

1주차 - 스택, 큐

검색



# 자료구조 학술회

게임소프트웨어 전공 3학년 고병진

# 자료 구조란?



## 사전적 의미

- 자료(Data)의 집합.
- 여러 데이터들의 묶음을 저장하고, 사용하는 방법을 정의한 것.



# ADT(추상적 자료형)?



× □ -

## 추상화

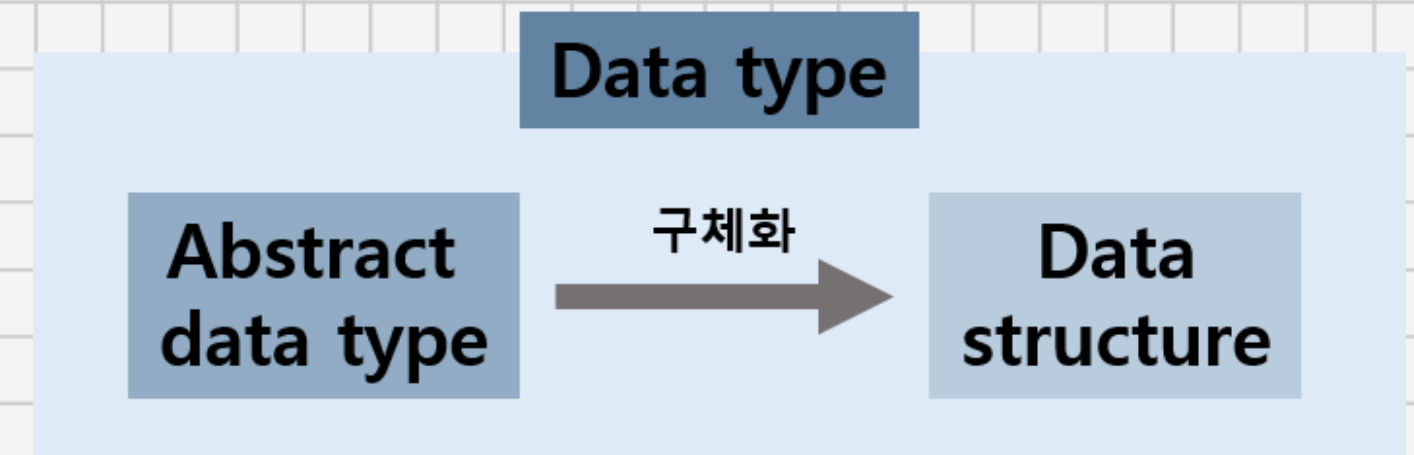
- 중요한 정보만 남기고 세부사항은 공개X

## 데이터

- 정수, 실수 같은 형태를 가지는 혹은 특성을 가지는 것

## ADT

- 자료 자체의 형태와 자료에 연관된 계산들을 수학적으로만 정의한 것



## 알고리즘의 성능 분석?



✕ □ -

### 수행시간 측정

- 두개의 알고리즘의 실제 수행 시간을 측정
- 실제로 구현하는 것이 필요
- 동일한 하드웨어를 사용하여야 함

### 알고리즘 복잡도 분석

직접 구현하지 않고서도 수행 시간을 분석

- 알고리즘이 수행하는 연산의 횟수를 계산하여 비교
- 일반적으로, 연산의 횟수는  $n$ 의 함수 (여기서  $n$ 은 충분히 큰 양의 정수를 가정)

| 입력 자료의 개수 | 프로그램 A | 프로그램 B             |
|-----------|--------|--------------------|
| $n = 6$   | 36초    | 64초                |
| $n = 100$ | 10000초 | $4 \times 10^{22}$ |

# 복잡도?



✕ □ -

## 시간 복잡도

- 알고리즘을 이루고 있는 연산들이 (주어진 입력  $n$ 에 대해) 몇 번이나 수행되는지를 숫자로 표시

## 공간 복잡도

- 프로그램을 실행시킨 후 완료하는 데 필요로 하는 자원 공간의 양



### 알고리즘 A

```
sum ← n * n;
```

### 알고리즘 B

```
for i ← 1 to n do
```

```
  sum ← sum + n;
```

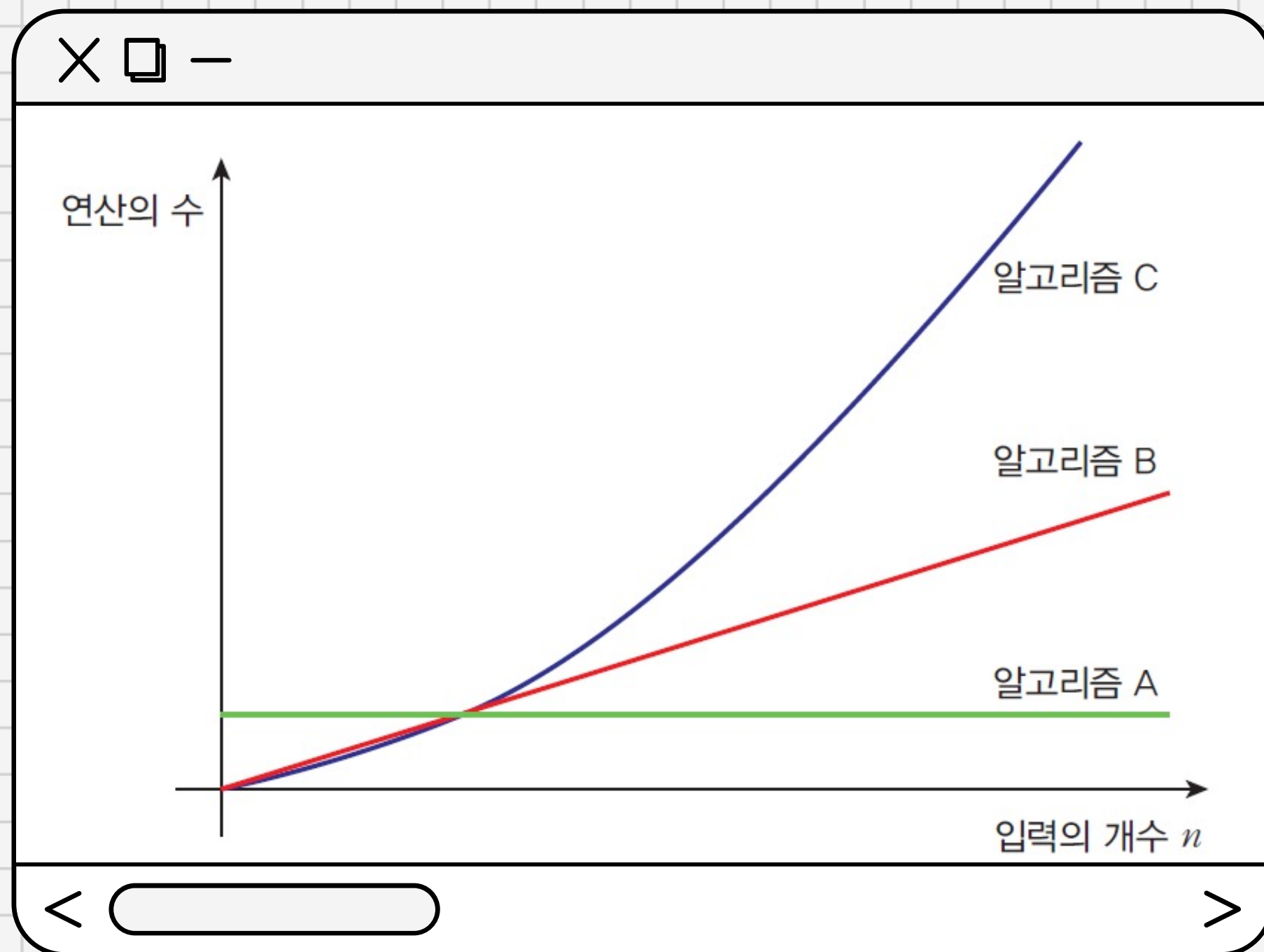
### 알고리즘 C

```
for i ← 1 to n do
```

```
  for j ← i to n do
```

```
    sum ← sum + 1;
```

# 복잡도?



## 빅오 표기법(Big-O Notation)?



✕ □ -

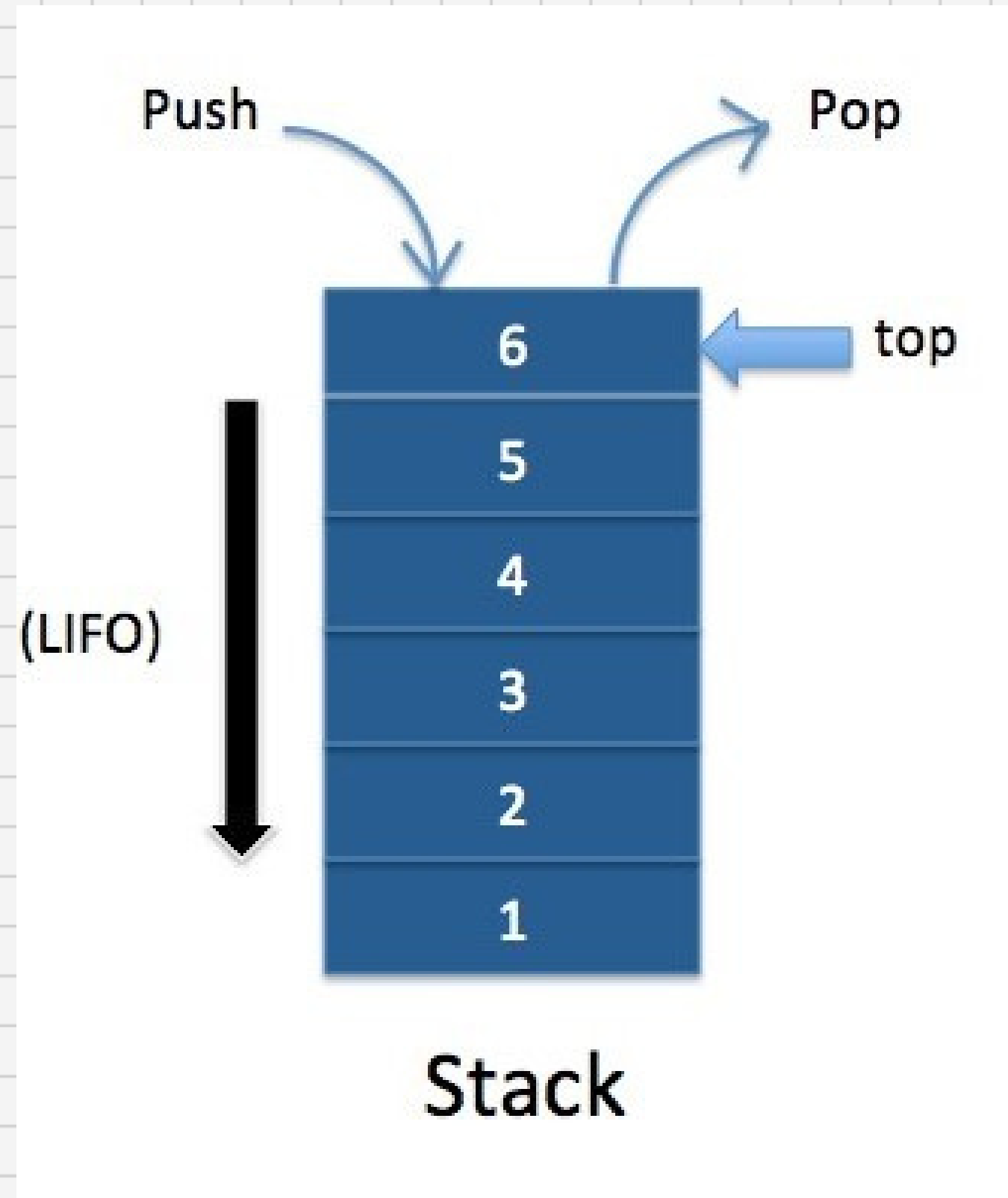
- 연산의 횟수를 대략적(점근적)으로 표기한 것  
(단,  $n$ 은 충분히 크다고 가정)

### 수학적 정의

- 두개의 함수  $f(n)$ 과  $g(n)$ 이 주어졌을 때, 모든  $n \geq n_0$ 에 대하여  $|f(n)| \leq c|g(n)|$ 을 만족하는 2개의 상수  $c$ 와  $n_0$ 가 존재하면  $f(n) = O(g(n))$ 이다.

- 빅오는 함수의 상한 (Upper bound)을 표시한다.  
Ex.  $n \geq 5$  이면  $2n+1 < 10n$  이므로  $2n+1 = O(n)$





## 스택(Stack)



Last In First Out: 마지막에 들어온 것이 제일 먼저 나감  
(후입선출) ex. 블록 쌓기, 그릇 쌓기

push: 제일 위(마지막)에 새로운 원소 넣기  
top(peek): 가장 위(마지막)에 있는 원소가 무엇인지 알리기  
pop: 제일 위(마지막)에 있는 원소 빼기  
isEmpty: 배열이 비어 있는지 확인  $\text{if}(\text{top의 위치}) == -1$   
isFull: 배열이 꽉 찼는지 확인  $\text{if}(\text{top의 위치}) == (\text{size}) - 1$   
size: 저장된 원소가 몇 개 있는지 알리기 ( $\text{top의 인덱스} + 1$ )

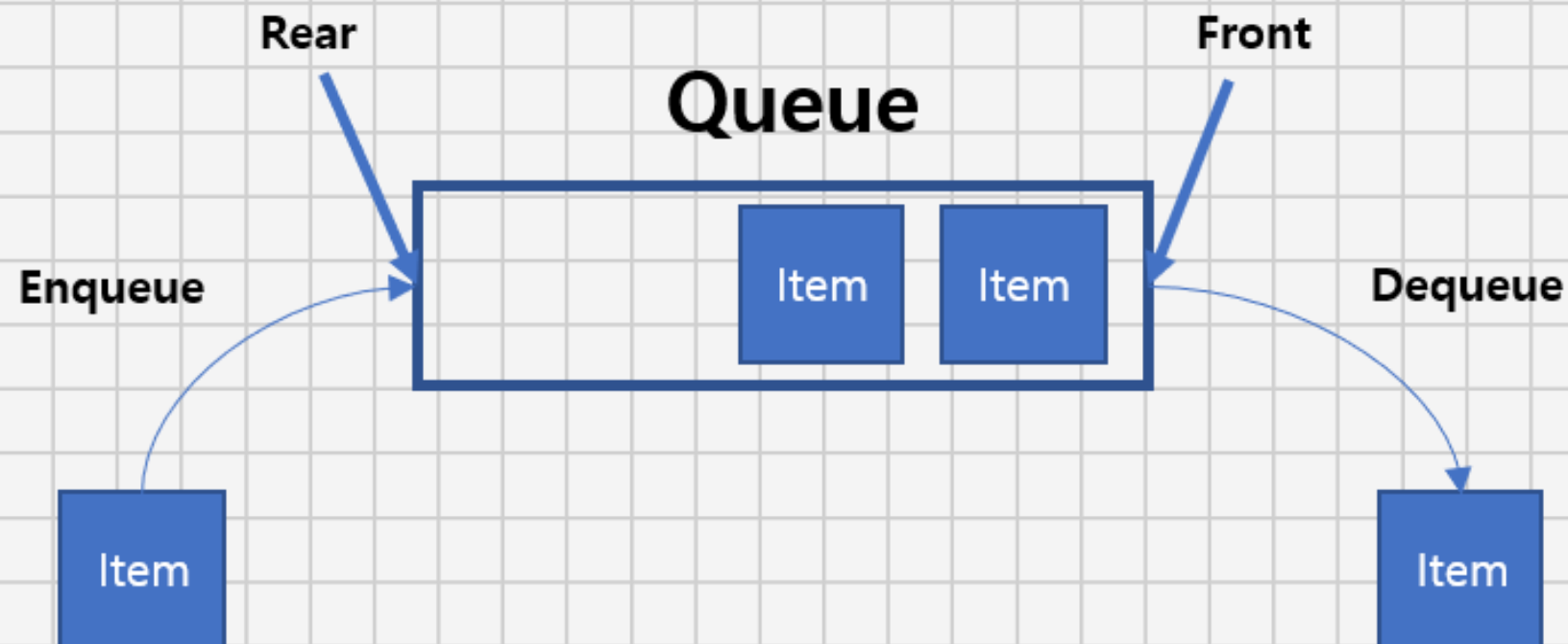
후입선출(LIFO)로 작동되게 설계된 컨테이너 어댑터  
컨테이너 어댑터: 이러한 방식을 이용하는 다른 자료구조에게  
인터페이스 제공  
인터페이스: 메뉴판같이 어떠한 기능을 제공한다고 말해주는  
기능



# 큐(Quque)



First In First Out: 먼저 들어온 것이 제일 먼저 나감  
(선입선출) ex. 키보드 입력, 맛집줄 기다리기



push: 제일 마지막에 새로운 원소 넣기  
front(peek): 가장 앞에 있는 원소가 무엇인지 알리기  
pop: 제일 마지막에 있는 원소 빼기  
empty: 배열이 비어 있는지 확인  
back(rear): 가장 마지막에 있는 원소가 무엇인지 알리기  
size: 저장된 원소가 몇 개 있는지 알리기 IF(back < front)

후입선출(LIFO)로 작동되게 설계된 컨테이너 어댑터  
컨테이너 어댑터: 이러한 방식을 이용하는 다른 자료구조에게  
인터페이스 제공  
인터페이스: 메뉴판같이 어떠한 기능을 제공한다고 말해주는  
기능