

컴퓨터의 구성 및 동작, 운영체제

논리연산

	비트논리연산	부울논리연산(참과 거짓,T/F)
• AND(논리곱)	1101&1010	a && b
• OR(논리합)	1101 1010	a b
• NOT(단항연산자)	~1100	~a
• XOR(베타논리곱)	1010^1100	
• NOR		
• NAND		

2의 보수

- 2의 보수는 어떤 수를 커다란 2의 제곱수에서 빼서 얻은 이진수이다.
2의 보수는 대부분의 산술연산에서 원래 숫자의 음수처럼 취급된다.

Overflow

- 메모리 용량을 넘어선 값이 들어가 생기는 오류

Underflow

- 부동소수점 연산에서 지수부가 타입의 한계를 넘어 작아지면 0에 가까워 지다가 결국 0이 되어버리는 현상을 의미한다.

운영체제

- UNIX
BSD
MAC OS X
- LINUX (UNIX Like OS)
Android

Ubuntu
CentOS
Fedora

- WINDOWS
NT

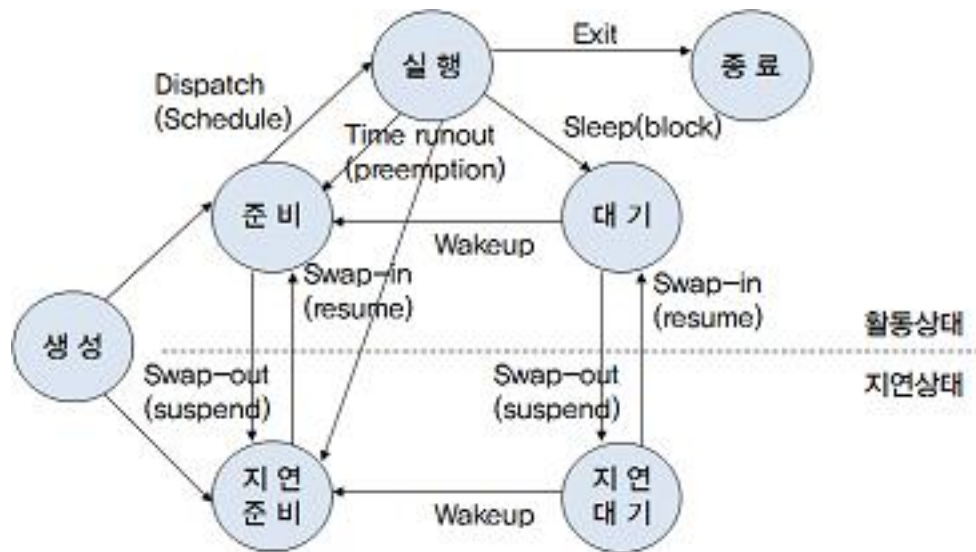
운영체제가 왜 필요할까?

- 하드웨어를 잘 조작하기 위해
- 사용자의 편리를 위해
- 보편타당화
- 범용적 사용을 위해
- 컴퓨터를 사용하는 사용자의 목적이 다양해짐에 따라 하드웨어가 다양해졌고 이를 제어하기 위한 응용프로그램이 필요하기 시작했다.

운영체제 커널이 하는 일

- 보안(시스템 하드웨어 관리)
프로그램의 오류나 잘못된 자원사용을 감시 입출력 장치 등의 자원에 대한 연산과 제어를 관리
- 추상화(가상 시스템 서비스 제공)
사용자에게 컴퓨터의 프로그램을 쉽고 효율적으로 실행 할 수 있는 환경을 제공
- 자원관리
컴퓨터 시스템 하드웨어 및 소프트웨어 자원을 여러 사용자 간의(응용프로그램) 효율적 할당, 관리, 보호

프로세스 상태 전이도



- **1.생성**
프로세스 생성(주기억장치에 적재)
- **2.준비**
중앙처리장치에 의해 프로세스가 실행되기를 기다린다.
- **3.실행**
중앙처리장치에 의해 프로세스가 실행된다.
- **4-1.종료**
프로세스가 종료된다.
- **4-2.대기 -> 준비**
어떤사건이 일어나기를 기다린다.

자원관리 - 프로세스 스케줄링

- 프로세스마다 얼마만큼의 자원을 사용해야 하는 결정 하는것
- **비선점 스케줄링(Non-Preemptive)**
할당된 CPU 를 다른 프로세스가 강제로 빼앗아 사용할수 없는 기법
한번 CPU 를 할당 받으면 작업이 완료될때 까지 CPU 를 사용
문제점 : 짧은 작업(중요한 작업)이 긴 작업(중요하지 않은 작업)을 기다리는 경우 발생
종류 : FCFS(FIFO), SJF, 우선순위, HRN, 기한부(Deadline)

- **FCFS(First Come First Served)**

준비 상태 큐에 도착한 순서에 따라 차례대로 중앙처리장치에 할당 한다.

- **SJF(Shortest Job First)**

실행 시간이 짧을 수록 Process 에 먼저 중앙처리장치에 할당 한다.

평균 대기시간이 가장 적은 알고리즘

실행시간이 긴 프로세스는 뒤로 밀림

예측 불가 및 비용 발생 (정확한 처리시간을 알수가 없을 뿐만아니라 측정하는데에 비용발생이 들기 때문이다)

- **기한부(Deadline)**

일정시간동안 프로세스 완료하는 기법

제한된 시간 안에 완료되지 않을 경우 제거 되거나 처음부터 다시 실행해야함

여러 프로세스들이 동시에 실행되면 스케줄링이 복잡해지며, 프로세스 실행 시 집중적으로 요구되는 자원관리에 오버헤드가 발생한다.

- **Priority Based Scheduling(우선순위)**

프로세스마다 우선순위 부여

우선순위가 동일한 경우 FCFS 기법으로 할당

가장 낮은 순위를 부여받은 프로세스는 무한 연기 또는 기아상태가 발생할 수 있다.

- **선점 스케줄링(Preemptive)**

우선순위가 높은 다른 프로세스가 CPU 를 강제로 빼앗아 사용할 수 있는 기법

빠른 응답 시간을 요구하는 대화식 시분할 시스템(Time Sharing System)에 사용

문제점 : 많은 오버헤드 초래

인터럽트용 타이머 클럭 필요(프로세스가 자원을 독점하는것을 방지)

종류 : RR(Round Robin), SRT, 선점 우선순위, 다단계 큐(MQ), 다단계

피드백큐(MFQ)

- **Round Robin Scheduling**

각 프로세스는 시간 할당량 동안만 실행한 후 완료되지 않으면 다음 프로세스에게

CPU 를 넘겨주고 준비상태 큐의 가장 뒤로 배치

시분할 시스템을 위해 고안된 방식

FCFS 기법 변형

할당된 시간이 클수록 FCFS 와 비슷

할당시간이 작을 수록 문맥교환과 오버헤드가 자주발생

- **SRT(Shortest Remaining Time)**

SJF 기법을 변형, 선점 SJF 라고도 한다.

실행중인 프로세스의 남은 시간과 준비상태 큐에 새로 도착한 프로세스의 실행 시간을 비교하여 짧은 실행 시간을 요구하는 프로세스에게 CPU 를 할당.

준비상태 큐에 있는 프로세스의 실행 기간 추적으로 오버헤드 증가

- **Multi Queue Scheduling**

프로세스를 특정 그룹으로 분류 할 수 있는 경우 그룹에 따라

각기 다른 준비 단계 큐 사용

준비 상태 큐마다 다른 스케줄링 기법 사용가능

다른 준비상태 큐로 이동불가

하위 단계 준비 큐에 있는 프로세스를 실행하는 도중이라도 상위 단계 준비상태 큐에 프로세스가 들어오면 상위 단계 프로세스에게 CPU 를 할당

자원관리 - 주기억 장치 관리

- **단순관리**

가상메모리: 보조기억장치를 주기억장치 처럼 활용

- **파일관리**

응용프로그램 <-> 운영체제 <-> 보조기억장치

(파일 입/출력 요청) (파일 입/출력 처리)

조각모음

- 파편화 되있는 데이터를 한곳에 모아 액세스 횟수를 줄인다.

메타데이터

- 컴퓨터 용어에서, 접두어 메타(Meta)는 일반적으로 "~에 관한"(about)이라는 의미로 사용된다. 따라서 메타언어는 다른 데이터를 기술하기 위해 사용되는 언어이며, 메타데이터는 다른 데이터를 기술하기 위해 사용하는 데이터라고 할수 있다.

RISC(Reduced Instruction Set Computer)

- CPU 명령어의 개수를 줄여 하드웨어 구조를 좀 더 간단하게 만드는 방식

CISC (Complex Instruction Set Computer)

- 복잡한 명령어 집합을 갖는 CPU 아키텍처

ARM(Acorn RISC Machine)

- 임베디드 기기에 많이 사용되는 RISC 프로세서

과제 내용

- 액티브 X 가 무일까?
- 국내 웹에 액티브 X 가 정착되게 된 이유 및 배경
- 기술적 부채(Technical Debt)에 대해 알아보기

학습링크

ALU

<http://goo.gl/ROO7KS>

가산기

<http://goo.gl/T2mYkl>

누산기

<http://goo.gl/5aQf2L>

운영체제

<http://goo.gl/4UalWu>, <http://goo.gl/njn9fl>, <http://goo.gl/kl3DP3>

스케줄링

<http://goo.gl/Wi9lri>, <http://goo.gl/7HwAe0>

파일시스템

<http://goo.gl/VuPAHg>

커널

<http://goo.gl/CG9zir>

리눅스 토발즈

<http://goo.gl/kQSSsR>, <http://goo.gl/xQ6JPi>

자료구조

<http://goo.gl/f8O7Vo>, <http://goo.gl/H1CKb0>

알고리즘

<http://goo.gl/GRz6tA>, <http://goo.gl/qdkZIF>