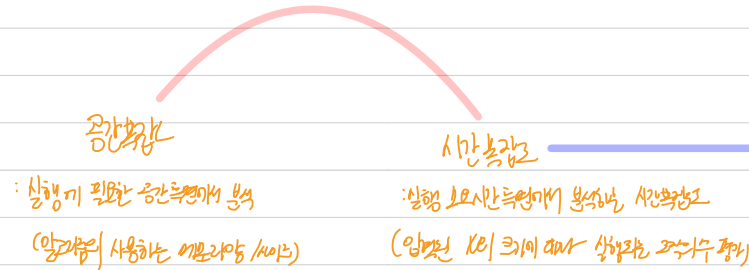


알고리즘: 어떤 문제나 목적을 달성하기 위해 가져야 하는 여러 과정

→ 알고리즘의 실행시간은 컴퓨터가 알고리즘을 실행하는 횟수이다



※ 시간 복잡도

① 최선: 빅오메가  $f(n)$ 이  $g(n)$ 보다 항상 작거나  $f(n)$ 이  $g(n)$ 보다 조금 증가

(최고 비용 시간 절감)

② 최악: 빅오  $f(n)$ 이  $g(n)$ 에 비례하거나  $g(n)$ 을 안 넘음

(시간이 최대 이만큼 걸림)

③ 평균: 세타  $f(n)$ 이  $g(n)$ 에 대하여 양쪽으로 제곱되고  $f(n)$ 이  $g(n)$ 과 비례

(시간이 평균적으로 이만큼 걸림)

효율성 ↑ 그래프 ↑ = 비용 ↓

→ 오메가는 최악의 시간인 양쪽 양자에 양자 줄 내림

세타는 평균 분석이 가장 잘 → 그래서 '빅오 사용'

비율 표기법

$O$  (증가량)

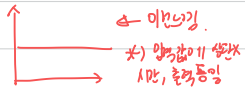
→ 입력 데이터의 크기 증가에 따라 증가하는 속도를 나타냄

시간 복잡도 ↑

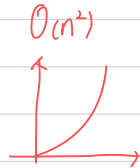
$$O(1) < O(\log n) < O(n) < O(n \log n) < O(n^2) < O(2^n) < O(n!)$$

→ 변수의 수가 증가하면 처리량을 제외한 모든 것의 속도가 = 효율적인 연산 처리.

$O(1)$ : 상수 시간



← 메모리  
\*) 양자값이 11 줄만  
시간, 공간 등임



\*) 여러 제곱의 제곱 증가  
e.g. 삼각/네모 패턴

$O(\log n)$



\*) 양자의 양자는 1/2로 줄임.

$O(n^3)$



Tip: 양자의 양자가  $n^3$  중 (증가량)

$O(n)$



\*) 양자 반올림  
양자가 1/2에 양자 등가.



\*) 양자면 (증가) 양자면  
양자 (가장) X 양자 (가장) 양자  
가장 양자