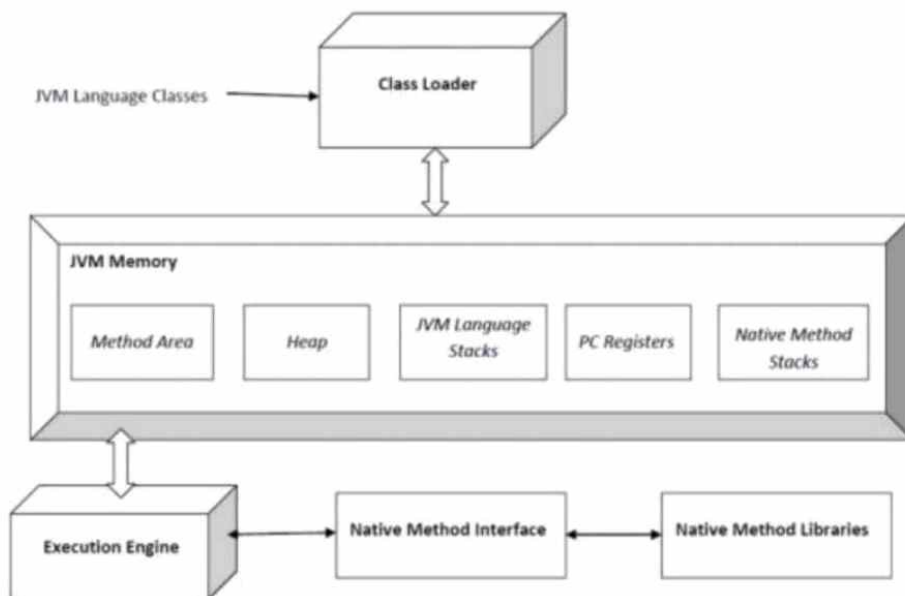


Computer Science 정리

2. 컴퓨터 시스템

1. 가상 머신

- 운영체제와 응용 프로그램 사이에서 작동하는 프로그램
- 가상머신을 설치하면 응용프로그램이 모두 동일한 환경에서 작동하는 것처럼 보임
- 자바는 UNIX와 Window에서 작동하는 다양한 가상머신을 만들어서 배포
- Java Virtual Machine (JVM) , 자바 가상 머신



자바 가상 머신 사양의 자바 SE7 에디션에 기반을 둔 자바 가상 머신(JVM) 아키텍처의 개요도.
*출처: 위키백과

2. 스레드(Thread)

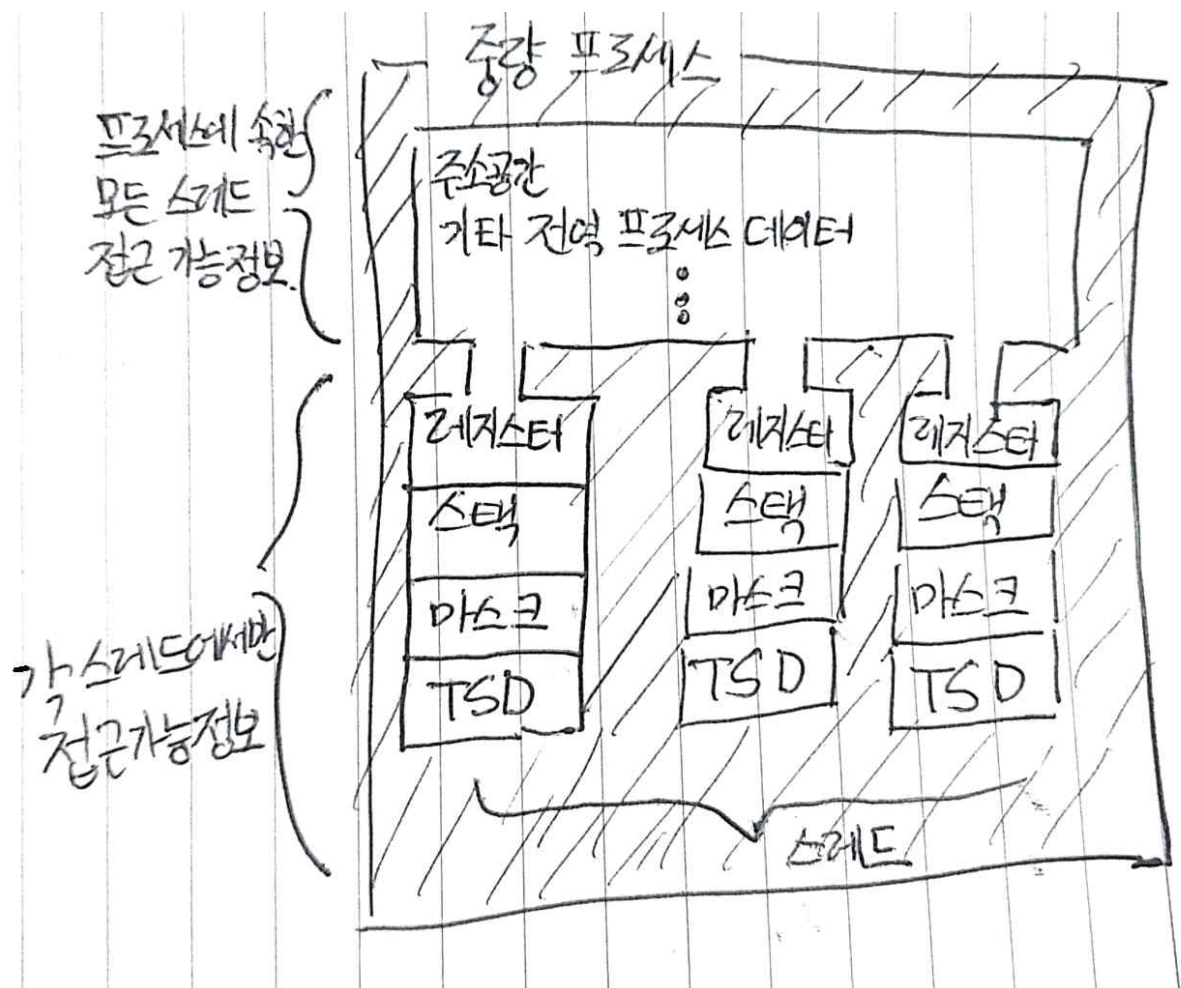
- 스레드의 정의

- 경량 프로세스(LWP)
- 명령어들의 한 줄기 혹은 프로그램 제어 흐름
- 프로세스에 있는 스레드들은 병행으로 실행되면서 공통의 목표위해 협력
- 레지스터, 스택, 신호 마스크처럼 스레드마다 있는 특정 데이터들은 각 스레드에 국한 되어있음
- 프로세스에 속한 주소 공간은 해당 프로세스에 속한 모든 스레드에 공통

* 특정 플랫폼에서 스레드를 구현한 방식에 따라

스레드를 운영체제 혹은 스레드를 생성한 사용자 프로그램에서 관리

Ex) Win32 threads , C - threads , P threads



- 멀티 스레딩을 선호하게 된 요인

소프트웨어 설계	- 병렬 작업 처리를 더 간단하게 표현 가능
성능	- 멀티 스레드 응용 프로그램이 작업 완료를 하는데 필요한 시간이 크게 감소
협력	- 한 프로세스에 속한 스레드들은 주소 공간을 공유하기 때문에 이를 통해 서로 통신 가능

- 스레드와 프로세스는 공통된 많은 연산들을 소유
ex) create , exit, resume, suspend
- 스레드의 생성은 운영체제가 부모 프로세스와 스레드 간에 공유되는 자원을 초기화 할 필요가 없음!
- 프로세스의 생성과 종료에 비교하여 스레드의 생성과 종료에 대한 오버헤드를 감소시키기 위함.

3. 교착상태(deadlock)

- 교착상태의 정의
 - 2개 이상의 프로세스가 다른 프로세스의 작업이 끝나기만 기다리며 작업을 더 이상 진행하지 못하는 상태(서로 작업이 안끝남)
- 아사 상태와의 차이점
 - 아사 현상 : 운영체제가 잘못된 정책을 사용하여 특정 프로세스의 작업이 지연되는 문제
 - 교착 상태 : 여러 프로세스가 작업을 진행하다보니 자연 발생적으로 일어나는 문제
- 교착상태의 필요조건
 - 다음 4가지 조건이 모두 발생해야만 교착상태가 발생(필요 조건)

상호 배제 (mutual exclusion)	한 프로세스가 사용하는 자원은 다른 프로세스와 공유할 수 없는 배타적인 자원이어야 한다
비선점 (non-preemptive)	한 프로세스가 사용중인 자원은 중간에 다른 프로세스가 빼앗을 수 없는 비선점 자원이어야 한다
점유와 대기 (hold and wait)	프로세스가 어떤 자원을 할당받은 상태에서 다른 자원을 기다리는 상황이어야 한다
원형 대기 (circular wait)	점유/대기를 하는 프로세스 간의 관계가 원을 이루어야 한다.

- 교착 상태의 해결 방법

교착 상태 예방 (prevention)	교착 상태를 유발하는 네 가지 조건이 발생하지 않도록 무력화하는 방식으로 교착상태를 예방
교착 상태 회피 (avoidance)	교착상태가 발생하지 않도록 자원 할당량을 조절하여 교착 상태를 회피하는 방식
교착 상태 검출/회복 (detection/recovery)	교착 상태 검출 : 어떤 제약을 가하지 않고 자원 할당 그래프를 모니터링 하면서 교착 상태가 발생 하는지 살펴보는 방식 만약 교착상태가 발생한다면 교착 회복 단계를 진행.

- 교착 상태 예방 (prevention)

- 원형 대기 예방 : 점유와 대기를 하는 프로세스들이 원형을 이루지 못하도록 막는 방법. 모든 자원에 숫자를 부여하고 숫자가 큰 방향으로만 자원을 할당.
- 교착 상태 예방 정리 : 교착상태를 유발하는 조건이 일어나지 않도록 제약을 가하는 방법.
자원을 보호하기 위해 상호배제와 비선점을 예방하기는 어려움.
점유와 대기 , 원형 대기는 프로세스 작업 방식을 제한하고 자원을 낭비하기 때문에 사용 X

- 교착 상태 회피 (avoidance)

- 프로세스에 자원을 할당할 때 어느 수준 이상의 자원을 나누어 주면 교착상태가 발생하는지 파악하여 자원할당량을 조절하는 방법.
- 교착 상태가 발생하지 않는 범위 내에서만 자원을 할당 , 교착 상태가 발생하는 범위에 있으면 프로세스를 대기 시킴
- 즉, 할당되는 자원의 수를 조절하여 교착 상태를 피한다.

- 교착 상태 검출

- 운영체제가 프로세스의 작업을 관찰 하면서 교착 상태 발생 여부를 계속하여 주시하는 방식

- 교착 상태가 발견되면 이를 해결하기 위해 교착 상태 회복 단계를 진행

- 교착 상태 회복

- 교착 상태가 검출된 후 교착상태를 푸는 후속 작업을 진행 하는 것

- 교착 상태 회복 단계에서는 교착 상태를 유발한 프로세스 강제 종료