

흥달샘과 함께하는

—

정보처리기사 실기 최종정리 특강

[3과목 - 운영체제]

1억뷰 N잡

이 자료는 대한민국 저작권법의 보호를 받습니다.

작성된 모든 내용의 권리는 작성자에게 있으며, 작성자의 동의 없는 사용이 금지됩니다.

본 자료의 일부 혹은 전체 내용을 무단으로 복제/배포하거나 2차적 저작물로 재편집하는 경우,
5년 이하의 징역 또는 5천만 원 이하의 벌금과 민사상 손해배상을 청구합니다.

YouTube 흥달샘 (<https://bit.ly/3KtwdLG>)

E-Mail hungjik@naver.com

네이버 카페 흥달샘의 IT 이야기 (<https://cafe.naver.com/sosozl/>)

01 운영체제

Section 1. 운영체제 기초

1. 기억장치

(1) 기억장치의 개념

- 데이터, 프로그램, 연산의 중간 결과 등을 일시적 또는 영구적으로 저장하는 장치

(2) 기억장치의 종류

- 1) 레지스터
- 2) 캐시 메모리
- 3) 주기억장치
- 4) 보조기억장치
- 5) 연관 메모리

2. 시스템 소프트웨어

(1) 시스템 소프트웨어의 종류

- 로더 : 목적 프로그램을 메모리에 적재하고 배치 주소를 옮기는 프로그램
- 링커 : 목적 파일(Object File)을 실행 파일(Execute File)로 변환해 주는 프로그램
- 유틸리티
- 번역기(컴파일러, 어셈블러)
- 장치 드라이버
- 운영체제

(2) 시스템 소프트웨어의 구성

1) 제어 프로그램

- 감시 프로그램(Supervisor Program)
- 작업관리 프로그램(Job Control Program)
- 데이터 관리 프로그램(Data Control Program)

2) 처리 프로그램

- 서비스 프로그램(Service Program)
- 문제 프로그램(Problem Program)
- 언어 번역 프로그램(Language Translator Program)

3. 운영체제

(1) 운영체제의 기능

- 프로세스 관리
- 메모리 관리
- 파일 관리
- 입출력 관리
- 보조기억장치 관리

- 네트워킹
- 정보 보안 관리
- 명령해석

(2) 운영체제 운용 기법

- 일괄 처리 시스템(Batch Processing System)
- 실시간 처리 시스템(Real Time Processing)
- 다중 프로그래밍 시스템(Multi Programming)
- 시분할 시스템(Time Sharing)
- 다중 처리 시스템(Multi-Processing)
- 다중 모드 시스템(Multi-Mode)
- 분산 처리 시스템(Distribute Processing)

4. 운영체제의 종류

(1) 윈도우(Windows)

- MS-DOS의 멀티태스킹 기능과 GUI 환경을 제공하는 운영체제

(2) 리눅스(Linux)

- 1991년 리누스 토발즈에 의해 오픈소스로 개발된 유닉스 호환 OS

(3) 유닉스(Unix)

1) Unix 시스템의 구성

- 커널(Kernel), 셸(Shell), 유틸리티 프로그램(Utility Program)

2) Unix 파일 시스템

- 부트블록 : 부팅시 필요한 정보
- 슈퍼블록 : 전체 파일 시스템에 대한 정보
- I-node 블록 : 파일이나디렉터리에 대한 정보
- 데이터 블록 : 실제 파일에 대한 데이터 정보

3) 파일 디스크립터(FD, File Descriptor)

- 유닉스 시스템에서 프로세스가 파일들을 접근할 때 이용

4) POSIX(Portable Operating System Interface)

- 이식 가능한 운영체제 인터페이스

(4) MacOS

- 애플사가 개발한 유닉스 기반의 운영체제

Section 2. 메모리 관리

1. 기억장치 관리 전략

- (1) 반입(Fetch) 전략
 - 보조기억장치에 보관 중인 프로그램이나 데이터를 언제 주기억장치로 적재할 것인지를 결정하는 전략
 - 요구 반입, 예상 반입
- (2) 배치(Placement) 전략
 - 새로 반입되는 프로그램이나 데이터를 주기억장치의 어디에 위치시킬 것인지를 결정하는 전략
 - 최초 적합 (First Fit), 최적 적합 (Best Fit), 최악 적합 (Worst Fit)
- (3) 교체(Replacement) 전략
 - 이미 사용되고 있는 영역 중에서 어느 영역을 교체하여 사용할 것인지를 결정하는 전략
 - 종류 : FIFO, OPT, LRU, LFU, NUR, SCR 등

2. 단편화

- (1) 단편화의 개념
 - 주기억장치에 프로그램을 할당하고 반납하는 과정에서 발생하는 사용되지 않는 작은 조각 공간
- (2) 단편화의 종류
 - 내부 단편화 : 공간을 할당 후 사용되지 않고 남아있는 공간
 - 외부 단편화 : 프로그램이 할당될 수 없어 사용되지 않고 남아있는 공간
- (3) 단편화 해결 방법
 - 통합(Coalescing) : 인접해 있는 공간을 하나로 통합
 - 압축(Compaction) : 주기억장치 내 분산되어 있는 단편화 공간들을 통합
 - 재배치 기법(Relocation) : 프로그램의 주소를 새롭게 지정해주는 기법

Section 3. 가상기억장치

1. 가상기억장치

- (1) 가상기억장치의 개념
 - 보조기억장치(하드디스크)의 일부를 주기억장치처럼 사용하는 것
- (2) 블록 분할 방법
 - 1) 페이징(Paging) 기법
 - 같은 크기의 블록으로 편성하여 운용하는 기법(내부단편화 발생)
 - 페이지 크기별 비교

페이지 크기	기억장소 효율	단편화	입출력 시간	맵 테이블
클수록	감소	증가	감소	감소
작을수록	증가	감소	증가	증가

2) 세그멘테이션(Segmentation) 기법

- 서로 크기가 다른 세그먼트로 분할하고 메모리를 할당하는 기법(외부 단편화 발생)

2. 가상기억장치 기타 관리사항

(1) 페이지 부재

- 프로세스 실행 시 참조할 페이지가 주기억장치에 없는 현상

(2) 지역성(Locality)

- 프로세스가 실행되는 동안 주기억장치를 참조할 때 일부 페이지만 집중적으로 참조하는 성질
- 지역성의 종류

1) 시간 구역성(Temporal Locality) : Loop(반복), Stack(스택), 부 프로그램(Sub Routine) 등

2) 공간 구역성(Spatial Locality) : 배열순회, 순차적 코드 실행 등

(3) 워킹 셋(Working Set)

- 프로세스가 일정 시간 동안 자주 참조하는 페이지들의 집합

(4) 스래싱(Thrashing)

- 프로세스의 처리 시간보다 페이지 교체에 소요되는 시간이 더 많아지는 현상

3. 페이지 교체 알고리즘

(1) FIFO(First In First Out)

- 가장 먼저 메모리에 적재된 페이지를 먼저 교체하는 기법
- 프레임 개수를 늘리면 부재 발생이 감소해야 하나, 오히려 더 늘어나는 Belady's Anomaly 이상 현상 발생

(2) OPT(Optimal replacement, 최적 교체)

- 앞으로 가장 사용되지 않을 페이지를 교체

(3) LRU(Least Recently Used)

- 최근에 가장 오랫동안 사용되지 않은 페이지를 교체

(4) LFU(Least Frequently Used)

- 사용 빈도가 가장 적은 페이지를 교체

(5) NUR(Not Used Recently)

- 참조비트와 변형비트를 이용해서 페이지 교체

(6) SCR(Second Chance Replacement)

- FIFO 기법의 단점을 보완하는 기법

Section 4. 프로세스

1. 프로세스

(1) 프로세스의 개념

- 컴퓨터에서 연속적으로 실행되고 있는 컴퓨터 프로그램

(2) 스레드(Thread)

1) 스레드의 개념

- 프로세스 내에서 실행되는 흐름의 단위
- 프로그램은 하나 이상의 프로세스를 가지고 있고, 하나의 프로세스는 반드시 하나 이상의 스레드를 갖는다.

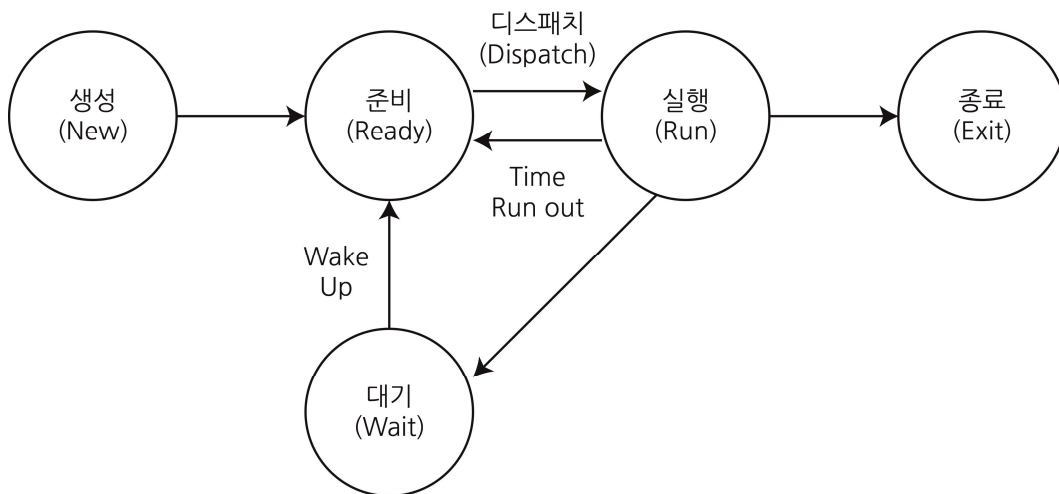
2) 스레드의 분류

- 사용자 수준의 스레드
- 커널 수준의 스레드

(3) 메모리상의 프로세스 영역

- 코드 영역 : 프로그램 코드 저장
- 데이터 영역 : 전역 변수, 정적 변수
- 스택 영역 : 지역 변수
- 힙 영역 : 동적할당

(4) 프로세스 상태 전이



(5) PCB(Process Control Block, 프로세스 제어 블록)

- 운영체제가 프로세스에 대한 정보를 저장해 놓는 공간

(6) 문맥 교환(Context Switching)

- 이전의 프로세스의 상태를 보관하고 또 다른 프로세스를 레지스터에 적재하는 과정

2. 프로세스 스케줄링

(1) 스케줄링의 목적

- 공평성, 효율성, 안정성, 반응 시간 보장, 무한 연기 방지

(2) 스케줄링 기법

1) 선점형 스케줄링(Preemptive)

- 다른 프로세스가 실행 중이더라도 운영체제가 CPU를 강제로 뺏을 수 있는 방식

- 종류 : Round Robin, SRT, 다단계 큐(MLQ, Multi-Level Queue), 다단계 피드백 큐(MLFQ, Multi-Level Feedback Queue) 등
- 2) 비선점형 스케줄링(Non-Preemptive)
- 프로세스가 CPU를 점유하고 있다면 이를 빼앗을 수 없는 방식
 - 종류 : FCFS, SJF, HRN, 우선순위, 기한부 등
- 3) 기아현상과 에이징 기법
- 기아현상(Starvation) : 우선순위가 낮은 프로세스가 무한정 기다리는 현상(SJF, 우선순위, SRT, MLQ)
 - 에이징 기법(Aging) : 기아현상을 해결하기 위한 기법(HRN, MLFQ)

3. 스케줄링 알고리즘

(1) 선점형 기법

- 1) Round Robin
- 시간단위(Time Quantum/Slice)를 정해서 프로세스를 순서대로 CPU를 할당하는 방식
- 2) SRT(Shortest Remaining Time)
- CPU 점유 시간이 가장 짧은 프로세스에 CPU를 먼저 할당하는 방식
- 3) 다단계 큐(MLQ, Multi-Level Queue)
- 그룹에 따라 각기 다른 준비 상태 큐를 사용하는 기법
- 4) 다단계 피드백 큐(MLFQ, Multi-Level Feedback Queue)
- 프로세스 생성 시 가장 높은 우선순위 준비 큐에 등록되며, 등록된 프로세스는 FCFS 순서로 CPU를 할당받아 실행되고, 할당된 시간이 끝나면 다음 단계의 준비 큐로 이동

(2) 비 선점형 기법

- 1) FCFS(First Come First Serve)
- 먼저 도착한 프로세스를 먼저 처리하는 스케줄링 알고리즘
- 2) SJF(Shortest Job First)
- 실행시간이 가장 짧은 프로세스에게 CPU를 할당하는 방식
- 3) HRN(Highest Response ratio Next)
- SJF 기법에서 비교적 실행시간이 긴 프로세스가 가질 수 있는 불리함을 보완한 스케줄링 방식
 - $\text{우선순위} = (\text{대기시간} + \text{실행시간}) / \text{실행시간}$
- 4) 우선순위(Priority)
- 프로세스마다 우선순위를 부여하여 높은 우선순위를 가진 프로세스에게 먼저 자원을 할당
- 5) 기한부(Deadline)
- 프로세스에게 일정한 시간을 주어 그 시간 안에 완료하도록 하는 기법

Section 5. 병행 프로세스와 교착상태

1. 병행 프로세스

(1) 병행 프로세스의 개념

- 두 개 이상의 프로세스들이 동시에 존재하며 실행상태에 있는 것

(2) 문제점과 해결책

1) 문제점

- 동시에 2개 이상의 프로세스를 병행 처리하면 한정된 자원(CPU, 메모리, 디스크, I/O 장치 등)에 대한 사용 순서 등 여러 가지 문제가 발생

2) 문제 해결책

- 임계구역, 상호배제 기법, 동기화 기법

2. 병행 프로세스 문제 해결책

(1) 임계구역(Critical Section)

- 공유 자원에 대해서 한 순간에는 반드시 하나의 프로세스만 사용되도록 지정한 영역

(2) 상호 배제(Mutual Exclusion)

- 하나의 프로세스가 공유 메모리 혹은 공유 파일을 사용하고 있을 때 다른 프로세스들이 사용하지 못하도록 배제시키는 제어 기법

- 상호 배제 기법

- ① 데커의 알고리즘(Dekker's Algorithm)
- ② 피터슨의 알고리즘(Peterson's Algorithm)
- ③ 다익스트라 알고리즘(Dijkstra Algorithm)
- ④ 램포트의 베어커리 알고리즘(Lamport's Bakery Algorithm)

(3) 동기화 기법

- 스레드들에게 하나의 자원에 대한 처리 권한을 주거나 순서를 조정해주는 기법
- 세마포어(Semaphore), 모니터(Monitor)

3. 교착상태(Dead Lock)

(1) 교착상태의 개념

- 상호 배제에 의해 나타나는 문제점으로, 둘 이상의 프로세스들이 자원을 점유한 상태에서 서로 다른 프로세스가 점유하고 있는 자원을 요구하며 무한정 기다리는 현상

(2) 교착상태 발생 조건

- 상호배제(Mutual Exclusion) : 공유 자원은 하나의 프로세스만이 사용
- 점유와 대기(Hold & Wait) : 자원을 점유 하면서 다른 프로세스의 자원을 추가로 요구하며 대기
- 비선점(Nonpreemption) : 강제로 빼앗을 수 없음
- 환형대기(Circular Wait) : 각 프로세스가 순차적으로 다음 프로세스가 요구하고 있는자원을 가지고 있는 상태

(3) 교착상태 해결 방법

- 예방 기법(Prevention)
- 회피 기법(Avoidance) : 은행원 알고리즘 사용
- 발견 기법(Detection)
- 회복 기법(Recovery)

Section 6. 디스크 스케줄링(Disk Scheduling)

1. 디스크 스케줄링

(1) 디스크 스케줄링 종류

1) FCFS 스케줄링(First Come First Service)

- 요청이 들어온 순서대로 처리

2) SSTF(Shortest Seek Time First)

- 현재 헤드에서 가장 가까운 트랙의 요청을 먼저 처리한다.

3) SCAN

- 헤드의 진행방향에 있는 요청을 처리하고, 다시 반대방향으로 틀어 반대방향에 있는 요청들을 처리한다.

4) C-SCAN

- 항상 한쪽 방향에서 반대방향으로 진행하며 트랙의 요청을 처리한다.

5) LOOK

- SCAN 기법을 기초로 사용하며, 진행 방향의 마지막 요청을 처리한 후 반대방향으로 처리하는 기법

6) C-LOOK

- C-SCAN 기법을 기초로 사용하며, 진행 방향의 마지막 요청을 처리한 후 반대방향으로 처리하는 기법

7) N-STEP SCAN

- SCAN 기법을 기초, 처리하는 과정 중에 요청이 들어오면 반대 방향으로 진행할 때 서비스

8) 에션바흐(Eschenbach)기법

- 부하가 매우 큰 항공 예약 시스템을 위해 개발

Section 7. 환경변수와 로그 파일

1. 환경변수

(1) 환경변수의 개념

- 프로세스가 컴퓨터에서 동작하는 방식에 영향을 미치는 동적인 값들의 모임

(2) UNIX/Linux 환경변수

- env, set, printenv 명령어들을 사용하여 환경변수와 그에 따른 모든 값을 볼 수 있다.
- export 명령을 이용하여 사용자 환경변수를 전역변수로 설정할 수 있다.

(3) Windows 환경변수

- 제어판 > 시스템 및 보안 > 시스템 > 고급 시스템 설정 > 환경변수 존재
- 커맨드 창에서 set 명령으로 확인

2. 로그 파일

(1) 로그의 개념

- 시스템의 모든 기록을 담고 있는 데이터

(2) 리눅스 로그 종류

종류	설명
messages	시스템 로그 파일
secure	보안인증에 관한 메시지 로그파일
maillog	메일 로그 파일
xferlog	ftp 로그파일
dmesg	부팅 시의 시스템 로그
wtmp	시스템에 로그인 기록이 저장되는 파일(전체 로그인 기록)
utmp	시스템에 로그인 기록이 저장되는 파일(현재 로그인 사용자에 대한 기록)
btmp	로그인 실패 정보 기록
lastlog	각 계정들의 가장 최근 로그인 기록

Section 8. 스토리지

1. 스토리지(Storage)

(1) 스토리지 종류

- DAS(Direct Attached Storage)
- NAS(Network Attached Storage)
- SAN(Storage Area Network)

(2) RAID(Redundant Array of Inexpensive Disks)

1) RAID 개념

- 복수의 하드디스크를 하나의 드라이브와 같이 인식하고 표기한다.

2) RAID 구성

① 스트라이핑(Stripping)

- 논리적으로 연속된 데이터들이 물리적으로 여러 개의 디스크에 라운드 로빈 방식으로 저장되는 형태

② 미러링(Mirroring)

- 데이터를 그대로 복제하는 것으로 신뢰성 및 가용성 확보를 위해 사용됨

3) RAID 형태

- RAID-0 : 스트라이핑
- RAID-1 : 미러링
- RAID-2 : 해밍코드 오류정정
- RAID-3 : 하나의 디스크는 패리티 정보, 나머지는 데이터 저장
- RAID-4 : RAID-3 과 같은 형태, 블록 단위로 분산 저장
- RAID-5 : 각각의 디스크에 패리티 정보 저장
- RAID-6 : 하나의 패리티를 두 개의 디스크에 분산 저장