

Algorithm Study

복잡도 계산

2022.09.09

방지호



목차 Contents

- (1) 시간 복잡도의 기본 개념
- (2) Big-O 표기법
- (3) 문제집

(1) 시간 복잡도의 기본 개념

동일한 작업을 수행하는 알고리즘이 둘 이상 있을 때
어떤 방식으로 가장 좋은 알고리즘을 선택할 수 있을까?

Objective measures for efficiency

- ① Execution time
- ② Number of statements
- ③ Number of fundamental operations
(Expensive and/or frequently occurring operations)

(1) 시간 복잡도의 기본 개념

Ex) 백준 코딩의 채점 방식

제출 번호	아이디	문제	결과	메모리	시간	언어	코드 길이
48916952	gs5010	2 1182	맞았습니다!!	30840 KB	464 ms	Python 3 / 수정	360 B
48914196	gs5010	1 6588	맞았습니다!!	38892 KB	3308 ms	Python 3 / 수정	870 B
48912826	gs5010	1 11653	맞았습니다!!	30840 KB	1512 ms	Python 3 / 수정	116 B
48912746	gs5010	1 11653	맞았습니다!!	30840 KB	1072 ms	Python 3 / 수정	144 B
48912459	gs5010	1 11653	맞았습니다!!	30840 KB	1020 ms	Python 3 / 수정	261 B
48911959	gs5010	2 9020	맞았습니다!!	30840 KB	296 ms	Python 3 / 수정	836 B
48909305	gs5010	1 2609	맞았습니다!!	30840 KB	68 ms	Python 3 / 수정	452 B
48908404	gs5010	3 15649	맞았습니다!!	30840 KB	208 ms	Python 3 / 수정	594 B
48878140	gs5010	3 4948	맞았습니다!!	32768 KB	760 ms	Python 3 / 수정	1068 B
48875750	gs5010	1 2748	맞았습니다!!	30840 KB	68 ms	Python 3 / 수정	281 B
48867485	gs5010	5 1978	맞았습니다!!	30840 KB	72 ms	Python 3 / 수정	752 B
48866089	gs5010	3 1929	맞았습니다!!	38512 KB	888 ms	Python 3 / 수정	554 B
48865211	gs5010	5 8393	맞았습니다!!	30840 KB	68 ms	Python 3 / 수정	74 B
48763374	gs5010	3 10992	맞았습니다!!	30840 KB	68 ms	Python 3 / 수정	209 B

Execution time (실행 시간)

Number of statements

같은 문제를 풀더라도
코드 길이와 실행시간이
모두 다를 수 있음!

(※ 간단한 코드로 가장
효율적으로 동작하는
알고리즘을 구상하는
것이 중요함!!)

(2) Big-O 표기법

Big-O Notation

정의 : 특정 값이나 무한대로 향하는 경향이 있을 때 함수의 제한 동작을 설명하는 수학적 표기법

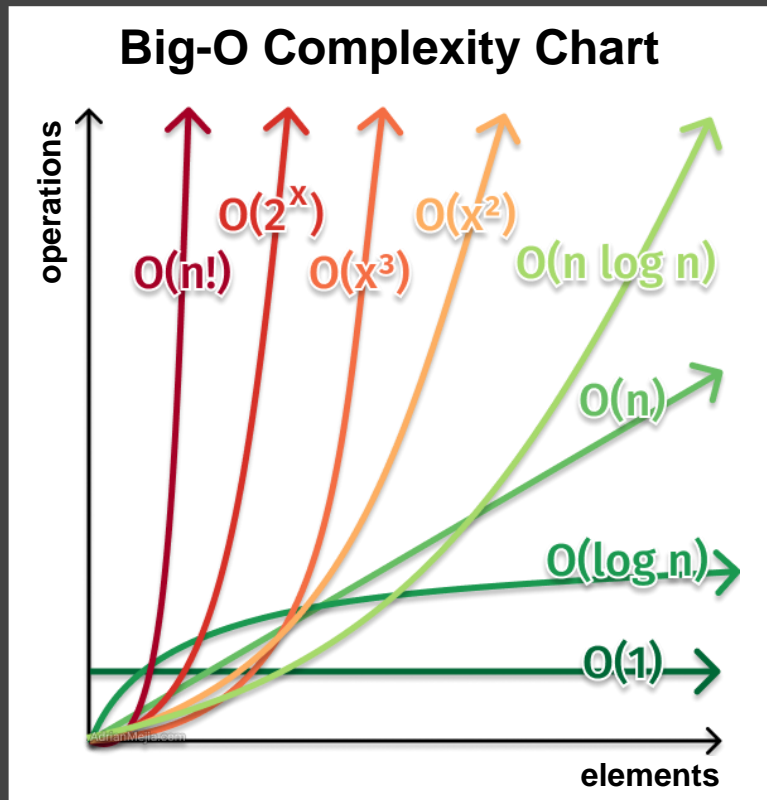
(즉, worst case를 비교하는 것임.)

표기 방법 : $O(-)$

Ex)

$f(N) = N^4 + 100N^2 + 10N + 50$ 의 Big-O Notation을 구하면? $\rightarrow O(N^4)$

(2) Big-O 표기법



Big-O Complexity		
complexity ↑	$O(N!)$	Factorial
	$O(2^N)$	Exponential
	$O(N^3)$	Cubic
	$O(N^2)$	Quadratic
	$O(N \log N)$	$N \times \log N$
	$O(N)$	linear
	$O(\log N)$	Logarithmic
	$O(1)$	Constant

(3) 문제집

Week 2 : 복잡도 계산 문제집

<https://www.acmicpc.net/workbook/view/12673>

전부 풀이 업로드
너무 오래 걸리는 문제는 풀지 않기

[알고리즘] 시간 복잡도 예제 15종

<https://dingrr.com/blog/post/%EC%95%8C%EA%B3%A0%EB%A6%AC%EC%A6%98-%EC%8B%9C%EA%B0%84%EB%B3%B5%EC%9E%A1%EB%8F%84-%EC%98%88%EC%A0%9C-15%EC%A2%85>

→ 시간 복잡도 개념이 부족하신 분은 풀어 보시면 됩니다.