GC – 가비지 컬렉터

1. 가비지 컬렉터란?

가비지 컬렉터는 가비지, 흔히 ‘정리되지 않은 메모리’, ‘유효하지 않은 메모리 주소’ 를 정리하여 다른 용도로 재활용할 수 있도록 도와주는 기능을 말한다. 메모리 관리 기법의 하나로 프로그램이 동적으로 할당했던 메모리 영역 중 필요 없게 된 영역을 해제하는 기능이다.

1. 장단점

GC를 사용하면 이미 해제된 메모리에 접근하는 것을 방지할 수 있으며 해제된 메모리를 다시 해제하는 것 또한 방지할 수 있다. 그리고 사용자가 미처 동적생성을 한 이후 미처 해제 시키지 못한 경우 발생하는 메모리 누수가 생기지 않는다.

하지만 장점만 있는 것은 아니다. 먼저 어떤 메모리를 해제할 지 결정하는 데 비용이 발생하며 GC가 작동하는 타이밍이나 점유하는 시간을 예측하기 힘들다. 할당된 메모리가 해제되는 시점을 알기 힘들다는 점도 있다..

1. 작동 원리

GC가 특정 조건을 만족하여 작동할 때는 현재 수행중인 쓰레드를 모두 중단하고 GC 스레드를 활성화한다. 이처럼 수행중인 모든 쓰레드를 중단하는 것을 Stop-The-World라고 한다. GC 쓰레드는 힙에서 사용 중인 개체 참조 그래프를 생성하고 사용 중인 객체의 위치를 재조정함으로써 사용하지 않는 객체들을 모두 힙에서 제거하는 방법을 사용한다.

객체 참조 그래프를 생성하기 위해 루트 참조를 필요로 하는데 루트 참조는 각 쓰레드를 수행중인 메서드의 지역 변수, CPU 레지스터 변수가 가지고 있는 참조, 사용 중인 각 클래스의 정적 필드, 전역 변수가 해당된다.

객체 참조 그래프를 생성하면 그래프에 포함되지 않은 모든 객체가 GC의 대상이 된다. GC를 실행하 때 그래프 상의 객체들을 힙 상에서 재배치하고 메모리 할당 포인터를 감소시키는 메모리 컴팩션을 수행하기도 한다.