복습 메모 및 요약

알고리즘 – 어떠한 문제를 해결하기 위해 정해진 일련의 절차나 방법을 공식화한 형태로 표현한것 실행하기 위한 단계적 절차를 의미.

알고리즘의 조건

* 입력
* 출력
* 명확성
* 유한성
* 효과성

알고리즘 표현 방법

* 자연어 : 한국어 불어 등등 각 나라 언어를 뜻함
* 의사코드 : 자연어를 프로그래밍 언어처럼 사용
* 순서도 : 도형을 이용한 표현
* 프로그래밍 언어 : c나 java같은 프로그래밍 언어

알고리즘 성능 분석

공간 복잡도 ( 메모리 공간이라고 생각하면 됨 )

* 프로그램을 실행시킨 후 완료하는 데 필요로 하는 자원 공간의 양을 말함.
* 공간복잡도 = 고정공간(변수, 클래스, 배열 등등) + 가변공간(동적 메모리 할당 (거기에 자료넣고 연산하고…))
* S(P) = c+ SP(n)

시간 복잡도( 일반적으로 얘를 많이 씀 )

* 알고리즘 프로그램으로 실행하여 완료하기까지의 총 소요시간
* 시간 복잡도 = 컴파일 시간 + 실행 시간

1. 컴파일 시간 : 프로그램 마다 거의 고정적인 시간 소요
2. 실행 시간 : 컴퓨터의 성능에 따라 달라질 수 있으므로 실제 실행시간보다 명령문의 실행 빈도수에 따라 계산. <- 이거 가지고 시간 복잡도를 측정함 (명령문을 몇번 실행한 프로그램, 알고리즘인가)

시간 복잡도 예시

스크린샷이(가) 표시된 사진

자동 생성된 설명1과 n의 의미를 정확히 알아야함

case 1번은 여러개의 명령이 한번 실행된 것 이지만 만약에 다른문제에는 이러한 똑같은 연산 명령이 한번 두번….천번 있어도 그걸 대표하는게 1번임 ( 수행이 한번 이루어 진다 라고 생각하면 됨) O(1) 이라고 함

1은 여러가지 숫자를 대표하는 숫자

n은 반복의 수를 보통 표현함 반복문을 이렇게 하면됨

하나의 반복문이 있으면 n 이고

하나의 반복문안에 반복문이 있으면 n^2임

case 2번(반복문) 은 n으로 표기함 여기서 n은 아주 큰 정수라고 생각함 O(n) 이라고 함

4종류의 분석

분석방법은 4가지가 있음

* 최선경우
* 최악경우
* 평균시간
* 상각분석

여기서 최악경우로 많이 따짐

알고리즘 수행 시간의 점근 표기법

표기법에는 대표적으로 3가지가 있음

* 빅 오 표기법
* 빅 오메가 표기법
* 세타 표기법

우리는 빅 오 표기법에 초점을 맞춰야함

스크린샷이(가) 표시된 사진

자동 생성된 설명f(n) 위에서 잠시 이야기 해준 n으로 이루어진 다항식임 ( 반복문에 관련되어있는거 )

이게 다항식이 하나 존재함

for 문이 몇개있네 라는식으로 다항식을 구해서 그걸 n의 다항식으로 표현함

결국 cg(N)을 구해야함

텍스트, 지도이(가) 표시된 사진

자동 생성된 설명

스크린샷이(가) 표시된 사진

자동 생성된 설명

2N^2는 for안에 for가 2개 있는거고

3N은 for문이 3개 있는거고

5는 일반 연산이 5개 있는거임

빅오 표기법 가장 간단한 방법이

다항식에서 최고 차수 항만을 취한 뒤 그 항의 계수를 제거하는 것

개체, 시계, 측정기이(가) 표시된 사진

자동 생성된 설명

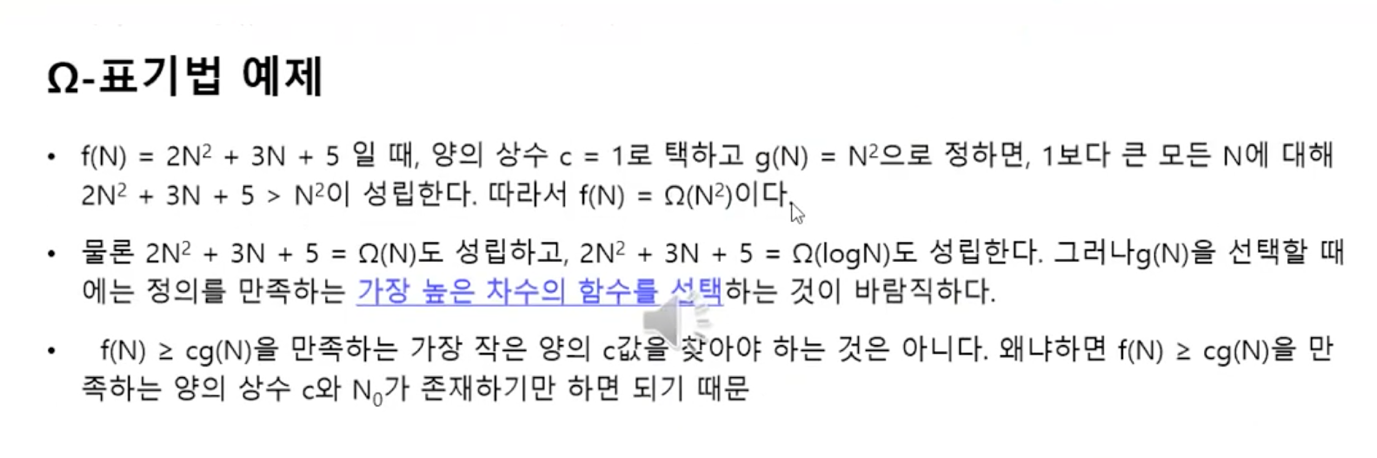
스크린샷이(가) 표시된 사진

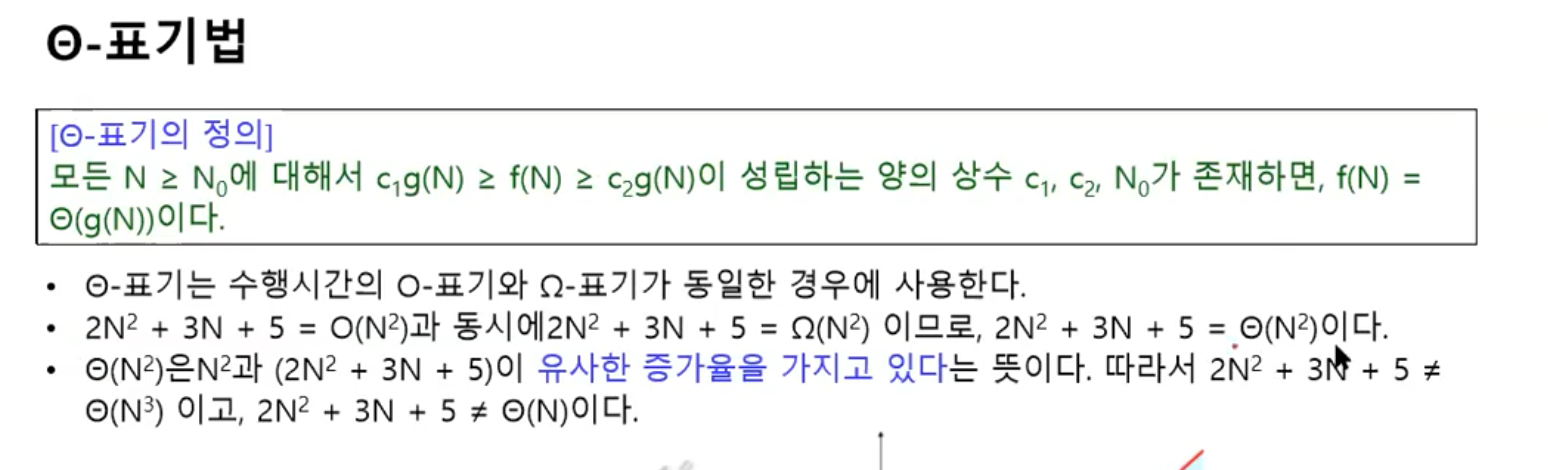
자동 생성된 설명

(이것도 똑같음)

스크린샷이(가) 표시된 사진

자동 생성된 설명





위에 방법도 있지만 우리는 빅 오 표기법을 가아아아아아아장 많이 사용함



O(1) 은 기본적인 연산문 한개든 백개든 1로 표기(위에서 배움)

O(N) 은 1에서 n 까지 1씩 증가시키는거



O(logN)은 O(N)에서 반으로 자르고 자르고 자르고… 2의 마이너스 승으로 쭉 감



O(N^2)는 for안에 for 있는거

O(N^3)은 for for for

O(2^N)은 일반적으로 재귀 함수

