



운영체제의 역사

- ✓ 운영체제의 발전방향
- ✓ 운영체제가 없는 시스템
- ✓ 단순 일괄처리 시스템
- ✓ 스팔링 시스템
- ✓ 멀티프로그래밍 시스템
- ✓ 시분할 시스템
- ✓ 멀티프로세서 시스템을 위한 운영체제
- ✓ 분산처리 시스템을 위한 운영체제
- ✓ 실시간 시스템을 위한 운영체제

2020-06-03

Yong-Seok Kim (yskim@kangwon.ac.kr)

1



이해하고 넘어가야 할 내용들

- ✓ 단순일괄처리 시스템, 스팔링 시스템, 멀티프로그래밍 시스템 및 시분할 시스템의 특징 이해
- ✓ 각 단계별 발전과정에서 어떠한 필요성이 있었고 그 해결 방안들은 무엇인지에 대한 체계적인 이해
- ✓ 각 단계별 발전과정에서 주변장치의 개발 및 컴퓨터 하드웨어 발전과의 연관성 이해

2020-06-03

Yong-Seok Kim (yskim@kangwon.ac.kr)

2



운영체제의 발전방향

✓ 운영체제의 발전방향

- ✓ 초창기 운영체제: 고가의 하드웨어를 효율적으로 활용하는 측면 강조
- ✓ 1980년대 이후: 하드웨어 성능향상과 가격 하락으로 사용자 입장에서 편리하도록 하는 측면 강조
- ✓ 1990년대 이후: 인터넷의 급속한 보급으로 시스템의 보안 측면 강조

✓ 최근의 컴퓨터 구조 발전에 따른 기능들

- ✓ 멀티프로세서 (multi-processor): 프로세스/스레드들을 각 CPU에 골고루 분배하는 기능
- ✓ 대규모 병렬처리 (MPP: massively parallel processing): 하나의 작업을 위해 여러 개의 CPU들을 끓음으로 배분하는 기능
- ✓ 클러스터링 (clustering) : 통신망으로 연결된 컴퓨터들간에 부하를 적절히 분산하여 실행하는 기능
- ✓ 내장형 시스템 (embedded system): 컴퓨터 시스템의 소형화/휴대형/실시간성 특성에 따라 운영체제 크기의 축소, 절전 기능, 실시간 스케줄링 등의 기능 필요

2020-06-03

Yong-Seok Kim (yskim@kangwon.ac.kr)

3



운영체제가 없는 시스템

- ✓ 1950년대 초반 이전
- ✓ 프로그램 입력, 실행, 출력을 위한 절차를 사용자가 직접 제어
- ✓ 소프트웨어: 어셈블러, 초기의 컴파일러 등 사용
- ✓ 입출력 장치: 종이 테이프, 천공 카드 사용

2020-06-03

Yong-Seok Kim (yskim@kangwon.ac.kr)

4

단순 일괄처리 시스템

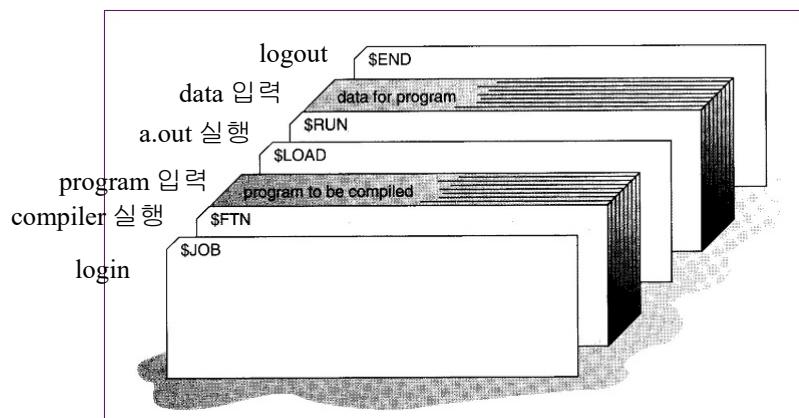
- ✓ 단순 일괄처리 시스템 (batch system)
 - ✓ 프로그램 실행 절차를 제어카드들을 활용하여 자동화
 - ✓ 프로그램 카드 읽기 → 컴파일링/어셈블링 → 실행 → 출력
- ✓ 입력 장치: 천공 카드, 자기 테이프
- ✓ 출력 장치: 라인프린터, 자기 테이프, 천공 카드
- ✓ 사용 환경: 오프라인 (off-line)
 - ✓ 프로그램 카드 묶음을 전문 오퍼레이터에게 제출, 나중에 출력을 수령
- ✓ 프로그램을 실행하기 위한 절차 예
 1. 자기테이프로부터 컴파일러를 메모리에 적재
 2. 컴파일러는 프로그램 카드 묶음을 읽고, 컴파일하여, 실행프로그램을 자기 테이프에 기록
 3. 자기테이프에서 실행프로그램 적재
 4. 실행 프로그램 실행 (데이터 카드를 읽고, 라인프린터에 출력)
- ✓ 입출력 속도향상을 위해 다수의 오프라인 입출력 장치들도 사용

2020-06-03

Yong-Seok Kim (yskim@kangwon.ac.kr)

5

일괄처리 방식의 카드묶음 구성



2020-06-03

Yong-Seok Kim (yskim@kangwon.ac.kr)

6



IBM System 360

- ✓ 1964.4.7 IBM이 Main frame인 System 360 발표
- ✓ 1968년 경제기획원 조사통계국에서 인구조사결과 분석용으로 도입
- ✓ 이후 System 370, System 390, ES 9000, System z9, System z10, Power Series (현재)

2020-06-03

Yong-Seok Kim (yskim@kangwon.ac.kr)

7



프로그램 카드 묶음 (CDC Cyber)



2020-06-03

Yong-Seok Kim (yskim@kangwon.ac.kr)

8



최초의 Hard Disk

- ✓ IBM 350 Data Storage Unit (5M char)
- ✓ 1956년, William Goddard and John Laynott
- ✓ 50 개의 24인치 디스크 (100 track/cylinder)
- ✓ 100 cylinder x 100 track x 5 sector x 100 characters x 7bit
- ✓ 1200 rpm, 8800 character / second
- ✓ 평균 seek time 600ms
- ✓ IBM 305 RAMAC 시스템에 장착 (진공관 사용)

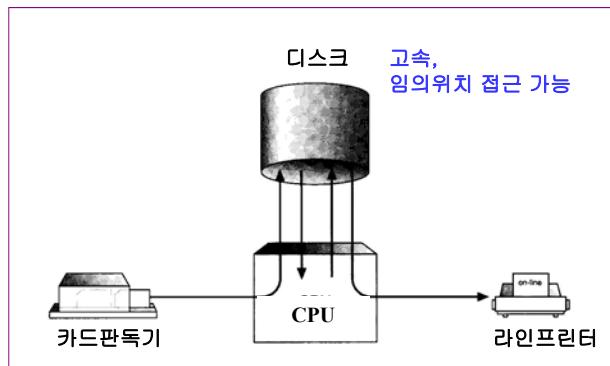


스풀링 시스템

- ✓ **스풀링 (spooling) 시스템**
 - ✓ SPOOL: Simultaneous Peripheral Operation On Line
 - ✓ 입출력 속도를 개선하기 위해 카드읽기, 실행, 프린터 출력을 동시에 진행
 - ✓ 고속이면서 임의위치 접근이 가능한 **하드디스크**의 개발로 가능
 - ✓ 디스크 상에 여러 작업들이 입력되어 있으므로 FCFS로 처리
- ✓ **입출력의 동시 진행**
 - ✓ 입출력 장치는 상대적으로 느리므로,
 - ✓ 디스크에 읽어들인 입력을 실행중에 카드 판독기에 다음 카드가 준비 되는대로 수시로 읽어서 디스크에 기록
 - ✓ 실행 결과는 디스크에 기록해 두고, 프린터가 다음 라인을 출력할 준비가 되는대로 수시로 디스크에 저장된 내용을 출력
- ✓ **사용자 환경**
 - ✓ 단순일괄처리 시스템과 동일



스풀링 시스템의 구성



2020-06-03

Yong-Seok Kim (yskim@kangwon.ac.kr)

11



멀티프로그래밍 시스템

- ✓ **멀티프로그래밍 (multi-programming) 시스템**
 - ✓ 메모리에 여러 작업들을 동시에 적재 ← 메모리 용량증가
 - ✓ 실행중인 작업이 입출력을 위해 대기하면, 실행가능 상태의 다른 작업으로 전환
 - ➔ 입출력 작업과 실행 작업을 병행하여 CPU 활용률 향상
 - ➔ 디스크 입출력 처리 시간도 활용
- ✓ **운영체제의 기능 확장**
 - ✓ 메모리에 적재된 작업들 간의 CPU 스케줄링 (단순히 FCFS)
 - ✓ 디스크에 대기중인 작업들 중에 적절한 것을 선택하여 메모리에 적재하는 Job 스케줄링
 - ✓ 작업들 간에 메모리 영역이나 입출력 장치의 보호 기능 필요
- ✓ **사용자 환경**
 - ✓ 여전히 단순일괄처리 시스템과 동일

2020-06-03

Yong-Seok Kim (yskim@kangwon.ac.kr)

12

시분할 시스템

✓ 시분할 (time sharing) 시스템

- ✓ 터미널 장치의 개발로 사용자들이 온라인으로 작업 가능
- ✓ → 다수의 사용자들이 동시에 사용하므로 이들간에 공평하게 실행 기회 (CPU 사용시간) 배분 필요
- ✓ → 프로세스들 간에 일정한 시간단위로 돌아가면서 실행하는 스케줄링 도입

✓ 사용자 환경

- ✓ 터미널을 통해 온라인으로 사용
- ✓ 사용자별 프로그램 및 데이터의 보관을 위해 파일 시스템 필요
- ✓ 프로세스 간의 보호, 사용자별 파일 보호 기능 필요
- ✓ 천공카드 입력기 불필요
- ✓ 출력도 일단 파일에 저장하고 필요한 것만 별도로 프린터로 출력

✓ 현재의 운영체제

- ✓ 시분할 시스템으로서, 가상메모리 기능도 보유
- ✓ 사용자의 편의성을 위해 GUI (Graphic User Interface) 제공

2020-06-03

Yong-Seok Kim (yskim@kangwon.ac.kr)

13

운영체제 발전과정

시스템	입출력 장치	스케줄링	파일시스템	사용자 인터페이스	목표
단순일괄 처리	카드판독기 라인프린터 자기테이프	없음	없음	오프라인 (천공카드, 인쇄물)	효율성
스풀링	하드디스크 추가	없음	없음	위와 동일	효율성
멀티 프로그래밍	(메모리용량 증가)	FCFS	없음	위와 동일	효율성
시분할	터미널 추가 카드판독기 제외	라운드Robin	사용	온라인 (터미널)	효율성 편리성

2020-06-03

Yong-Seok Kim (yskim@kangwon.ac.kr)

14



컴퓨터 구조의 발전에 따른 기능

✓ 멀티프로세서 시스템을 위한 운영체제

- ✓ 멀티프로세서 (multi-processor) 시스템: 프로세스/스레드들을 각 CPU에 골고루 분배하는 기능
- ✓ 대규모 병렬처리 (MPP: massively parallel processing): 하나의 작업을 위해 여러 개의 CPU들을 끓음으로 배분하는 기능

✓ 분산처리 시스템을 위한 운영체제

- ✓ 네트워크 파일시스템: 다른 컴퓨터에 있는 파일들을 공유
- ✓ 클러스터링 (clustering) 시스템: 통신망으로 연결된 컴퓨터들간에 부하를 적절히 분산하여 실행

✓ 실시간 시스템을 위한 운영체제

- ✓ 실시간 시스템 (real-time system) / 내장형 시스템 (embedded system)
- ✓ 컴퓨터 시스템의 소형화/휴대형/실시간성 특성에 따라
- ✓ 운영체제 크기의 축소, 절전 기능, 실시간 스케줄링 등의 기능 필요

2020-06-03

Yong-Seok Kim (yskim@kangwon.ac.kr)

15