍 언어와 구성 요소
 - 주석, 문장, 변수, 제어구조(순차, 선택, 반복 구조)
- 객체지향 프로그래밍
 - 객체지향 언어 : 자바

1

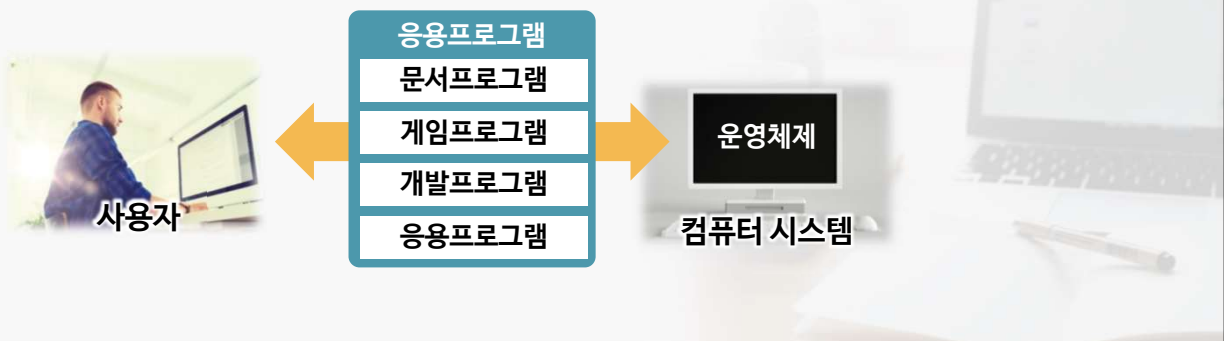
운영체제의 개요

1] 운영체제의 개념

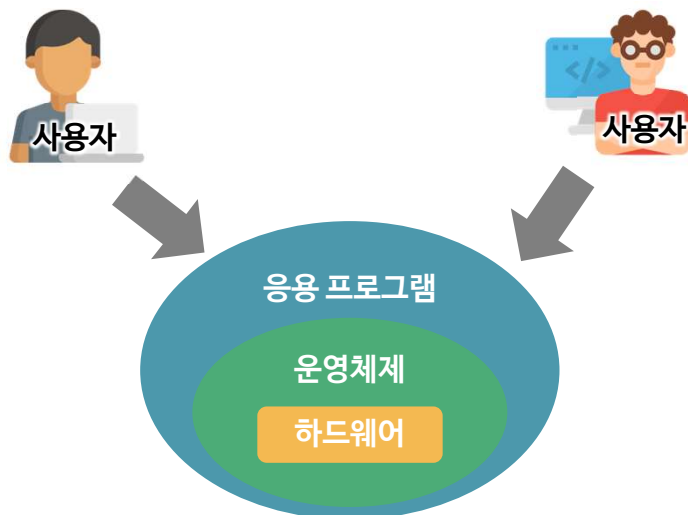
학습하기

운영체제

사용자와 컴퓨터 시스템 사이에 위치하여 **컴퓨터 시스템**의 전반적인 동작을 제어하고 조정하며, 사용자에게 편리성을 제공하여 한정된 시스템 자원을 효율적으로 사용할 수 있도록 관리하는 여러 시스템 프로그램들의 집합



컴퓨터 시스템의 구성



2] 운영체제의 목적

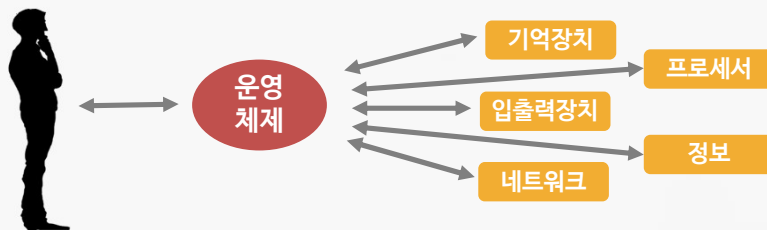
학습하기

- ✓ 컴퓨터 시스템의 자원을 **편리하게 사용할 수 있는 환경** 제공
- ✓ 컴퓨터 시스템의 자원들을 효율적으로 관리하여 **시스템의 성능 최적화**



3] 운영체제의 기능

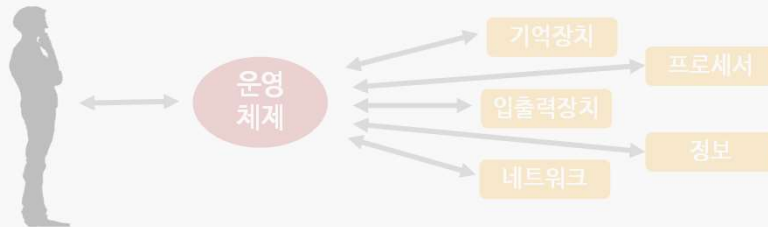
학습하기



- 1 사용자와 대화할 수 있는 사용자 인터페이스를 화면에 **출력**함
- 2 프로그램들을 사용자가 사용할 수 있도록 컴퓨터의 주기억장치에 **적재**함
- 3 프로그램들이 컴퓨터의 다른 하드웨어, 소프트웨어들과 원활히 작동할 수 있도록 **조정**함
- 4 **정보**가 디스크에 저장되고 검색될 수 있도록 **관리**함

3] 운영체제의 기능

학습하기



- 1 사용자와 대화할 수 있는 사용자 인터페이스를 화면에 출력함
- 2 프로그램들을 실행시 하드웨어가 성능을 잘 발휘하도록 관리
- 3 프로그램들이 컴퓨터의 다른 하드웨어와 소프트웨어들과 원활히 작동할 수 있도록 조정함
- 4 정보가 디스크에 저장되고 검색될 수 있도록 관리함

3] 운영체제의 기능

학습하기

초기화

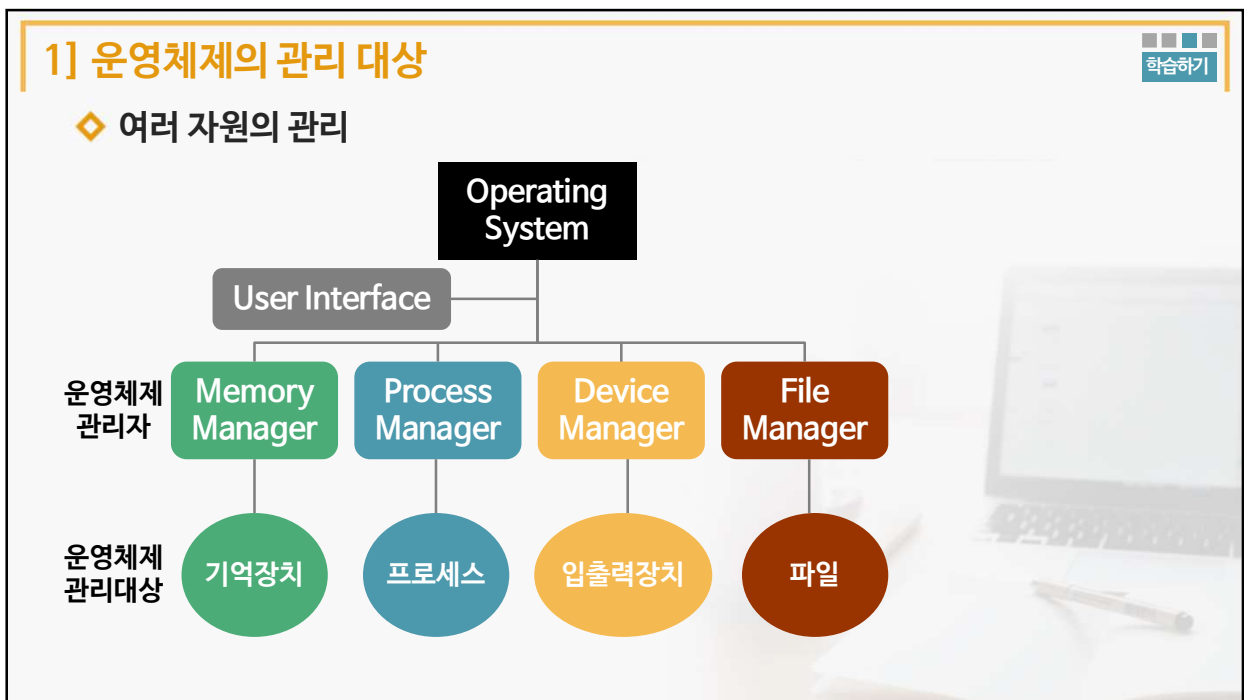
- 컴퓨터 시스템 초기화 설정 기능

자원 관리

- 중앙처리장치
- 저장장치
- 입출력장치
- 주기억장치
- 네트워크장치
- 파일시스템

기타

- 사용자와 컴퓨터 간 편리한 인터페이스 기능
- 오류 검사 및 복구 기능
- 사용자 계정관리
- 자원 공유 및 보안
- 하드웨어 및 자원의 공유



2] 기억장치 관리

학습하기

기억장치 관리

여러 프로그램을 실행하는 동안의 주기억장치, 보조기억장치를 관리하는 프로그램

◇ 기능

- ✓ 주기억장치의 공간이 사용 가능하면 어느 프로세스를 주기억장치에 적재할지 결정하여 적절한 주기억장치에 프로세스 적재
- ✓ 프로세스가 더 이상 주기억장치를 필요로 하지 않으면 다시 주기억장치를 회수

2] 기억장치 관리

학습하기

◇ 주기억장치를 관리하는 방식의 종류

단일 프로그램 방식

하나의 실행되는 프로그램을 주기억장치에 적재하여 실행하고 다른 프로그램을 실행하면 이전의 프로그램을 내리고 다시 실행할 프로그램을 적재하는 방식

Operating System

Program
and Date

Memory

다중 프로그램 방식

하나 이상의 프로그램이 동시에 주기억장치 내부에 적재하고 동시에 프로그램을 실행할 수 있도록 하는 방식

Operating System

Program 1
Program 2
Program 3
Program 4

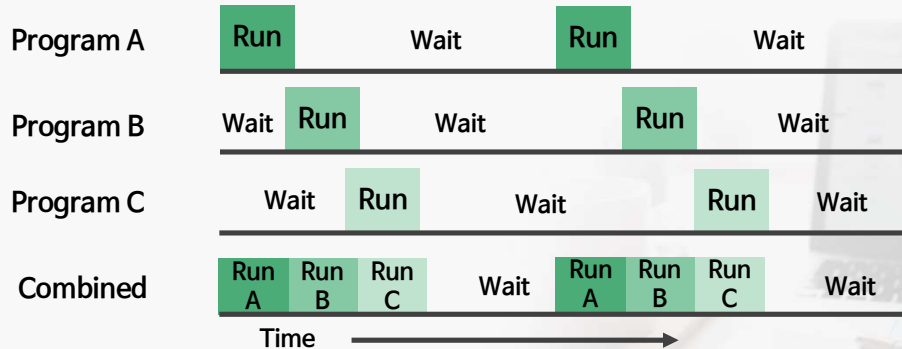
Memory

2] 기억장치 관리

학습하기

◇ 주기억장치를 관리하는 방식의 종류

예 3개의 프로그램을 수행하는 다중 프로그램 방식



3] 프로세스 관리

학습하기

◇ 기능

- ✓ 프로세스의 생성과 삭제, 프로세스의 중지와 계속, 프로세스의 동기화 등의 기능을 수행

◇ 프로세스의 정의

프로세스

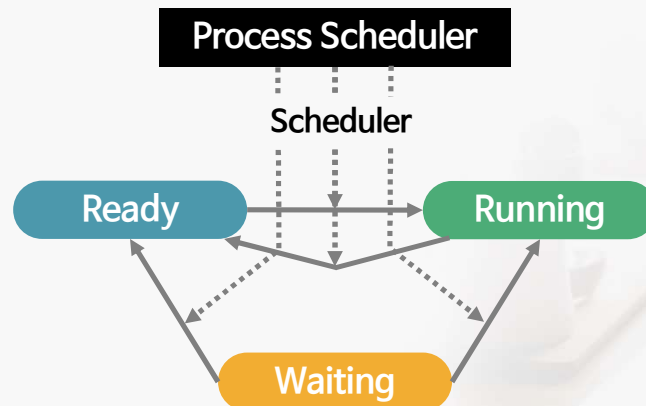
- 실행중인 프로그램
 - 시작했지만 아직 종료되지 않은 프로그램으로 주기억장치에 적재되어 있는 프로그램
- CPU에 선택되어 처리가 수행되는 프로그램

3] 프로세스 관리

학습하기

◇ 프로세스의 상태

“ 프로세스의 상태 변경은 프로세스 스케줄러에 의해 관리 ”



3] 프로세스 관리

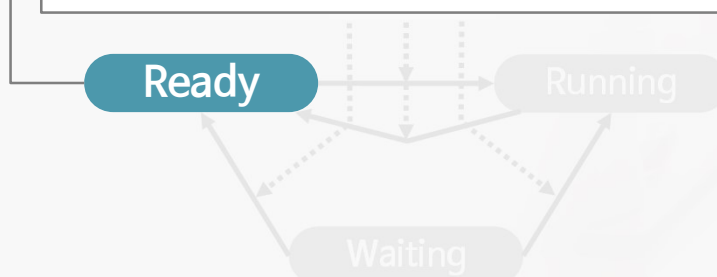
학습하기

◇ 프로세스의 상태

“ 프로세스의 상태 변경은 프로세스 스케줄러에 의해 관리 ”

준비 상태 (Ready State)

프로세스가 주기억장치 등 필요한 자원들을 할당받은 상태에서 프로세서(CPU)를 할당받기 위해 기다리고 있는 상태

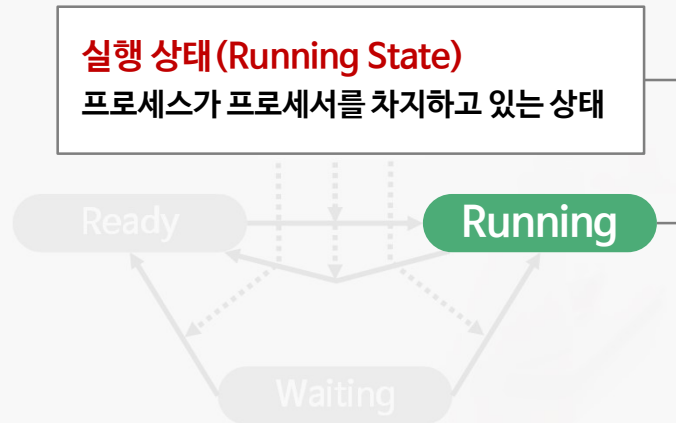


3] 프로세스 관리

학습하기

◇ 프로세스의 상태

“ 프로세스의 상태 변경은 프로세스 스케줄러에 의해 관리 ”

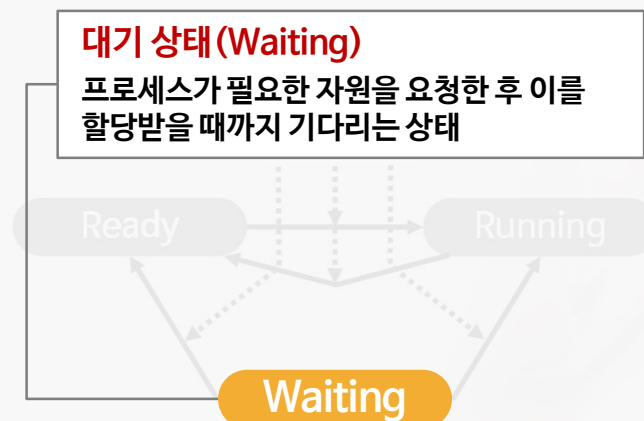


3] 프로세스 관리

학습하기

◇ 프로세스의 상태

“ 프로세스의 상태 변경은 프로세스 스케줄러에 의해 관리 ”



3] 프로세스 관리

학습하기

◇ 프로세스의 상태

“ 프로세스의 상태 변경은 프로세스 스케줄러에 의해 관리 ”

Process Scheduler

시스템 이상이나 과부하 등으로 인해 프로세스가 기억장치를 빼앗기는 경우

보류된
준비상태

보류된
대기상태

Waiting

4] 장치 관리

학습하기

◇ 기능

✓ 입출력장치의 효과적인 할당과 회수



입출력장치 관리 예:



4] 장치 관리

학습하기

◇ 기능



프로세스의 입출력

- 실행 중인 프로세스의 입출력을 **인터럽트 (Interrupt)** 를 이용하여 처리

- 1 실행 중인 프로세스가 입출력을 요구하면 실행을 잠시 멈춤
- 2 요구한 입출력에 해당하는 장치를 할당함
- 3 입출력이 종료되면 다시 입출력장치 등의 자원을 회수하여 중단된 시점부터 프로세스의 실행을 다시 실행함

4] 장치 관리

학습하기

◇ 기능



프로세스의 입출력

- 실행 중인 프로세스의 입출력을 **인터럽트 (Interrupt)** 를 이용하여 처리

- 1 컴퓨터 장치나 프로그램에서 특정한 일이 발생했을 때 운영체제에게 특정한 서비스를 수행하도록 요구하는 이벤트
- 2 잠시 멈춤
- 3 입출력이 종료되면 다시 입출력장치 등의 자원을 회수하여 중단된 시점부터 프로세스의 실행을 다시 실행함

4] 장치 관리

학습하기

◇ 장치 제어기(Device Driver) = 드라이버

- ✓ 장치관리자와 하드웨어인 입출력장치 사이의 인터페이스를 담당하는 프로그램
- ✓ 하드웨어와 운영체제의 연결 고리가 되는 프로그램으로 하드웨어 구성요소가 운영체제 하부에서 적절히 작동하는데 필요한 프로그램



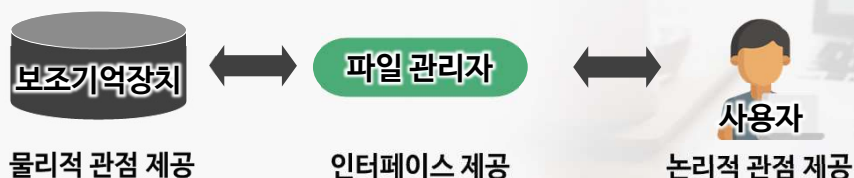
5] 파일 관리

학습하기

보조기억장치에서 저장되는 파일을 관리

◇ 역할

- 1 사용자와 보조기억장치 사이의 인터페이스 제공
- 2 실제 기억장치에 대한 사용자의 논리적인 관점과 실제 기억장치의 물리적인 관점을 연결하는 역할 수행



5] 파일 관리

학습하기

◇ 기능

- ✓ 파일의 접근을 제어
- ✓ 파일의 생성, 삭제, 수정을 감독
- ✓ 파일을 디스크의 어디에 저장할지를 감독
- ✓ 파일을 여러 사용자가 공유하는 기법을 제공
- ✓ 폴더의 구조를 제공하고 관리

3

운영체제의 분류

1) 초기 운영체제 및 여러 운영체제 형태

학습하기

운영체제가
없던 초기

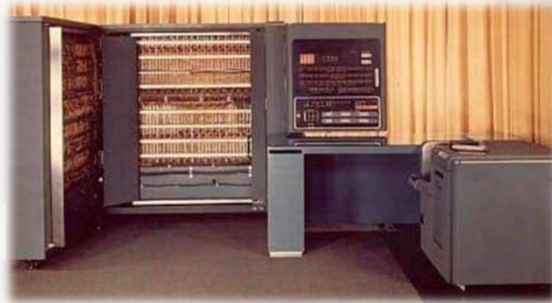
- 프로그래머가 컴퓨터 하드웨어를 직접 조작하였으므로 작업을 하나 수행하려면 준비 시간이 많이 소요됨
- 비효율적인 중앙처리장치의 이용으로 **작업 시간의 낭비를 초래함**

1) 초기 운영체제 및 여러 운영체제 형태

학습하기

운영체제의
개념 등장

- 1952년 **IBM-701** 컴퓨터의 사용을 위해 GM (General Motors) 연구소에서 개발한 프로그램

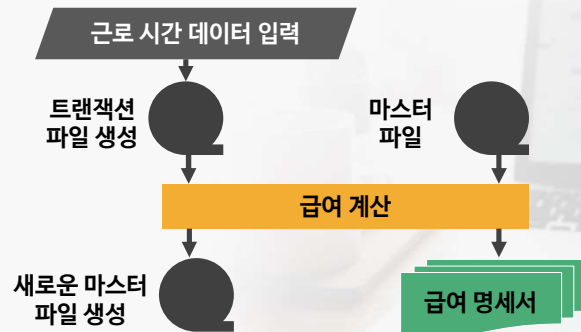


1) 초기 운영체제 및 여러 운영체제 형태

학습하기

일괄처리
방식

- 일정 기간 또는 일정량의 자료 처리를 모아 두었다가 **한 시점에 순서적으로 일괄처리**하는 자료 처리 방식



1) 초기 운영체제 및 여러 운영체제 형태

학습하기

일괄처리
방식

장점

- 한 번 시스템을 차지한 자료는 시스템 자원을 독점하여 처리하므로 **컴퓨터 시스템을 효율적으로 사용**

단점

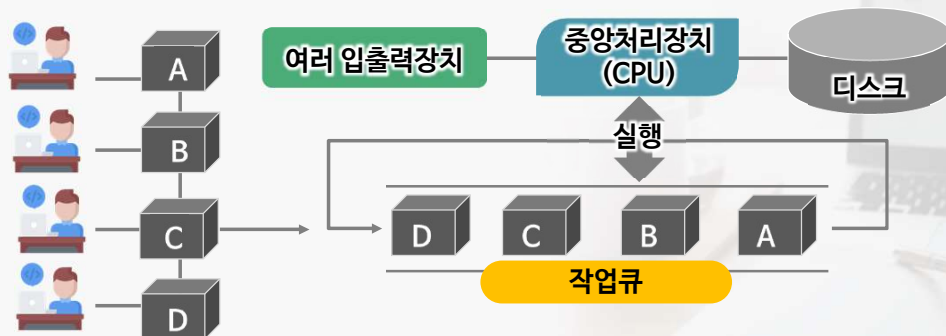
- CPU가 사용 가능하더라도 다른 처리를 할 수 없는 **유휴 시간(Idle time)**을 가짐
- 작업 제출과 작업 완료 사이의 시간(**Turn around time**)이 많이 걸려 작업 결과를 빠르게 확인할 수 없음

1) 초기 운영체제 및 여러 운영체제 형태

학습하기

시분할
시스템

- 하나의 시스템을 여러 명의 사용자가 단말기를 이용하여 여러 작업을 처리할 때 이용하는 처리 방법

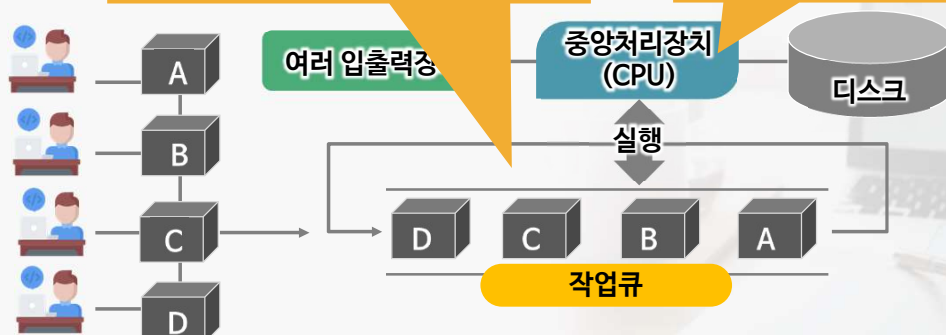


1) 초기 운영체제 및 여러 운영체제 형태

학습하기

시분할
시스템

- 하나의 시스템을 여러 명의 사용자가 단말기를 이용하여 여러 작업을 처리할 때 이용하는 처리 방법
- 실행 프로그램은 연산처리장치의 할당 시간을 돌아가며 부여 받는 타임 조각(Time slice) 방법으로 시분할 구현
- CPU의 이용 시간을 작게 분할하여 여러 사용자의 작업을 순환하며 수행



1) 초기 운영체제 및 여러 운영체제 형태

학습하기

시분할
시스템

장점

- 단말기에서 컴퓨터와 **대화형식**으로 프로그램을 작성하거나 실행할 수 있음
- 대화형식으로 진행할 수 있는 기능이 있어서 컴퓨터가 한 시각에 하나 이상의 문제들을 해결하게 함으로써 중앙처리장치의 유휴 시간을 줄일 수 있음

단점

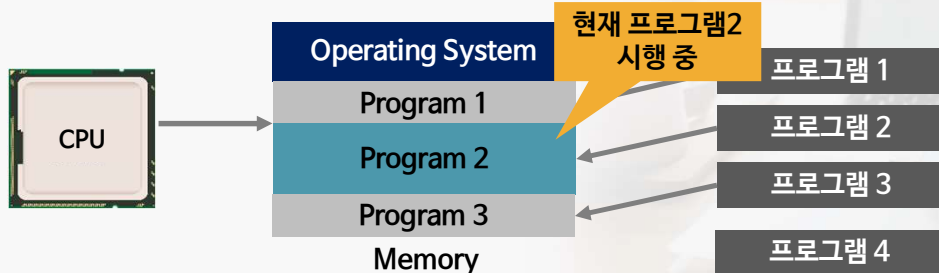
- 시스템의 사용량과 사용하는 단말기의 수가 적정해야 하고, 그렇지 않으면 시스템의 반응속도가 현저하게 떨어짐
- 기억장치 관리 기법, CPU 스케줄링 기법 등이 필요하므로 운영체제가 복잡함

1) 초기 운영체제 및 여러 운영체제 형태

학습하기

다중프로그래밍
시스템

- 2개 이상의 여러 프로그램을 주기억장치에 동시에 저장하고 하나의 CPU로 실행하는 방식
- 하나의 프로그램이 주변장치의 처리를 기다리는 동안 다른 프로그램이 CPU에서 실행될 수 있게 함으로써 전체적인 시스템의 처리 효율을 높이는 방식



1) 초기 운영체제 및 여러 운영체제 형태

학습하기

다중프로그래밍
시스템

- 여러 개의 프로그램을 준비 상태에 두고 관리하며 다른 한 작업을 실행할 프로그램으로 선정하기 위한 **기억관리 기법**과 **CPU 스케줄링 기법**이 필요
- 한 프로그램이 입출력 대기상태가 되면 다른 프로그램이 CPU를 사용할 수 있도록 전환하고, 입출력 동작이 끝나면 **우선순위**에 따라 실행 가능한 프로그램 중에서 하나를 선정하여 실행

1) 초기 운영체제 및 여러 운영체제 형태

학습하기

다중프로그래밍
시스템

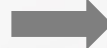
- 여러 개의 프로그램을 준비 상태에 두고 관리하며 다른 한 작업을 실행할 프로그램으로 선정하기 위한 기억관리 기법과 CPU 스케줄링 기법이 필요
- 한 프로그램이 입출력 대기상태가 되면 다른 프로그램이 CPU를 사용할 수 있도록 전환하고, 입출력 동작이 끝나면 **우선순위**에 따라 실행 가능한 프로그램 중에서 하나를 선정하여 실행

실행할 프로그램을 선정하기 위한 기준

작업요청
시간

입출력 동작
시간

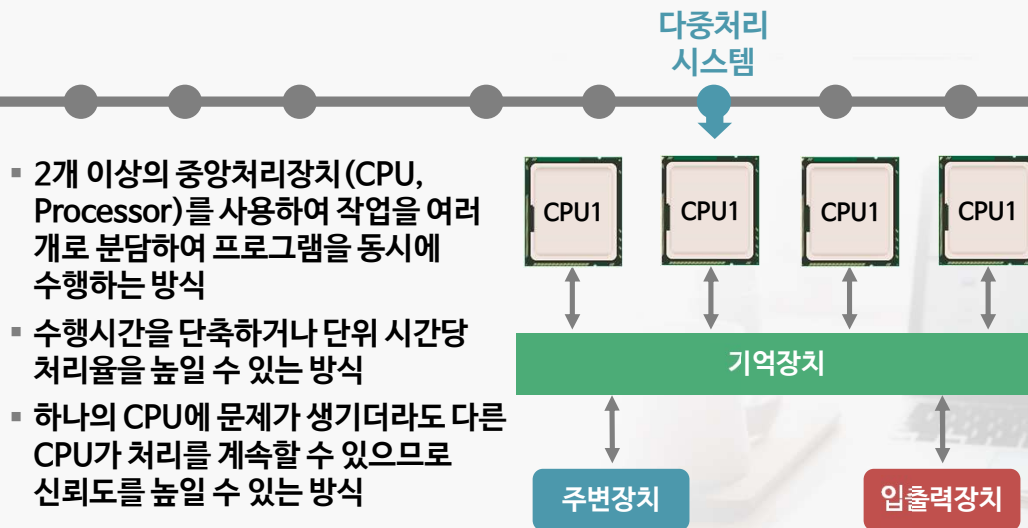
CPU 사용
시간



우선순위
결정

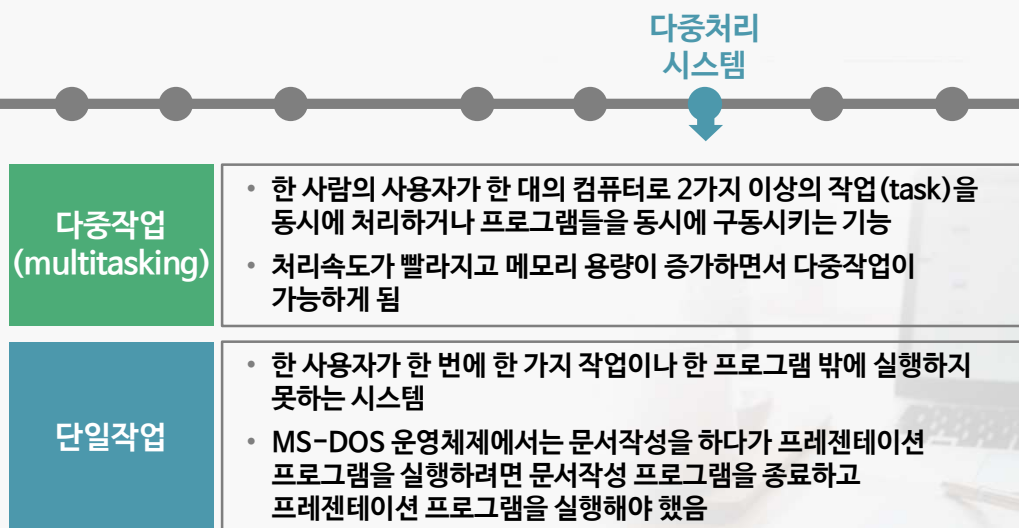
1) 초기 운영체제 및 여러 운영체제 형태

학습하기



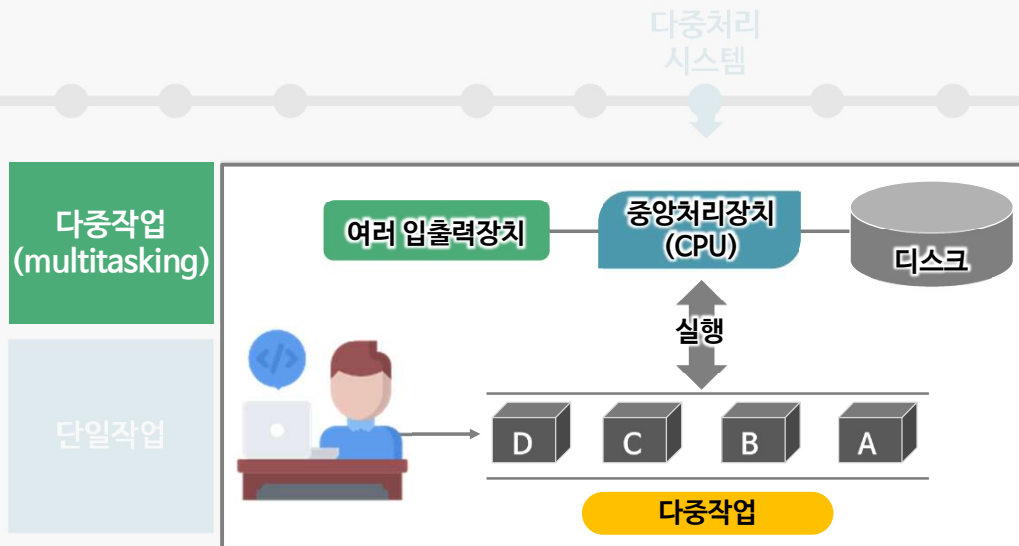
1) 초기 운영체제 및 여러 운영체제 형태

학습하기



1) 초기 운영체제 및 여러 운영체제 형태

학습하기

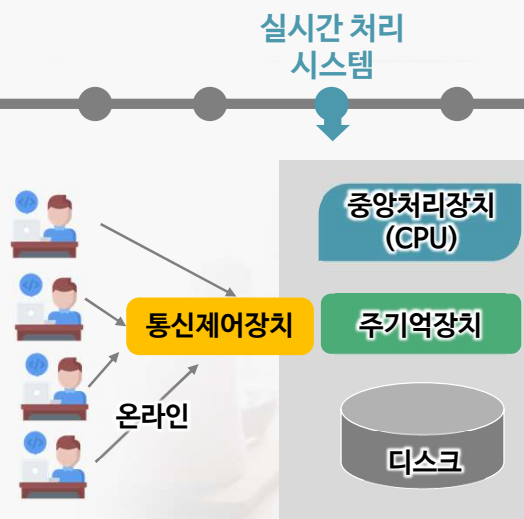


1) 초기 운영체제 및 여러 운영체제 형태

학습하기

- 처리를 요구하는 작업이 발생할 때마다 지정된 짧은 시간 내에 작업을 처리하여 확실한 응답이나 출력을 보장하는 시스템

예 미사일 방어 시스템,
좌석 예약 시스템,
은행의 예금 업무



1] 초기 운영체제 및 여러 운영체제 형태

학습하기

실시간 처리
시스템

장점

- 자료가 발생한 단말기를 통해 그 처리가 가능함
 - ⇒ 사용자가 편리
 - ⇒ 처리 시간의 단축
 - ⇒ 처리 비용의 절감

단점

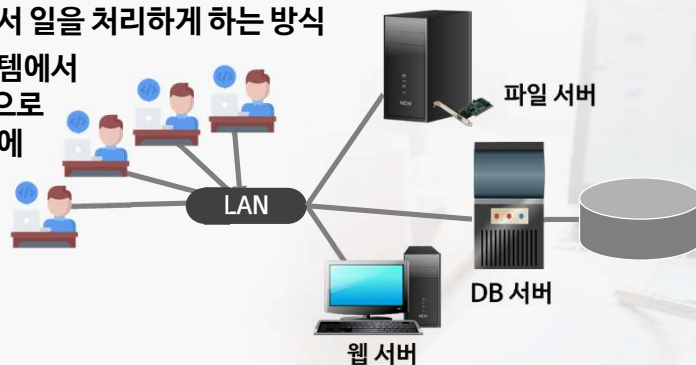
- 시스템에 오류가 발생하면 심각한 문제가 발생할 수 있는 위험성이 있는 시스템

1] 초기 운영체제 및 여러 운영체제 형태

학습하기

분산 처리
시스템

- 네트워크를 통해 연결된 여러 자료 저장 장소와 컴퓨터 시스템에 작업과 자원을 나누어 서로 통신을 하면서 일을 처리하게 하는 방식
- 하나의 대형 컴퓨터 시스템에서 수행하던 기능을 물리적으로 분산된 여러 개의 시스템에 분담시킨 후 네트워크를 통하여 상호 교신하여 일을 처리하는 방식



1] 초기 운영체제 및 여러 운영체제 형태

학습하기

분산 처리
시스템

장점

- 여러 개의 자료 저장 장소와 시스템을 이용하여 동시에 여러 작업을 수행함으로써 성능이 향상 될 수 있음
- 자료도 복사본을 여러 곳에 유지할 수 있어 신뢰도를 높일 수 있음
- 네트워크에 새로운 처리 시스템을 추가함으로써 쉽게 시스템 확장도 가능함

2] 운영체제의 발전 과정

학습하기

세대	특징	의미
0세대 1940년대	운영체제가 없었음	기계어 사용
1세대 1950년대	일괄처리 시스템	IBM 701용 OS 단일 흐름 일괄 처리
2세대 1960년대	다중 프로그래밍, 시분할 시스템 등장, 실시간 처리 시스템	고급 언어로 운영체제 개발
3세대 1960년대 중반 ~70년 중반	IBM S/360, S/370 등장, 다중모드 시스템 도입	일괄처리, 시분할 처리, 다중 처리를 하나의 시스템에서 제공(다중모드), 유닉스 개발
4세대 1970년 중반 ~1989년	GUI 방식 발전, 분산 처리 시스템 개념, 개인용 컴퓨터 OS 등장	개인용 컴퓨터와 워크스테이션 등장, TCP/IP의 등장
5세대 1990년대~	분산 처리 시스템 실현, 지식 기반 시스템 등장, 인공 지능 시스템 실현	MS 윈도우즈의 강세, 다양한 GUI 방식의 운영체제

4

운영체제의 종류

1) 도스

학습하기

◇ 도스(DOS : Disk Operation System)



디스크 중심의 명령행 사용자 인터페이스(Command Line User Interface) 방식의 운영체제

1981년 8월 버전 1.0으로 시작

```

current date is Tue 1-01-1980
Enter new date:
current time is 7:48:27.13
Enter new time:

The IBM Personal Computer DOS
Version 1.10 (C)Copyright IBM Corp 1981, 1982

>>dir /w
COMMAND COM  FORMAT COM  CHKDSK COM  SYS COM  DISKCOPY COM
DISKCOMP COM  COMP COM  EXE2BIN EXE  MODE COM  EDLIN COM
DEBUG COM  LINK EXE  BASIC COM  BASICA COM  ART BAS
SAMPLES BAS  MORTGAGE BAS  COLORBAR BAS  CALENDAR BAS  MUSIC BAS
DONKEY BAS  CIRCLE BAS  PIECHART BAS  SPACE BAS  BALL BAS
COMM BAS

26 File(s)
>>dir command.com
COMMAND COM  4959  5-02-82 12:00p
1 File(s)
>>

```

IBM 개인용 컴퓨터인
IBM PC 호환 컴퓨터의 16비트
마이크로프로세서를 위한 운영체제

1] 도스

학습하기

◇ MS-DOS의 특징

특징	비고
단일 작업	여러 개의 응용 프로그램이 보호를 받으면서 동시에 기억장치에서 실행되는 것을 지원하지 않음 (단일작업)
제한된 메모리의 이용	RAM에서 640KB만 인식할 수 있도록 설계
8비트, 16비트 운영체제	8비트, 16비트의 CPU용으로 설계
하드웨어 설치 복잡	주변장치마다 고유한 드라이버가 필요
8.3 파일명	파일이름 8자의 이름과 3자의 확장자로 구성
명령행 인터페이스	그래픽 사용자 인터페이스가 아닌 문자를 입력하는 형식의 명령행 사용자 인터페이스

2] 윈도우즈

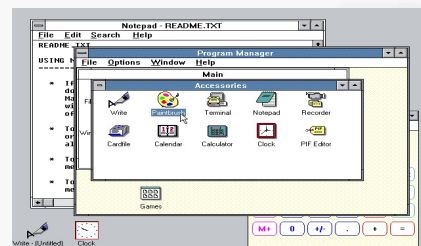
학습하기

◇ 윈도우(GUI 방식)

도스 운영체제의 명령행 인터페이스 방식에서 발전하여 그래픽 사용자 인터페이스(GUI) 방식을 채택한 운영체제

1983년

- 윈도우 3.0 처음 출시



1993년

- 윈도우 3.1에서 GUI 사용
- 윈도우 3.1도 운영체제라기 보다는 운영환경에 불과한 것이었고, 도스가 실질적인 운영체제

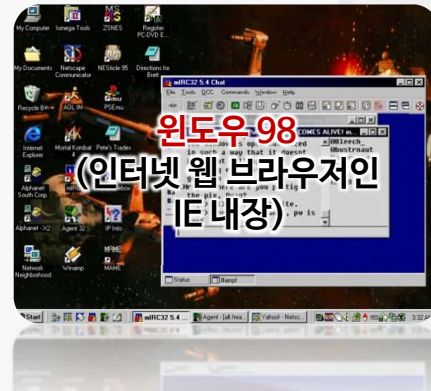
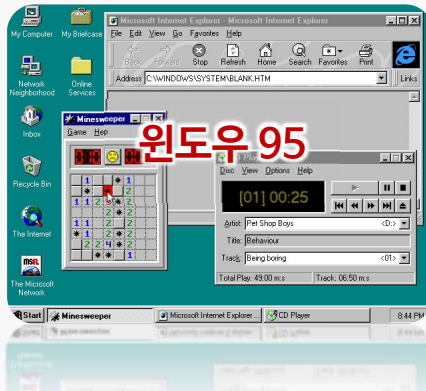
2] 윈도우즈

학습하기

◆ Windows 95/98

✓ 실질적인 GUI 운영체제

- 파일 시스템 이외에는 MS-DOS의 기능을 사용하지 않는 실질적인 그래픽 사용자 인터페이스를 채택한 하나의 독립된 운영체제



2] 윈도우즈

학습하기

◆ Windows NT/2000/2003

클라이언트/서버 환경에서 서버로 동작하는 시스템에 사용될 목적으로 개발

윈도우 NT

- 1988년 5월, 최초의 Windows NT 버전이 출시
- 1996년 발표된 NT 4.0은 성능의 증가와 비약적인 안정성 증가, 쉬운 사용법으로 UNIX의 위치를 위협

윈도우 2000

- 버전 5.0부터 2000이라 명칭이 바뀐 NT는 사용자 층을 개인까지 확장

윈도우 2003

- 서버 운영체제에서 필요한 기능인 보안, 신뢰성, 가용성 및 확장성 등의 기능들이 모두 포함

2] 윈도우즈

학습하기

◇ Windows Vista → 7, 8

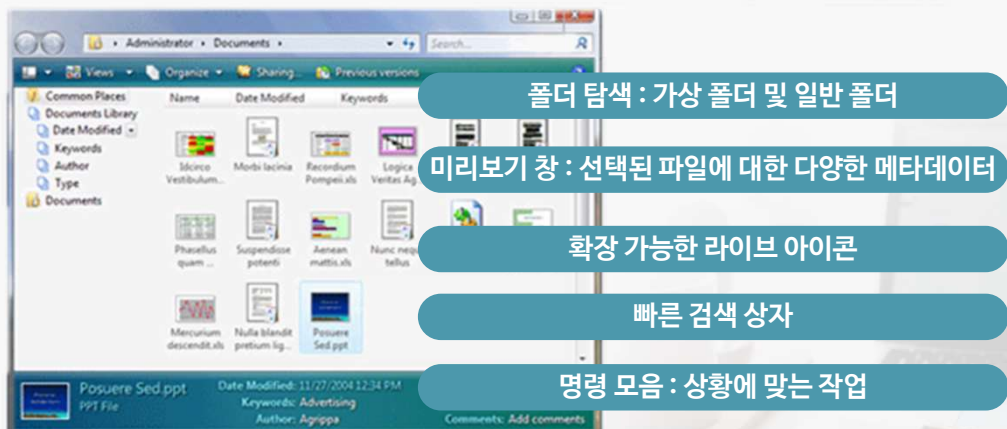
✓ 코드이름 롱혼(Longhorn)



2] 윈도우즈

학습하기

◇ Windows Vista → 7, 8



2] 윈도우즈

학습하기

◇ Windows Vista → 7, 8

◆ 윈도우 비스타 사양

구분	내용
사용자현황 (UI)	3D
편리성	사이드바 추가(날씨 · 날씨 · 시간 · 주식 정보 작업)
보안	스파이웨어 차단 솔루션 디펜더 탑재, 피싱 의심사이트 자동 식별
검색	실시간 검색기능 강화
멀티미디어	포토갤러리에 편집기능 추가, 별도 플레이어 없이 음악파일 재생
CPU	1GHz(기가헤르츠)
메모리	1~2GB(기가바이트)

2] 윈도우즈

학습하기

◇ Windows 10

◆ 2015년 코드이름 스레시홀드(Threshold)

유니버설 윈도우 플랫폼 (UWP)



3] 매킨토시 운영체제

학습하기

1976년 ● 애플1

- 스티브 워즈니악(Steve Wozniak)과 스티브 잡스(Steve Jobs)는 애플(apple) 컴퓨터를 창설하고 애플1을 발표



3] 매킨토시 운영체제

학습하기

1976년 ● 애플1

1981년 ● 제록스 스타 워크스테이션 (Star Workstation)

- 제록스(Xerox) 사는 그래픽 운영체제의 모체가 된 그래픽 운영환경을 채택한 제록스 스타 워크스테이션을 발표

3] 매킨토시 운영체제

학습하기

1976년 ● 애플1

1981년 ● 제록스 스타 워크스테이션 (Star Workstation)

1983년 ● 애플 컴퓨터의 리사(Lisa)

- 처음으로 그래픽 사용자 인터페이스 방식의 운영체제를 장착한 리사를 발표
- 일반 비즈니스 시장을 목표로 한 **최초의 개인용 컴퓨터**이며 최초로 그래픽 사용자 인터페이스의 운영체제를 탑재한 개인용 컴퓨터



3] 매킨토시 운영체제

학습하기



4] 유닉스

학습하기

1960년대 시분할 운영체제
멀틱스(Multics)

여러 사용자
동시 지원

대형 컴퓨터에서만
작동

복잡한 OS

고가

미니컴퓨터에서 작동이 가능하도록
기능을 대폭 축소, 단순화 시킨 OS 개발 필요

4] 유닉스

학습하기

1969년 AT&T 벨 연구소의 데니스 리치와 켄 톰슨이 초기 버전 OS 개발

1973년 이것을 발전시켜 새로 개발한 언어 C를 이용하여 시스템 PDP-11을 위해 만든 운영체제가 유닉스(Unix)



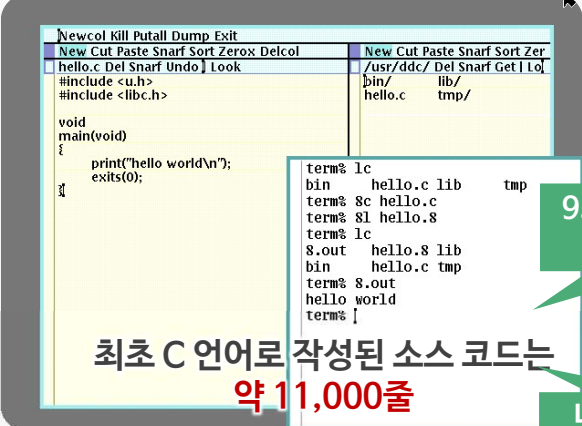
켄 톰슨

데니스 리치

4] 유닉스

학습하기

◇ 이식성이 좋은 운영체제



95%인 10,000줄은 C 언어로 작성

최초 C 언어로 작성된 소스 코드는
약 11,000줄

나머지 1,000줄은 어셈블리 언어로 작성

4] 유닉스

학습하기

◇ 유닉스(Unix)의 장점과 단점

항목	장점	단점
이식성, 호환성, 공개성	개발에 용이함	다양한 버전이 있음
파일 시스템	트리 구조로 사용이 편함	처리 속도 지연
시분할 방식	모든 사용자에게 컴퓨터 자원 균등 분배	업무의 우선 순위 결정이 어려움
명령어 축약	최소로 최대의 효과	초보자는 사용이 어려움
통신	다양한 유틸리티 제공 (UUCP, e-메일 등)	-
분산처리	서로 다른 기종 간에도 가능	-
프로그램 개발 도구	다양한 도구 제공	-
한글 지원	자국어 처리 시스템 이용	-
에러 메시지	-	간단하고 불분명함

5] 리눅스

학습하기

◇ 유닉스(Unix) 기반의 운영체제



1991년 핀란드의 **리누스 토발즈(Linus Torvalds)**에 의하여 개발된 공개 운영체제

일반공중허가
(GPL; General Public License)



리눅스의 소스 코드를 GPL에 따라 인터넷에 공개해 모든 사람이 자유롭게 사용

일반인들이 직접 리눅스의 개선에도 참여

5] 리눅스

학습하기

◇ 리눅스(Linux)의 특징

- 1 인터넷 프로토콜인 TCP/IP를 강력하게 지원하는 등 **네트워크 기능의 강화**
- 2 다중 사용자(Multi User), 다중 프로그래밍(Multi Programming) 방식을 지원하는 **안정성 있는 운영체제**
- 3 UNIX와 유사한 형태를 가지기 때문에 **익히기가 쉬움**
- 4 서버용 소프트웨어를 포함한 **여러 소프트웨어**를 기본으로 제공함
- 5 사용자가 사용하기 쉽도록 한 **배포판을 무료로 다운** 받을 수 있음

SUMMARY 정리하기

정리하기

운영체제의 개요

- 운영체제의 개념
 - 시스템의 하드웨어를 제어하고 사용자 및 응용 소프트웨어와 상호작용하는 시스템 소프트웨어의 일종
- 운영체제의 사용자 인터페이스 방식
 - CLI, GUI

운영체제 관리

- 기억장치 관리, 프로세스 관리, 입출력장치 관리, 파일 관리

SUMMARY 정리하기

정리하기

운영체제의 분류

- 시분할 시스템, 다중 프로그램 시스템, 다중 처리 시스템, 다중 작업, 실시간 처리 시스템, 분산처리 시스템 등

운영체제의 종류

- 도스, 윈도우(GUI 방식), 매킨토시 운영체제, 유닉스(Unix), 리눅스(Linux) 등

차시예고

ANNOUNCEMENT

정리하기

1주차 2주차 3주차 4주차 5주차 6주차 중간고사

데이터베이스

- 수고하셨습니다.