

자료구조 및 실습

과목소개

- 교수:

- 권 오 흠 (kwonlecture@gmail.com)
- A13동 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

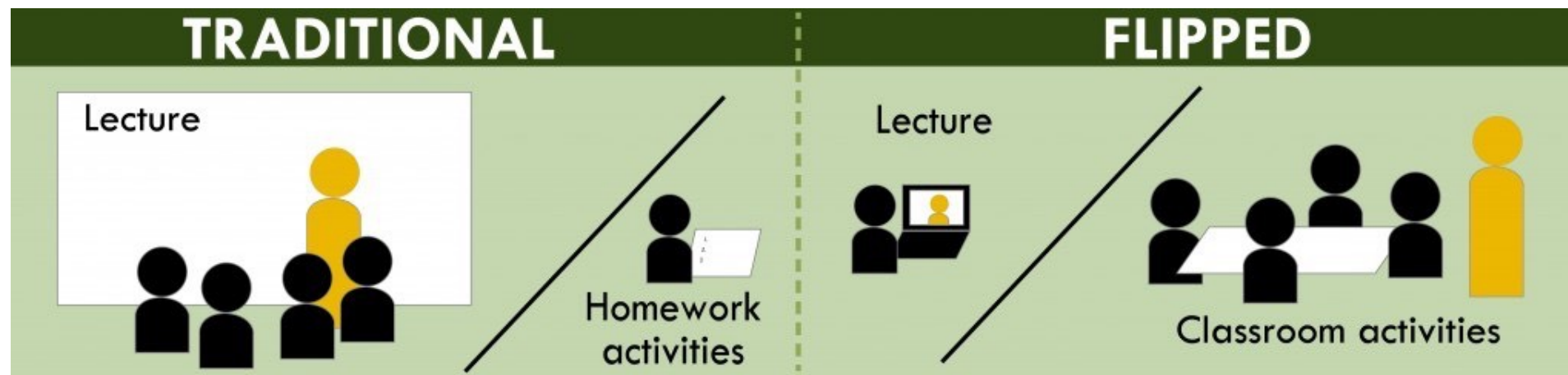
- 사용언어: C, C++

- 수업 및 실습: Flipped Class 방식

- 강의 홈페이지

- 부경 LMS

Flipped Class



- 주 단위로 강의 동영상
- 주 단위로 **프로그래밍 과제**와 **연습문제**가 주어짐
- 매주 첫 번째 2시간 수업에서는 group activity(팀별 연습문제 풀이)
 - 팀은 매 시간마다 즉흥적으로 구성, 3~4인, 매주 팀구성이 달라도 됨
- 매주 두 번째 2시간수업에서는 개별적으로 **프로그래밍 과제**
 - 원칙적으로 수업시간 중에 완료·채점

● 성적 평가 기준

평가기준	반영비율	평가방법
시험	50%	<ul style="list-style-type: none"> • 중간 및 기말 시험 (Open Book 실기 시험)
프로그래밍 실습	40%	<ul style="list-style-type: none"> • 매 주 단위로 실습과제 제출
출석, 수업참여도	10%↑	<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

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

주	주제	개요
9	스택	스택 및 큐 ADT의 개 념, 구현, 응용
10	스택	
11	큐	
12	큐	용용
13	순환	
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?**

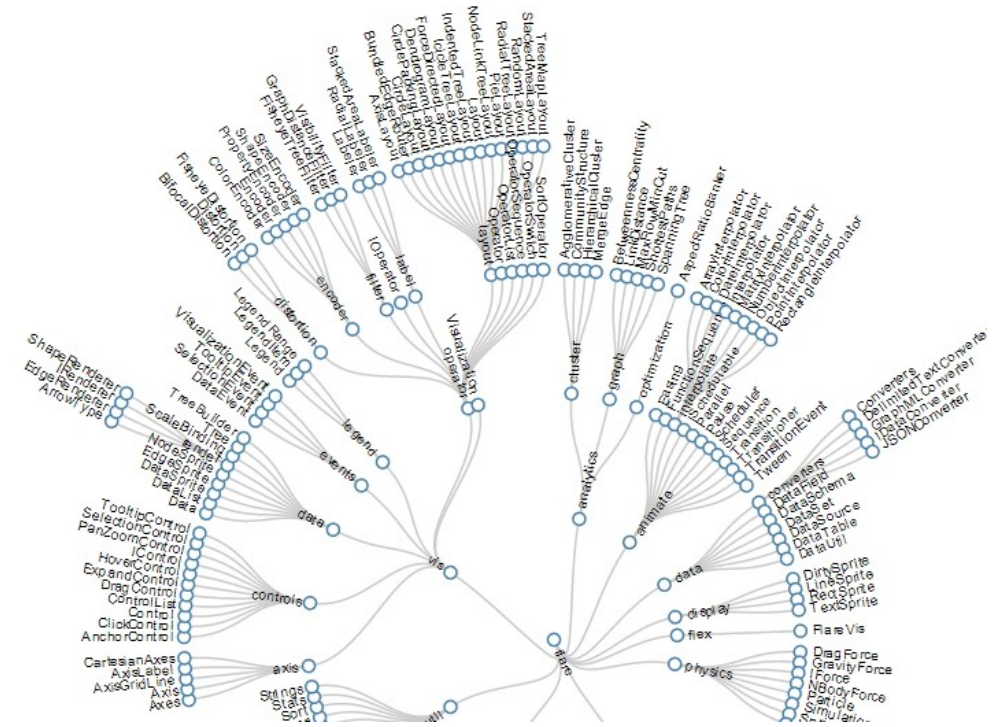


더 복잡하고 효율적인 방법들도 있음

- binary search tree
- hash table
- ...

다른 자료구조들

- 데이터 자체가 복잡한 구조를 가지는 경우가 있다.
- 어떤 기관의 조직도, 마인드 맵 등: **트리(tree)**
- 도로망, 통신망 등: **그래프(graph)**
- 이런 자료구조를 프로그램에서 어떻게 표현하고 다룰 것인가?

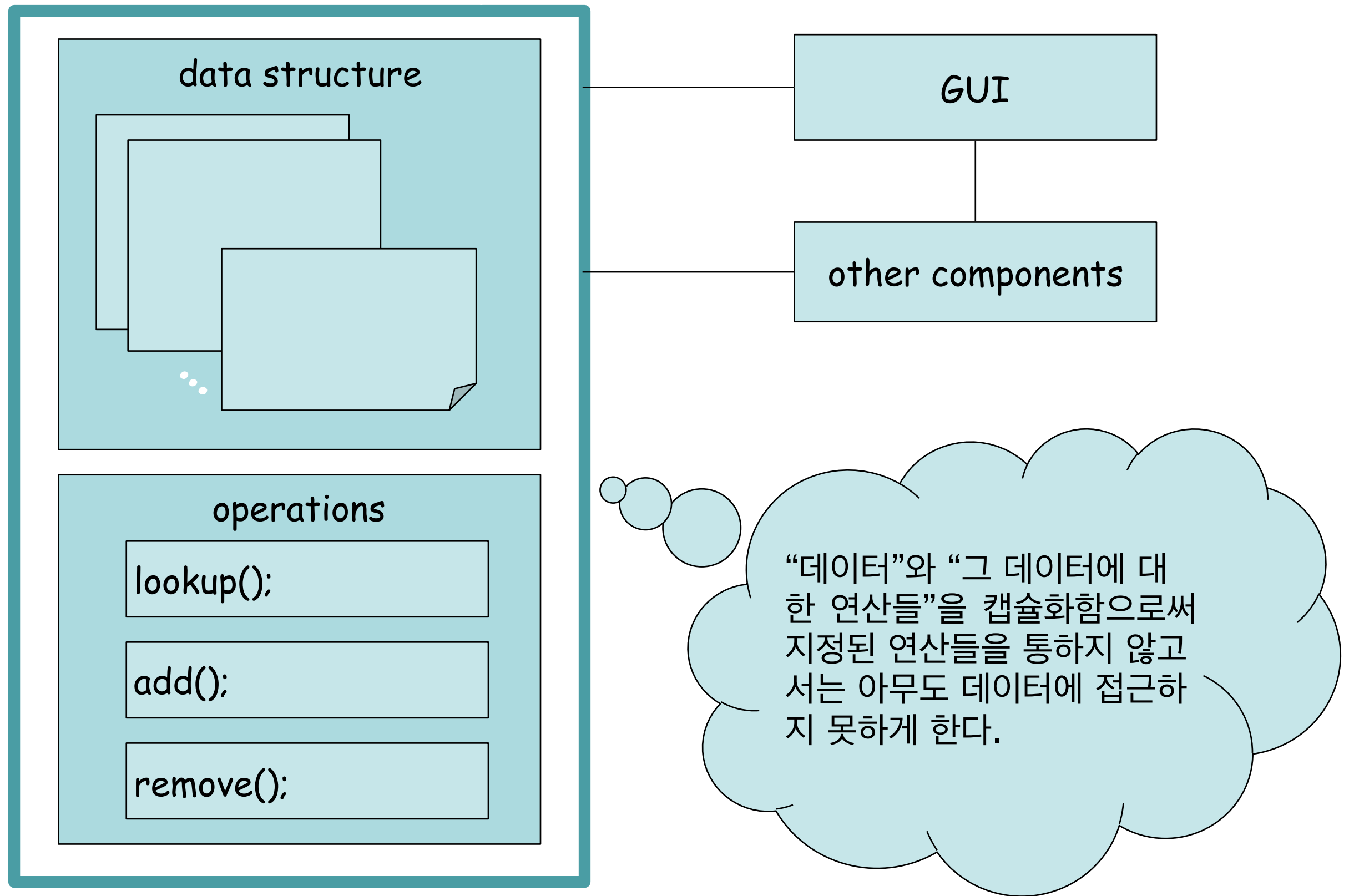


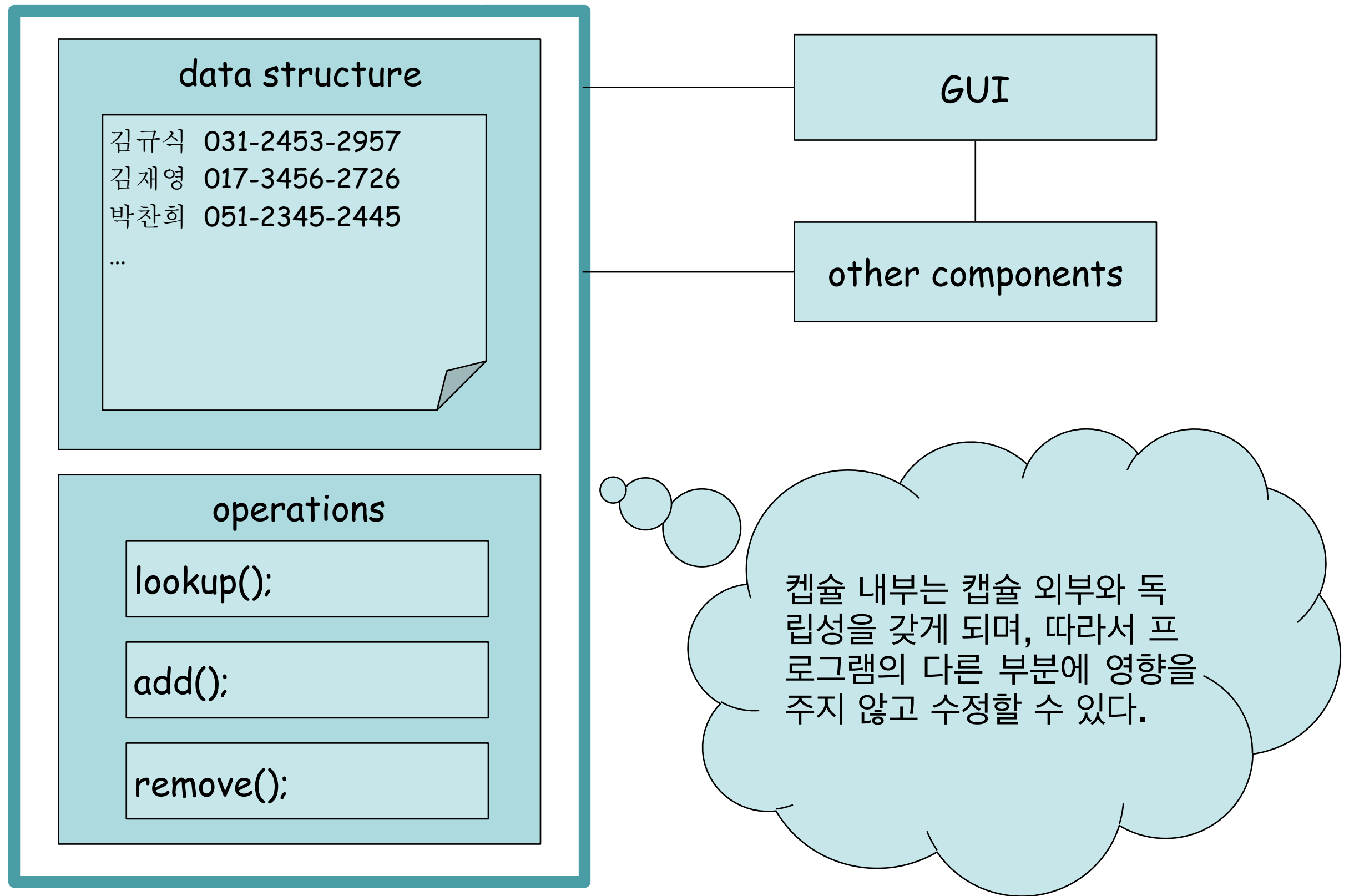
좋은 자료구조 ?

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

High Modularity

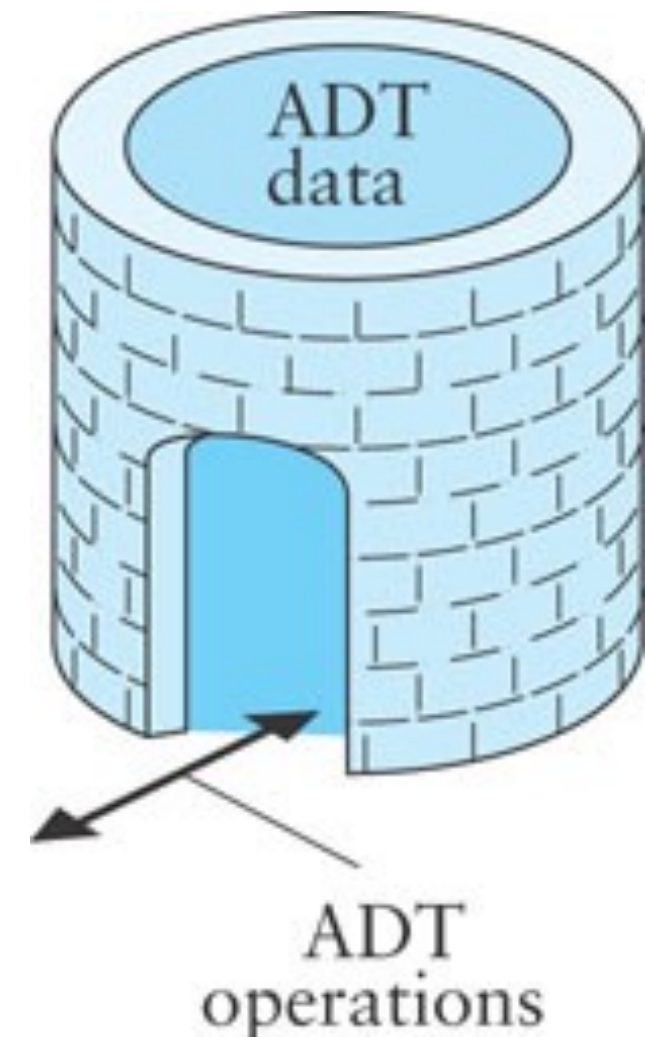
- 높은 모듈성은 프로그램을 이해, 수정, 재사용 하기 쉽게 한다.
- 높은 모듈성을 달성하는 한가지 방법은 “데이터”를 “그 데이터에 대해 작용하는 연산들”과 함께 하나의 독립된 모듈로 캡슐화(encapsulation)하는 것





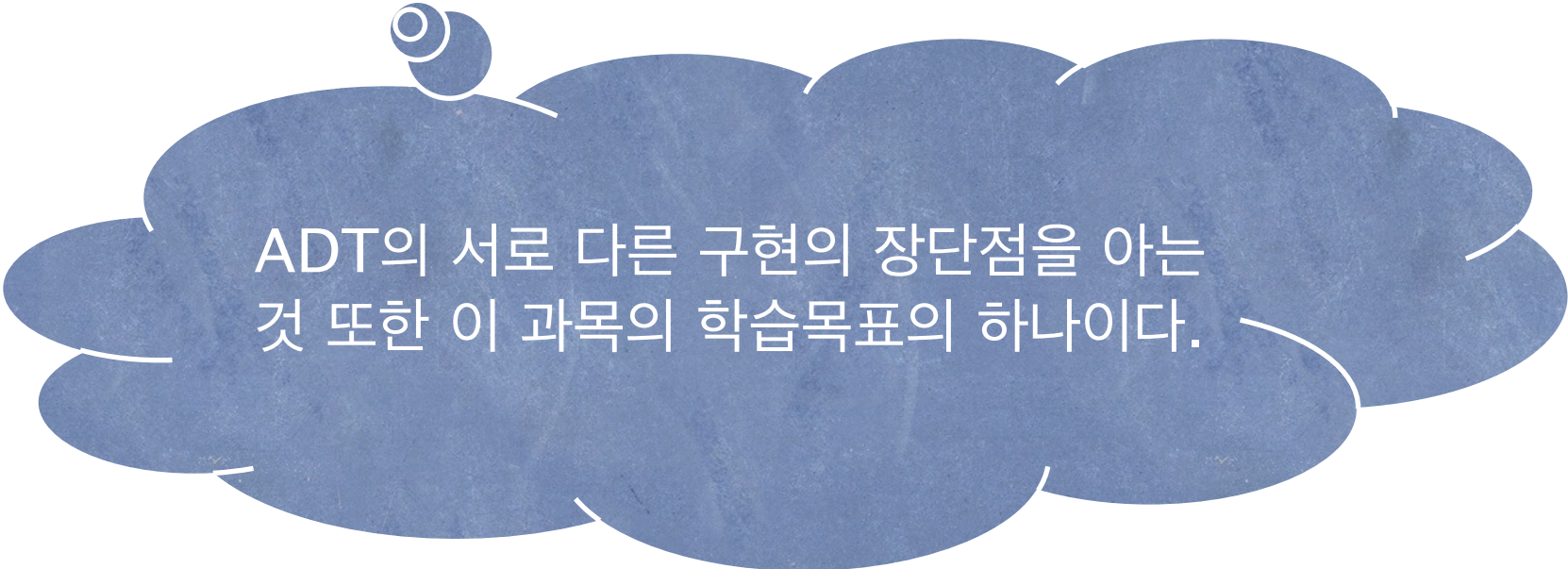
Abstract Date Type

- **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.