

Programmer

Before
COVID-19



During
COVID-19



After
COVID-19



자료구조 및 실습

과목소개

• 교수:

- 권 오 흠 (ohheum@gmail.com)
- A13동(누리관) 2312호, 051-629-6255

• 참고서적

- 천인국 등, C 언어로 쉽게 풀어쓴 자료구조, 생능출판
- K. N. King, C Programming: A Modern Approach, 2nd Ed., Norton.
- C로 배우는 쉬운 자료구조, 한빛아카데미

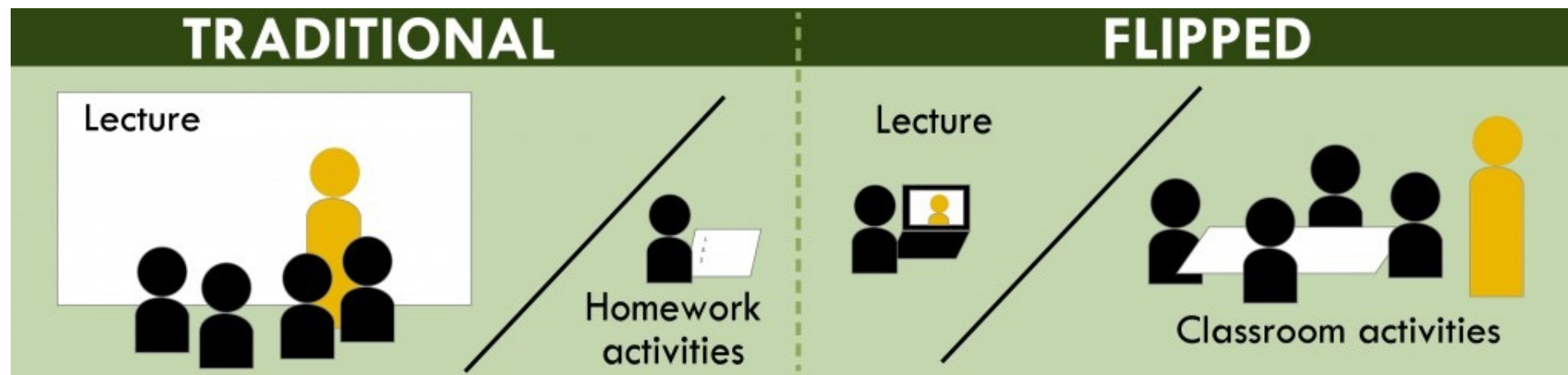
• 사용언어: C

• 수업 및 실습: 플립드러닝(flipped learning)

• 강의 홈페이지

- 부경 LMS

Flipped Class



- 주 단위로 강의 동영상 - 수업에 참석하기 전에 미리 수강
- 주 단위로 **프로그래밍 연습문제**가 주어짐
- 수업에서는 각자 연습문제를 풀고, 질의 응답

● 성적 평가 기준

평가기준	반영비율	평가방법
시험	50%	<ul style="list-style-type: none"> • 중간 및 기말 시험 (Open Book 필기 및 실기 시험)
프로그래밍 연습문제	15%	<ul style="list-style-type: none"> • 매주 프로그래밍 연습문제
프로그래밍 과제	30%	<ul style="list-style-type: none"> • 3~4개의 프로그래밍 과제
출석, 수업참여도	5%↑	<ul style="list-style-type: none"> • 동영상 강의 시청 여부 (LMS에 의해서 자동 집계) • 결석 3번 부터 성적에 반영(nonlinear degradation) • 지각 2회 = 결석 1회

* 104분반의 경우 의공학IT융합 학생들과 나머지 학생들은 분리해서 평가한다.

주	주제	개요
1	강의소개	
2	C 언어 리뷰	
3	C 언어 리뷰	포인터, 구조체 등을 활용한 예제 프로젝 트
4	C 언어 리뷰	
5	리스트	
6	리스트	용용
7	리스트	
8	중간시험	

주	주제	개요
9	스택	
10	스택	스택 및 큐 ADT의 개 념, 구현, 응용
11	큐	
12	큐	
13	순환	용용
14	순환	
15	시간복잡도	시간복잡도 및 점근적 분석법의 이해
16	기말시험	

What is Data Structure?

- 연필과 공책으로 전화번호부를 만든다고 가정해보자.
- 필요한 것 2가지:
 - 전화번호와 이름을 공책에 어떻게 (순서, 구조 등) 기록할 것인지 (**data structure**)
 - 검색, 추가, 삭제 등을 어떻게 할 것인지 (**algorithm**)
- 일반적으로 프로그램은 “자료구조 + 알고리즘”

Unordered List

홍길동 010-4566-2746

김재영 017-3456-2726

박찬희 051-2345-2445

한준식 042-1234-5679

김규식 031-2453-2957

...

👁 특별한 규칙 없이 나열

👁 how to **lookup**?

👁 how to **add**?

👁 how to **remove**?

순서 리스트(Ordered List)

김규식 031-2453-2957

김재영 017-3456-2726

박찬희 051-2345-2445

한준식 042-1234-5679

홍길동 010-4566-2746

...

👁 **Alphabetical order**

👁 **Look up?**

👁 **Add?**

👁 **Remove?**

Linked List



- 👁️ **One card for each person**
- 👁️ **Lookup?**
- 👁️ **Add?**
- 👁️ **Remove?**

Indexed Structure



- 👁 **One page for each alphabet**
- 👁 **Lookup?**
- 👁 **Add?**
- 👁 **Remove?**



더 복잡하고 효율적인 방법들도 있음

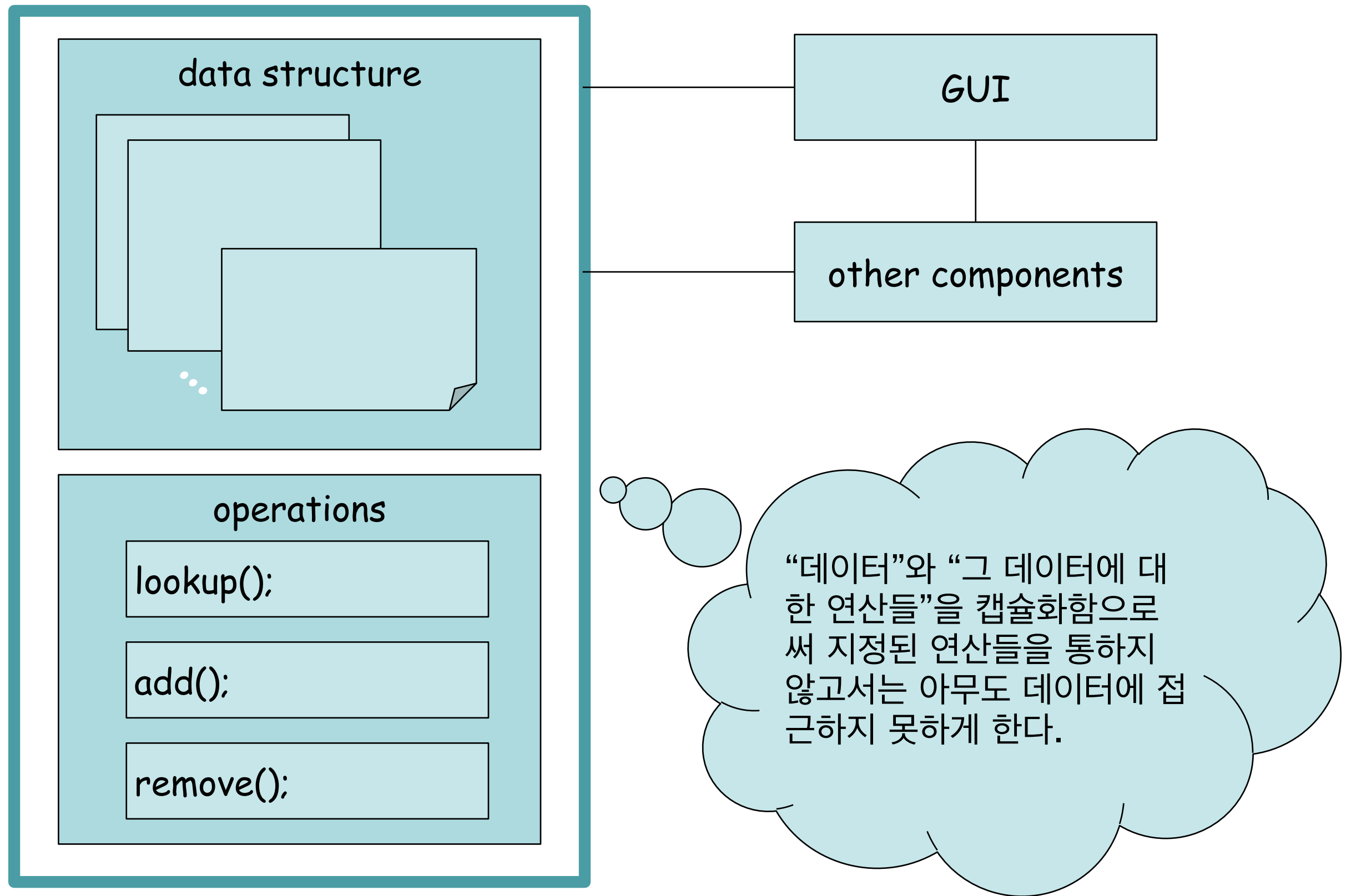
- binary search tree
- hash table
- ...

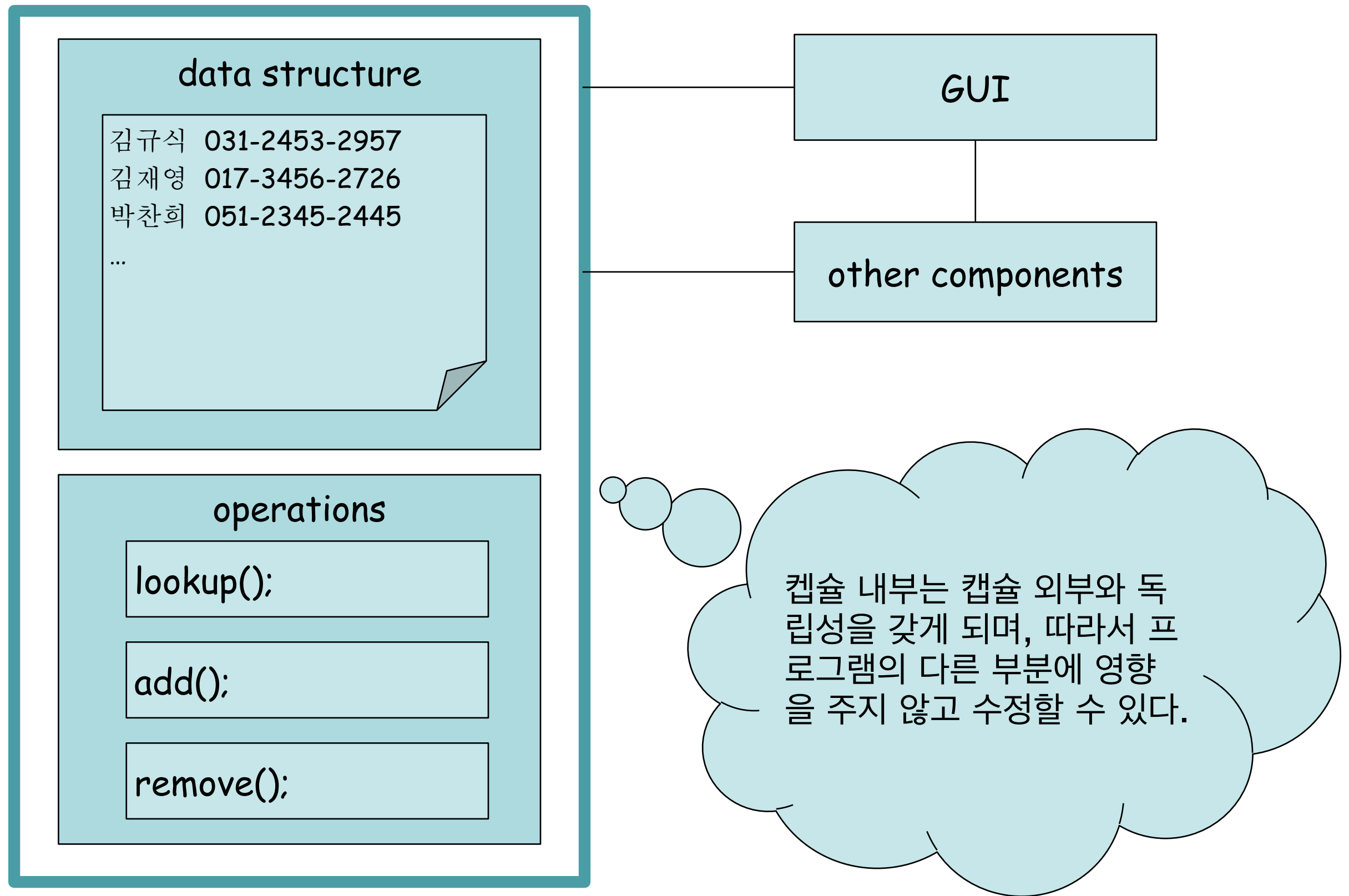
좋은 자료구조 ?

- 좋은 자료구조는 좋은 프로그램의 핵심이다.
- 그렇다면 좋은 자료구조란?
 - 성능 (performance) - 자료구조에 의해 알고리즘의 성능이 결정된다.
 - 높은 모듈성 (high modularity)

High Modularity

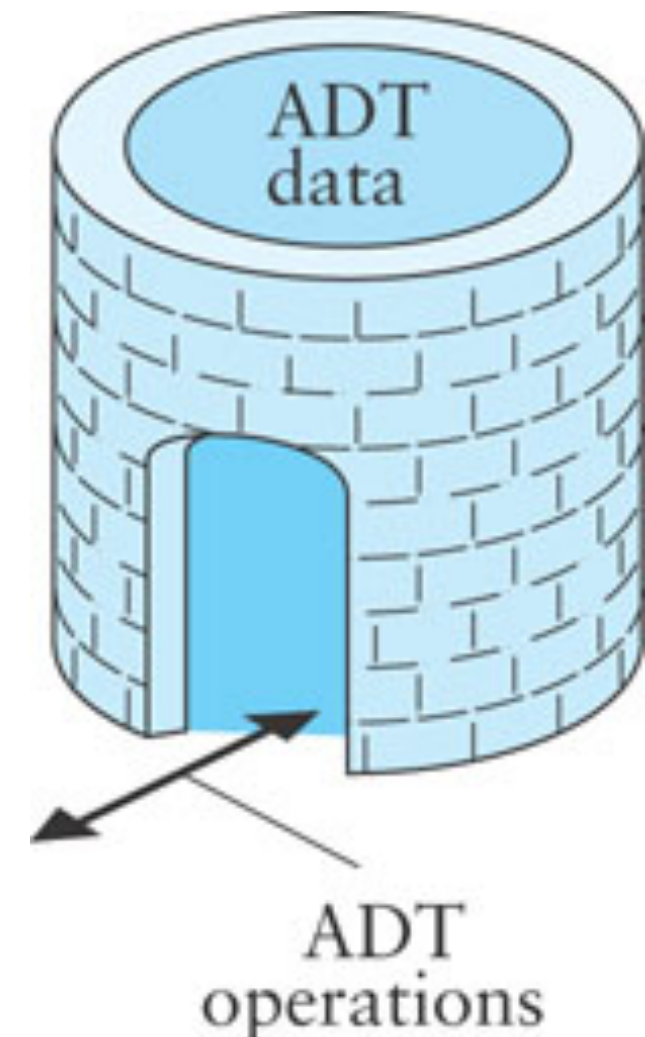
- 높은 모듈성은 프로그램을 이해, 수정, 재사용 하기 쉽게 한다.
- 높은 모듈성을 달성하는 한가지 방법은 “데이터”를 “그 데이터에 대해 작용하는 연산들”과 함께 하나의 독립된 모듈로 캡슐화(encapsulation)하는 것





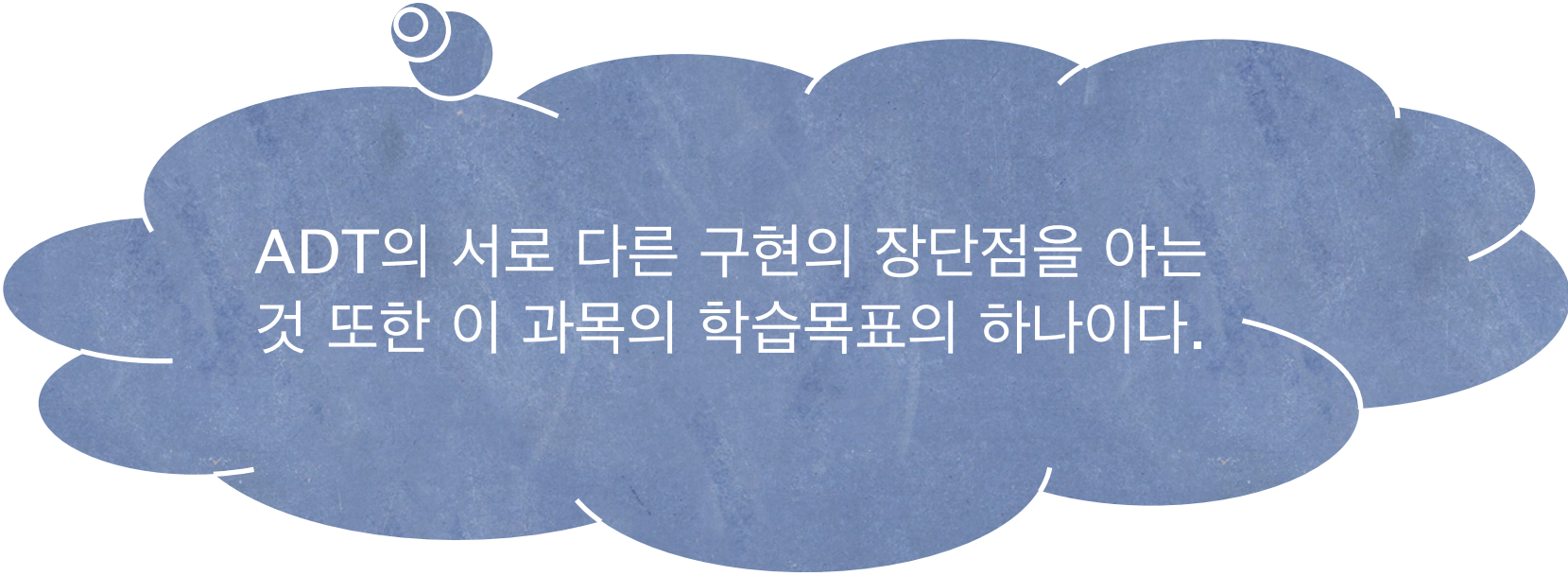
Abstract Date Type

- **Abstract Data Type (ADT)**
 - 자료구조와 그에 관련된 연산들의 조합
 - C 언어에서는 ADT에 대한 지원이 미흡
- **사용자는 ADT의 구현에 대해 알 필요가 없음**



Abstract Data Type

- 이 수업에서는 대표적인 **ADT**들을 학습
 - List, Stack, Queue, Tree 등.
- 각각의 **ADT**은 서로 다른 방식으로 구현될 수 있음
 - 가령 List ADT는 배열이나 연결 리스트 혹은 트리로도 구현될 수 있음



ADT의 서로 다른 구현의 장단점을 아는
것 또한 이 과목의 학습목표의 하나이다.

Why C ?

- 프로그래밍에 대한 교육을 C언어로 시작하는 것에 대해서는 많은 논쟁이 있다.
- C언어는 고정되어 있지 않고 계속 진화한다.
- 이 과목은 C언어에 대한 것이 아니라 프로그래밍에 대한 것이다.