

D.cOm



Algorithm - 스터디

#Beginner 편

3강. 자료구조



오늘의 학습 목표!

1. 자료구조를 배워야 하는 이유 알기!

2. 기초적인 자료구조의 종류와 개념을 알아봅시다!

강의 자료 개요

별표 친 페이지는 반드시 읽어주세요!



이론

이론 내용입니다. 가벼운 마음으로 읽어주세요!

실습

百聞이 不如一打!
직접 따라해보세요!

읽을거리

읽어두면 배운 것을 심층적으로 이해하는데 있어 도움이 되는
내용입니다! 여유가 있으면 읽어주세요!



알고리즘

Algorithm

어떤 문제를 해결하는 방법

자료구조

Data Structure

데이터를 표현하고 저장하는 방법

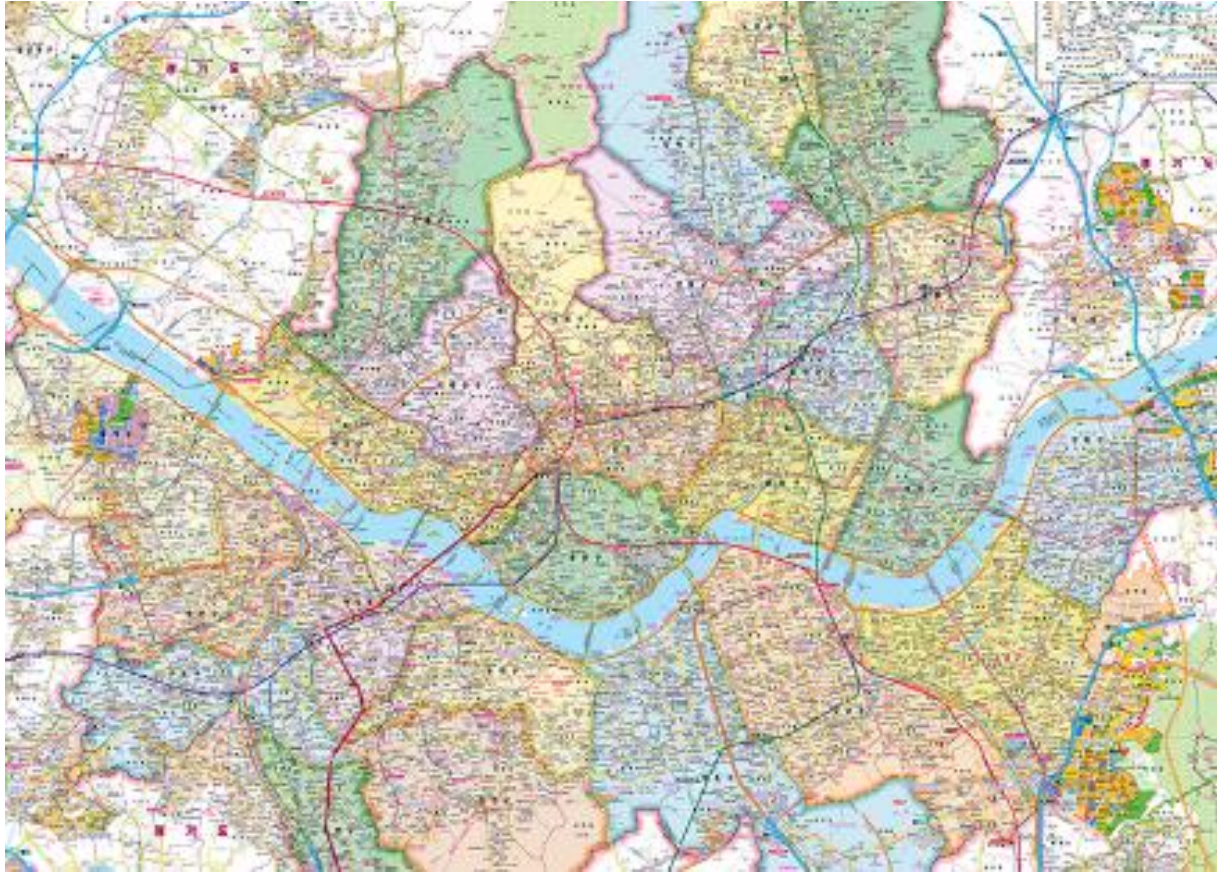
우리는 왜 자료구조를 배워야 할까요?



왜 자료구조를 배워야 하는가?

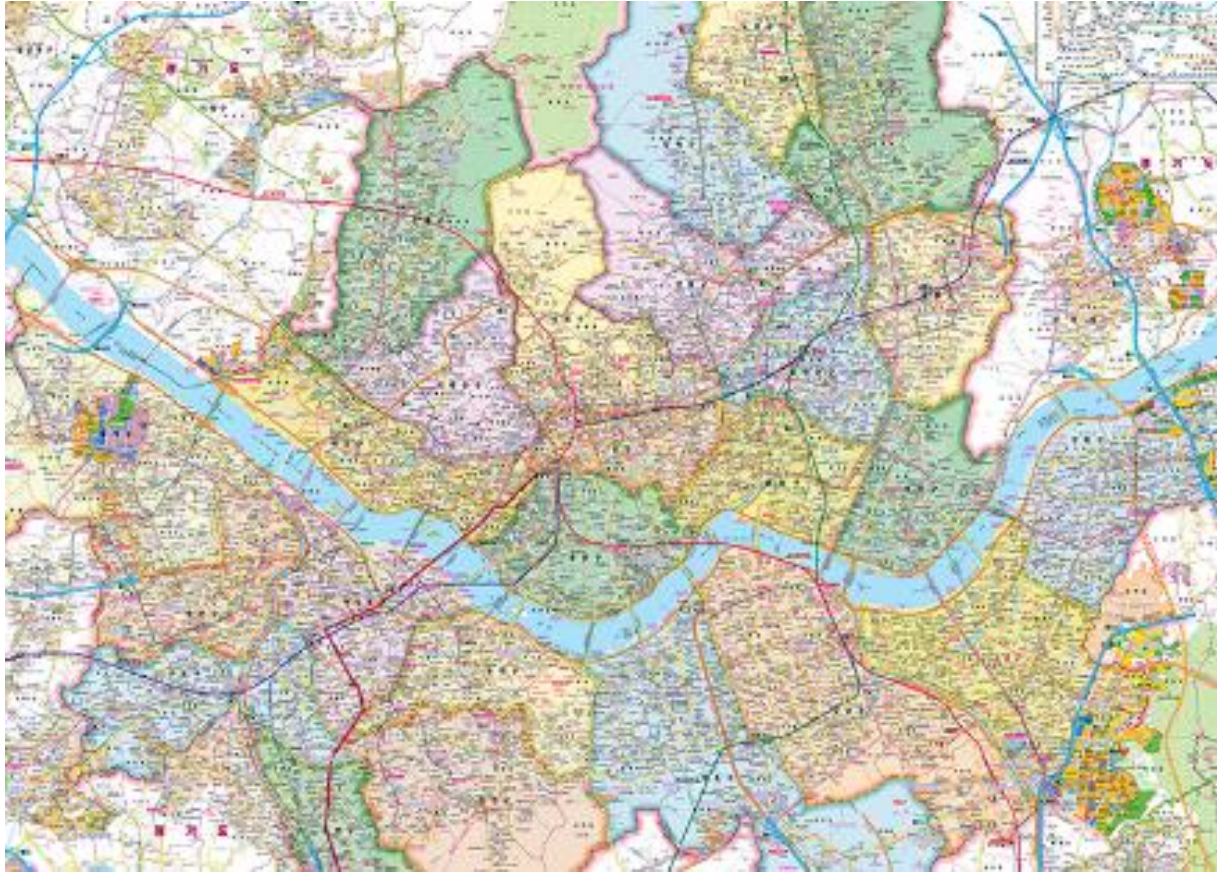
복잡한 문제를 간단하게
'추상화' 및 '단순화'
할 수 있습니다.

왜 자료구조를 배워야 하는가?



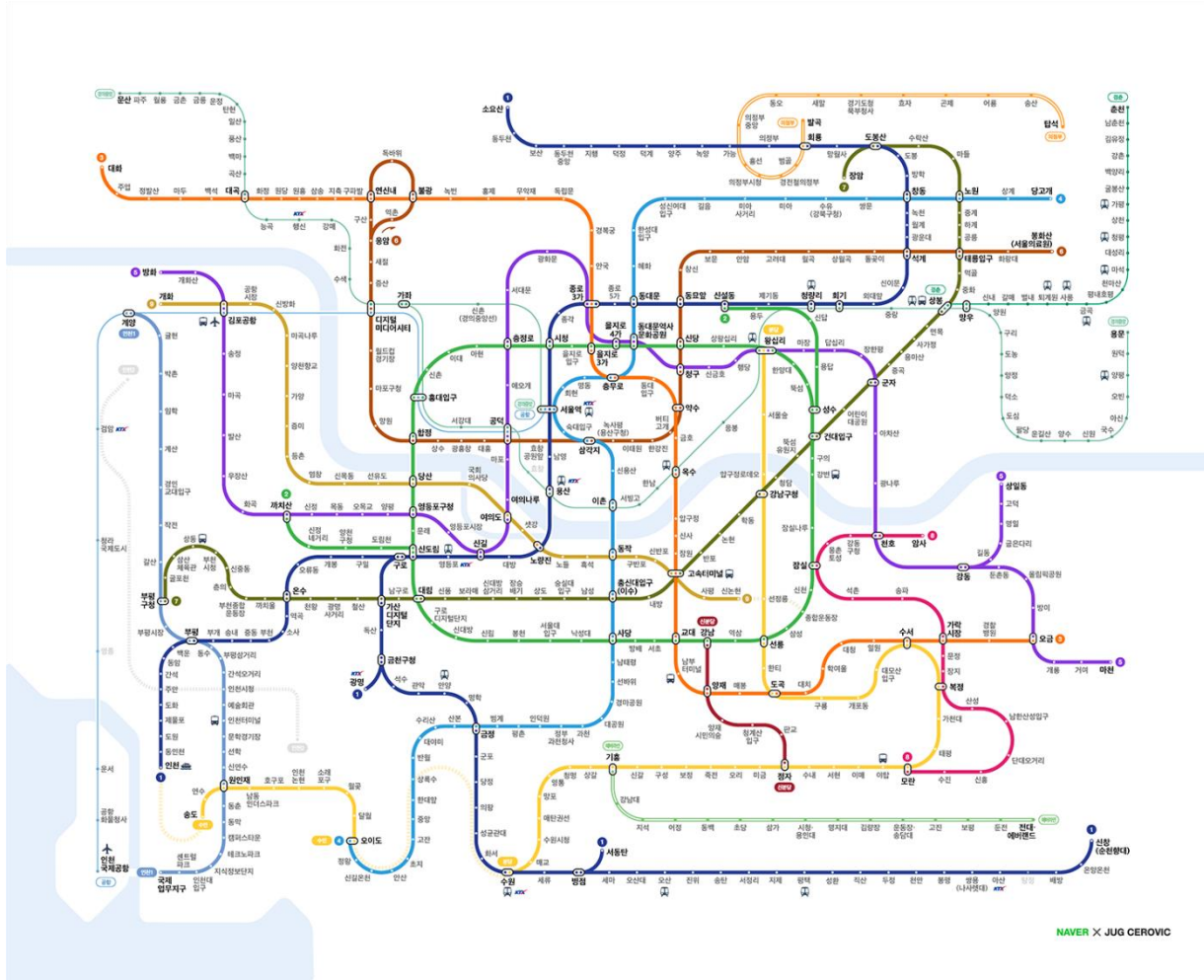
지하철을 타고 '종로'에서 '은평'을 가려면
어떻게 해야 할까요?

왜 자료구조를 배워야 하는가?



실제 지하철 역과 노선의 좌표 데이터를 활용하면
문제가 매우 복잡 해집니다.

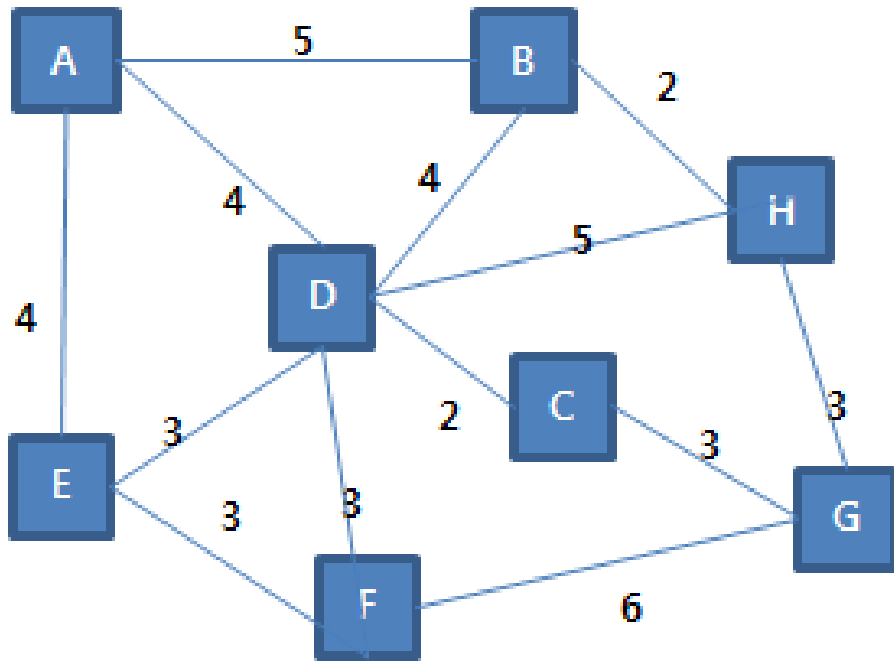
왜 자료구조를 배워야 하는가?



따라서 우리는 지하철 노선을
다음과 같이 ‘추상화’와 ‘단순화’하여 사용합니다.

#서울 지하철 노선도가 간단한가요..?

왜 자료구조를 배워야 하는가?



#그래프는 추후 강의자료에서 직접도록 다룹니다!

실제로 우리는 ‘그래프’라는 자료구조를 이용하여
지하철 역간 최적의 경로를 구할 수 있습니다!

왜 자료구조를 배워야 하는가?

즉, 자료구조를 통해서
우리는 문제를 단순화 및 추상화 시켜,
문제해결의 실마리를 찾을 수 있습니다!

#고등학교 때 수학 공부에서 유형 숙달을 강조하는 이유와 비슷합니다!

자료구조



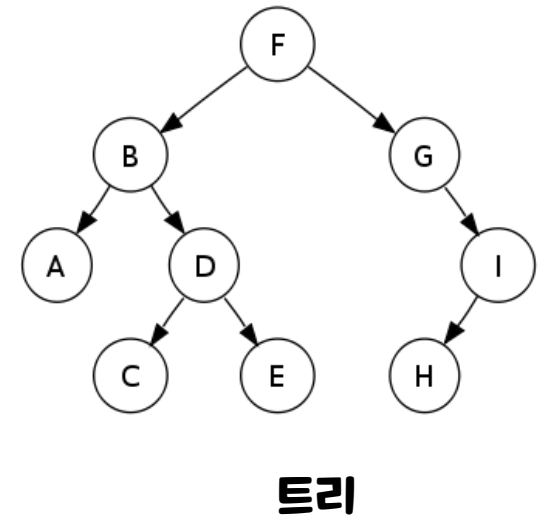
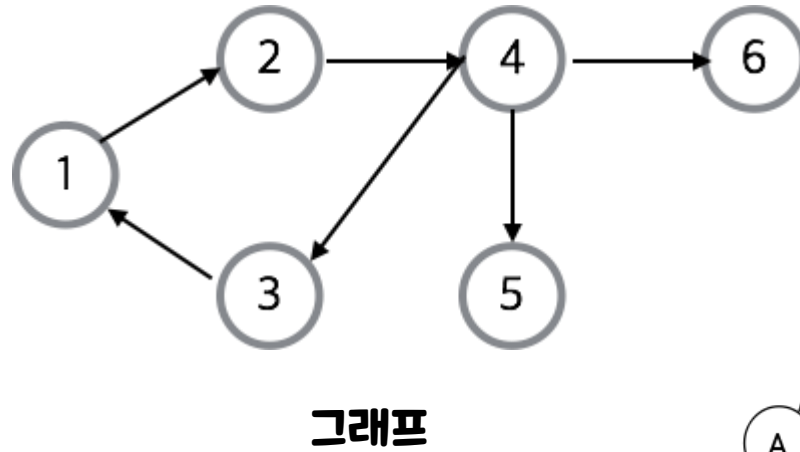
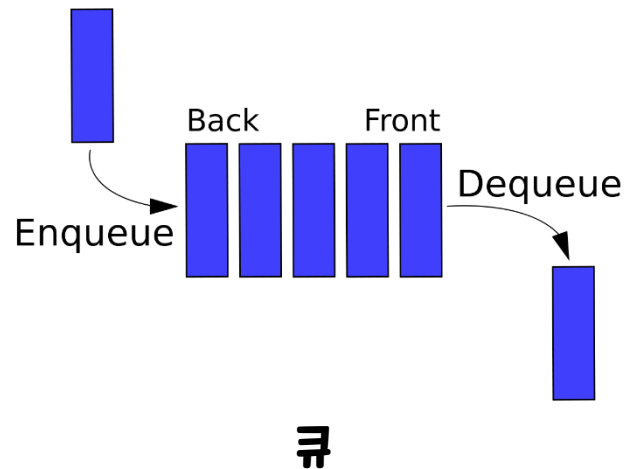
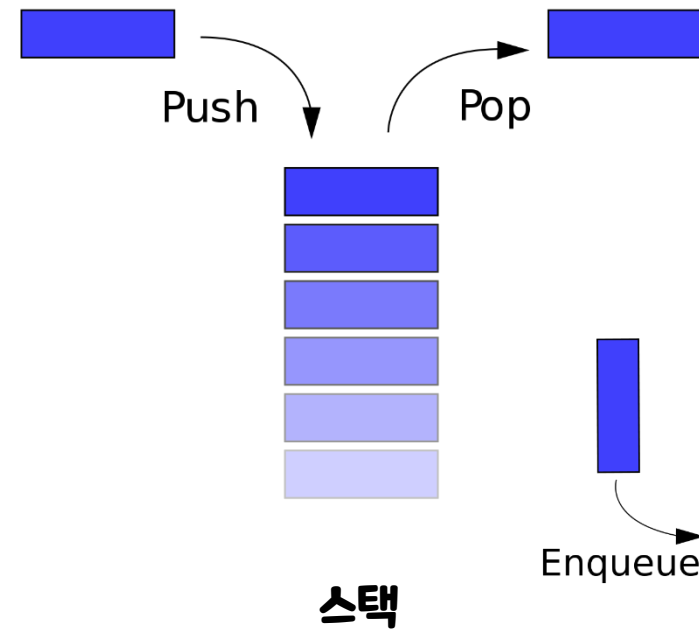
자료구조

Data Structure

데이터를 표현하고 저장하는 방법

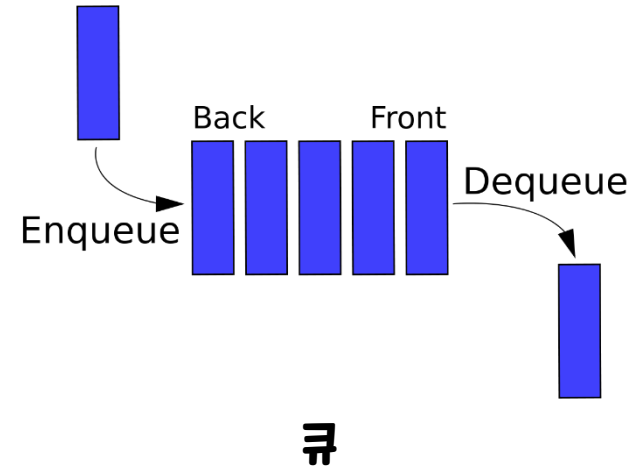
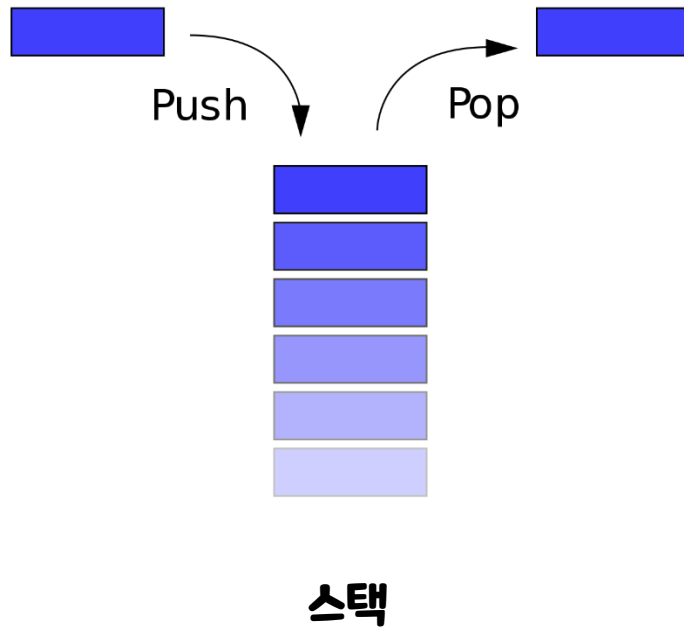
자료구조

대표적인 자료구조는 다음과 같습니다!

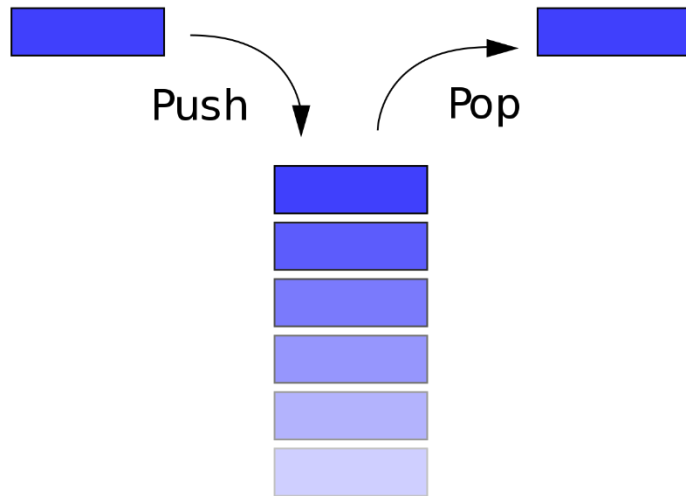


자료구조

이번 강좌에서는 스택과 큐에 대해 알아보시다!



스택



스택

Stack



#Like 프링글스

한 쪽 끝에서만 자료를 넣거나 뺄 수 있는 선형 자료구조

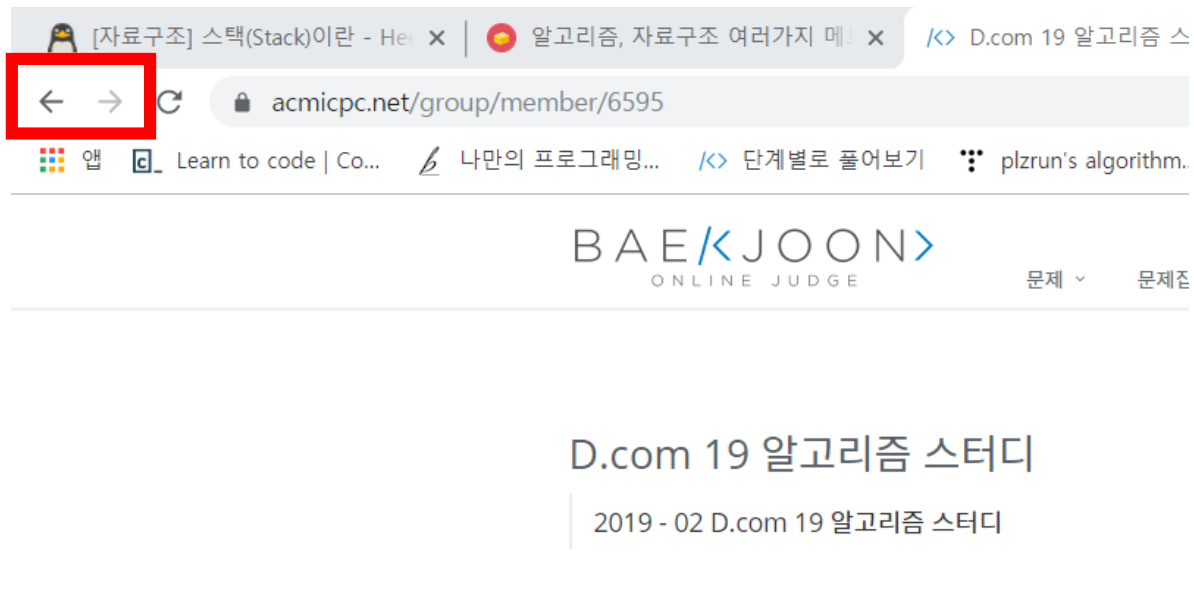
LIFO(Last In First Out)

스택

스택을 어디에 활용할 수 있을까요?!

대표적인 예시로는 이전 작업(페이지), 다음 작업(페이지) 기능이 있습니다!

직접 사용해봅시다 <=



스택



네이버 접속 -> 네이버 메일 접속-> 뒤로 가기 -> 네이버 블로그 접속

이제 한번 더 뒤로 가기를 하면 어느 페이지로 갈까요?

스택



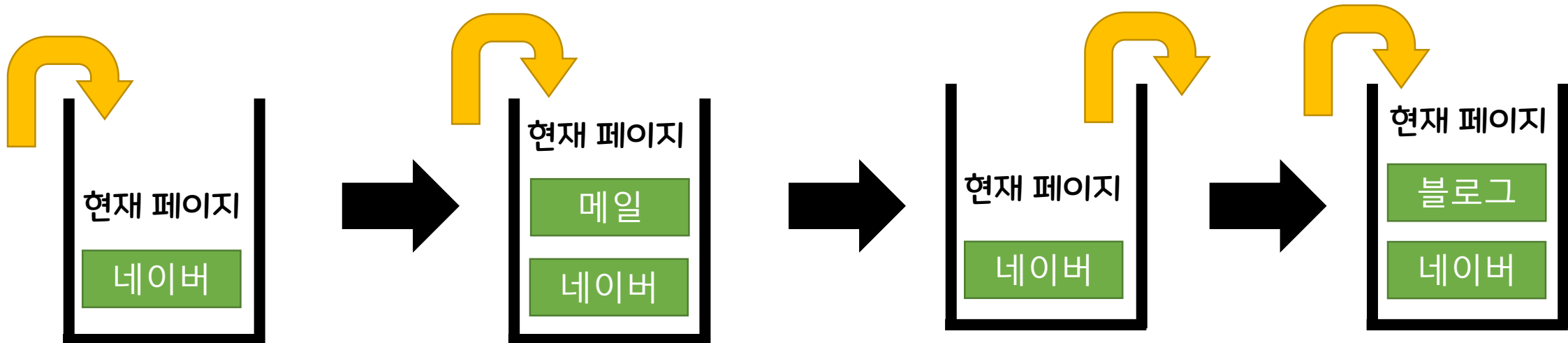
네이버 접속 -> 네이버 메일 접속-> 뒤로 가기 -> 네이버 블로그 접속

한번 스택을 이용해서 생각해봅시다!

스택

네이버 접속 -> 네이버 메일 접속-> 뒤로 가기 -> 네이버 블로그 접속

상단의 데이터가 현재 페이지입니다!
접속하는 페이지를 스택에 넣고,
뒤로 가기를 하면 스택 상단의 페이지를 뺍니다!

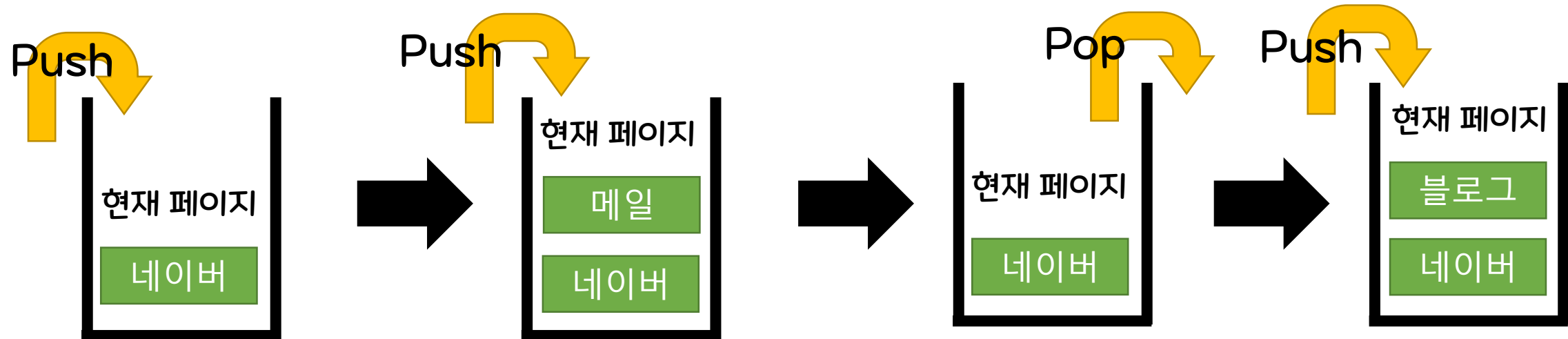


스택

네이버 접속 -> 네이버 메일 접속-> 뒤로 가기 -> 네이버 블로그 접속

상단의 데이터가 현재 페이지입니다!
접속하는 페이지를 스택에 넣고,
뒤로 가기를 하면 스택 상단의 페이지를 뺍니다!

➡ Top
➡ Push
➡ Pop



스택



스택을 구현하려면 다음과 같은 연산들이 필요합니다!

- Push: 스택에 자료를 넣는 연산
- pop: 스택에 자료를 빼는 연산
- top: 스택 최상단의 자료를 조회하는 연산
- Empty: 스택이 비어 있는지 판정하는 연산
- size: 스택에 저장 되어있는 자료의 개수를 조회하는 연산

스택



이제 스택을 구현해봅시다!

(이미 구현되어 있는 STL 라이브러리를 이용합니다!)

직접 손수 구현하는 방법은 2학년 자료구조 수업에서 배웁니다!

스택



STL(Standard Template Library) 표준 C++ 라이브러리

스택, 큐, 벡터 등 프로그램에 필요한 자료구조와 알고리즘을
템플릿으로 제공하는 라이브러리입니다!

스택



C++ STL Stack 기본 사용법

<https://twpower.github.io/75-how-to-use-stack-in-cpp>

실습문제



C++ STL Stack을 활용하여 간단한 문제를 풀어봅시다!

<https://www.acmicpc.net/problem/10828>

실습문제

스택을 응용한 문제를 풀어봅시다!

<https://www.acmicpc.net/problem/9093>

**#Stack은 LIFO(Last In First Out)
구조임을 명심합시다!**

실습문제

**스택을 이용해서
괄호 검사는 어떻게 할 수 있을까요?**

<https://www.acmicpc.net/problem/9093>

**#가장 늦게 들어간 열린 괄호[(]가 그 뒤 가장 먼저 나오는 닫는 괄호[)]와
짝을 이루어야 한다!**

실습문제

힘내서 스택 문제만 좀 더 봅시다!

<https://www.acmicpc.net/problem/1873>

#뒤 슬라이드에서 문제 부연 설명 들어갑니다~!

실습문제

1873번 부연 설명

어떤 스택 S에 1,2,3 .. N까지 차례대로 push 할 수 있습니다.
이때 S에서 pop한 수들로 문제에서 입력으로 주어진 수열R을
표현할 수 있는가 묻는 문제입니다.

Ex) 4 3 6을 만들려면?

S : 1	R :	-> +
S : 1, 2	R :	-> +
S : 1, 2, 3	R :	-> +
S : 1, 2, 3, 4	R :	-> +
S : 1, 2, 3	R : 4	-> -
S : 1, 2	R : 4, 3	-> -
S : 1, 2, 5	R : 4, 3	-> +
S : 1, 2, 5, 6	R : 4, 3	-> +
S : 1, 2, 5	R : 4, 3, 6	-> -

실습문제

에디터도 만들 수 있습니다!

<https://www.acmicpc.net/problem/1406>

#문제에서 스택을 꼭 하나만 쓰라는 법은 없습니다.

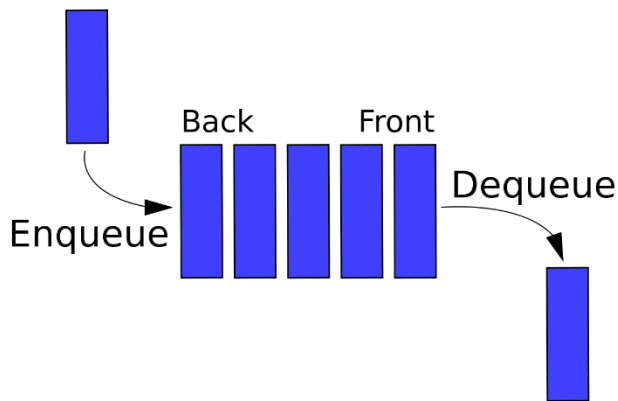
큐

큐



© Can Stock Photo

#Like 버정 줄



Queue

한 쪽 끝에서만 자료를 넣고 다른 한쪽
끝에서만 뺄 수 있는 선형 자료구조

LILO(Last In Last Out)

#게임에서의 큐도 비슷한 의미입니다! “여우 큐 더럽게 안 잡히네”

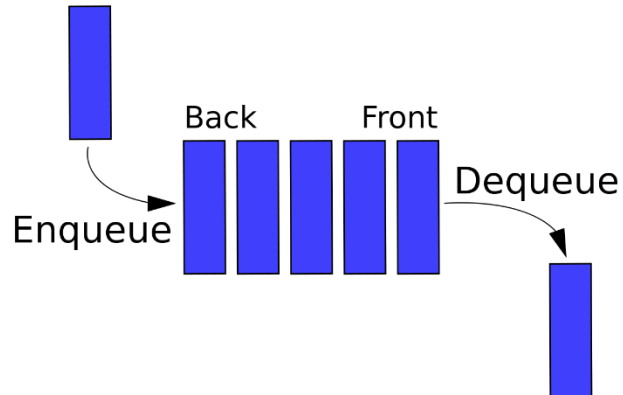
큐

큐는 어디에 활용할 수 있을까요?!

큐는 이미 일상생활에서의 대기열 등 많은 곳에서 찾아볼 수 있습니다!

컴퓨터에서도 프린터 문서 대기, 버퍼링 등
순차적인 작업이 필요하면 거의 큐가 사용됩니다!

#새치기는 안됩니다!



큐



**큐도 C++ STL을 활용하여 간단하게
구현할 수 있습니다!**

<https://twpower.github.io/76-how-to-use-queue-in-cpp>

큐



큐를 구현하려면 다음과 같은 연산들이 필요합니다!

- Push: 큐에 자료를 넣는 연산
- pop: 스택에 자료를 빼는 연산
- front: 큐의 가장 앞의 자료를 조회하는 연산
- Back: 큐의 가장 뒤의 자료를 조회하는 연산
- Empty: 큐가 비어 있는지 판정하는 연산
- size: 스택에 저장 되어있는 자료의 개수를 조회하는 연산

실습문제



C++ STL Queue를 활용하여 간단한 문제를 풀어봅시다!

<https://www.acmicpc.net/problem/10845>

실습문제

C++ STL Queue를 활용하여 간단한 문제를 풀어봅시다!2

<https://www.acmicpc.net/problem/1158>

#문제에서 요구하는 대로 큐를 활용하면 되는 간단한 문제입니다!

덱(Deque)은 뭐죠!?

<https://www.acmicpc.net/problem/10866>

**#스택과 큐를 합친 개념입니다. 양쪽에서 자료를 넣고 빼고 할 수 있습니다.
물론 덱도 STL을 제공합니다!**

읽을 거리



이번 강좌에서는 스택과 큐에 대해서 배웁니다.
두 자료구조는 앞으로 배울 그래프, 트리 등의 자료구조를
구현할 때 사용이 됩니다!

또한 CPU, 메모리, 프로그램 등의 동작과 연산에서
가장 기초적이고 중요하게 사용되는 자료구조 입니다.

반드시, 꼭! 스택과 큐의 특징은 기억해두시기 바랍니다!

다음시간!

수학의 꽃이 미적분이라면,
알고리즘의 꽃은 DP입니다.

다음시간에는 Dynamic Programming에 대해 알아보니다!



The end.

“네 시작은 미약하였으나 네 나중은 심히 창대하리라”
-욥기 8장 7절