

# 자료구조 기초

2021년 2월 10일 수요일    오후 4:12

## 242. 자료구조 기초

### • 자료구조(data structure)란?

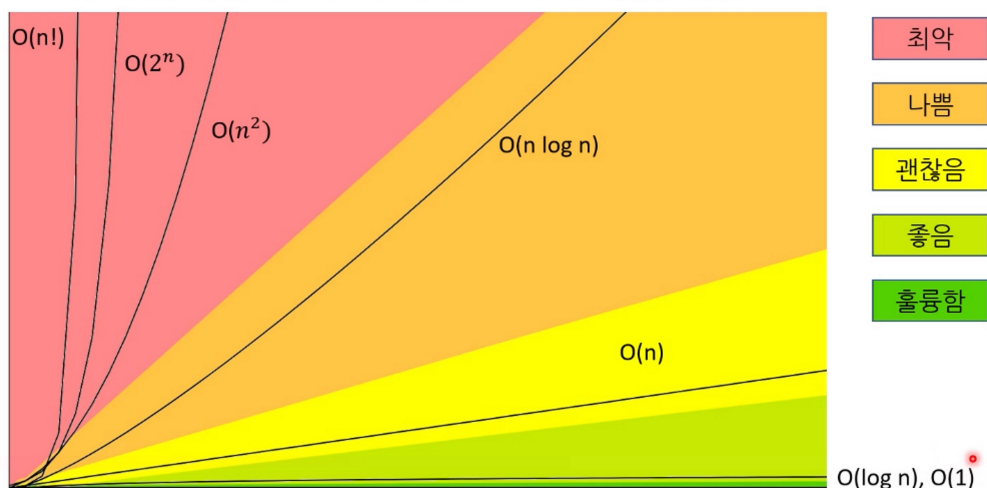
- 컴퓨터에서 여러 자료들을 조직적, 체계적으로 저장하는 방법
- 보통 동일한 자료형을 여럿 저장하는 구조를 의미
- 자료구조에 따라 요소들 사이의 관계를 정의하는 규칙이 있음
- 다음 요인에 따라 상황마다 보다 효율적인 자료구조가 존재
  - 데이터에 접근하는 빈도
  - 데이터에 접근하는 방법
    - 예: 삽입, 검색, 읽기, 지우기 등

### • 자료구조의 효율성

- 효율성은 주로 시간 복잡도(time complexity)를 말함
- 공간 복잡도(space complexity)를 포함하는 경우도 있음
- 따라서 주로 빅오(Big-O) 표기법을 사용
- 보통 효율성을 논할 때는 하드웨어 최적화를 고려 안한 이론이 전부
  - 대용량의 데이터를 사용할 때는 그래도 맞음
  - 적은 용량의 데이터는 그렇지 않을 수 있음

### • 복습: 흔히 알려진 O(?)의 연산속도

- 소프트웨어 공학용 수학 과목에서 이미 배운 내용



### • c에서 굳이 자료구조를 배우는 이유

- 배열을 제외한 자료구조는 하드웨어 위에서 프로그래머가 만든 개념
- 즉, 우리의 상위 호환성 사람들이 만들어 준 것
- 우리도 그 사람들처럼 이런 것 만들 줄 알아야겠지?
- 우리도 프로그래머니까
- 다른 언어에서는 이런 자료구조들을 라이브러리로 제공하기에 프로그래머가 제대로 구현하며 배울 기회가 적음

- 자칫하면 추상적으로만 이해해서 '이건 마법처럼 도는구나'라고 믿고 넘어갈 수 있는 부분을 제대로 이해할 좋은 기회
- 포인터를 연습하기도 매우 훌륭한 기회

## • 이 과목에서 살펴볼 자료구조들

- 다양한 자료구조 중에서도 기초 중의 기초 몇 가지만 배울 예정
  1. 배열
  2. 스택
  3. 큐
  4. 연결 리스트
  5. 해시 테이블(플러스알파)
- 나머지는 '알고리즘 및 자료구조' 과목에서 배움
- 다른 과목에서도 조금씩 배움

## • 자료구조의 시간 복잡도

자료구조	평균(average)			최악(worst)		
	검색	삽입	삭제	검색	삽입	삭제
배열	$O(n)$	$O(n)$	$O(n)$	$O(n)$	$O(n)$	$O(n)$
스택	$O(n)$	$O(1)$	$O(1)$	$O(n)$	$O(1)$	$O(1)$
큐	$O(n)$	$O(1)$	$O(1)$	$O(n)$	$O(1)$	$O(1)$
연결 리스트	$O(n)$	$O(1)$	$O(1)$	$O(n)$	$O(1)$	$O(1)$
해시 테이블	$O(1)$	$O(1)$	$O(1)$	$O(n)$	$O(n)$	$O(n)$

- 참고: 공간 복잡도는 다  $O(n)$
- 그 밖의 다른 자료구조의 시간 복잡도가 궁금하면 스스로 찾아보자