

DA01 – Designed AI

NEW STL IMPLEMENTING CONCURRENCY

성재모 (mentor)

김규용

구교진

이병현

조준형

송우민

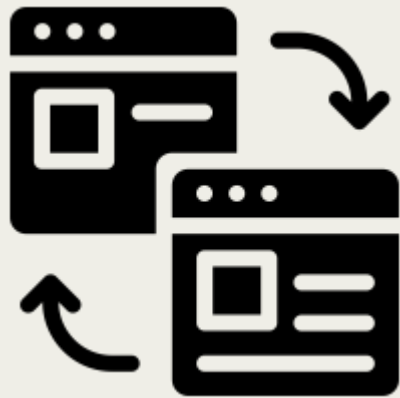
TABLE OF CONTENTS

- *Purpose for development*
- *Process and complements*
 - *Data Structures using in Project*
 - *Linked List*
 - *Binary Search Tree*
 - *Red-Black Tree*
 - *Multithreading*
 - *concept of Concurrency*
 - *Concurrency implemented in Data Structure*
- *Value of this project*

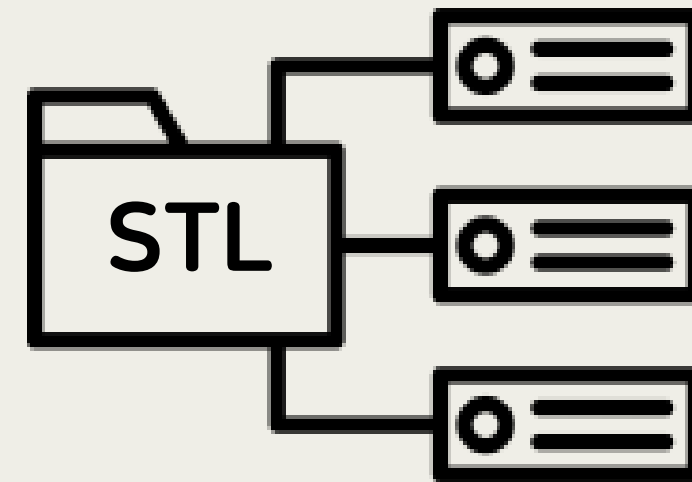
Purpose for development

Purpose for development

Data Structure with concurrency 를 활용하여 자원을 효율적으로 사용, 시간과 비용 감소



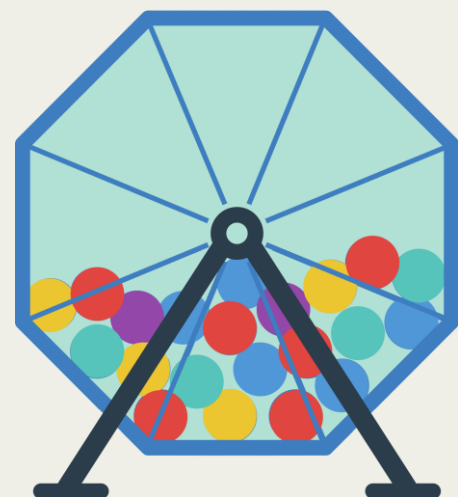
Memory pool이 공유 가능하며
thread safe하고 overhead를
최소한으로 줄이기 위한 코드 개발



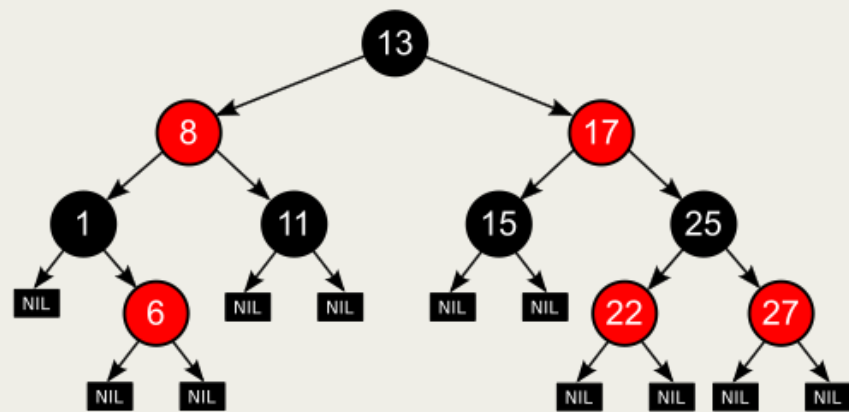
STL Data structure에 concurrency 도입,
AI 등 대용량 dataset의 처리를
편리하게 다루는 기반 기술을 갖추

Procedure

초기 데이터 생성 방법



Uniform($1, 10^6$)으로부터
40만개의 난수 추출



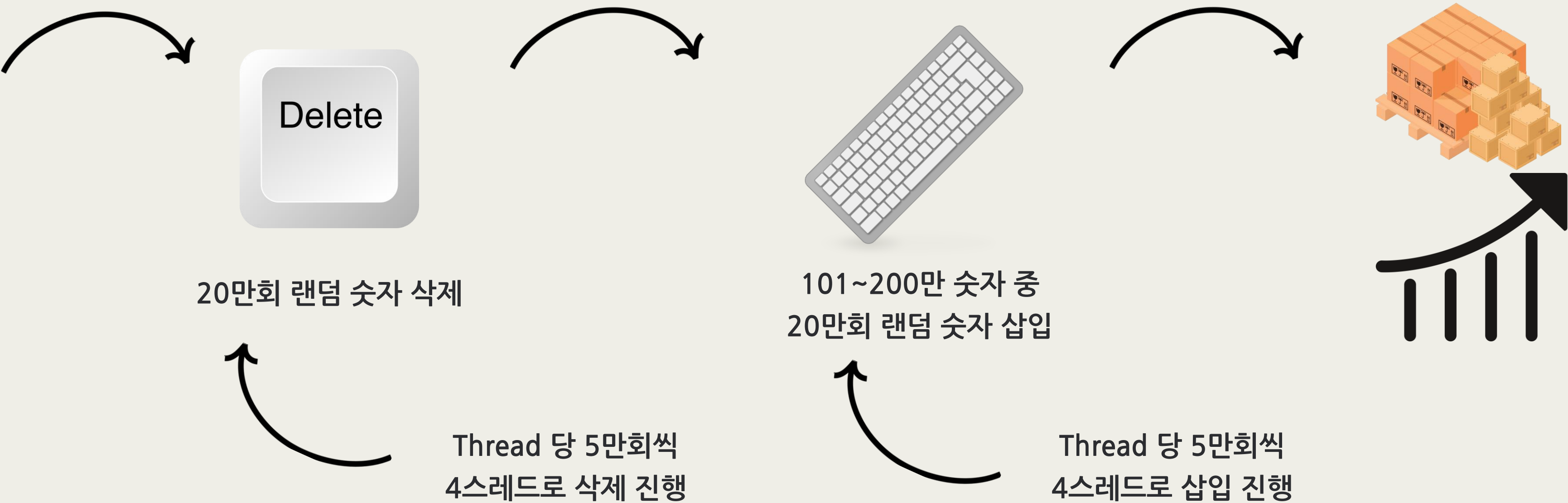
Data structure에 추출한 데이터 입력



Data structure에서 10만회 탐색

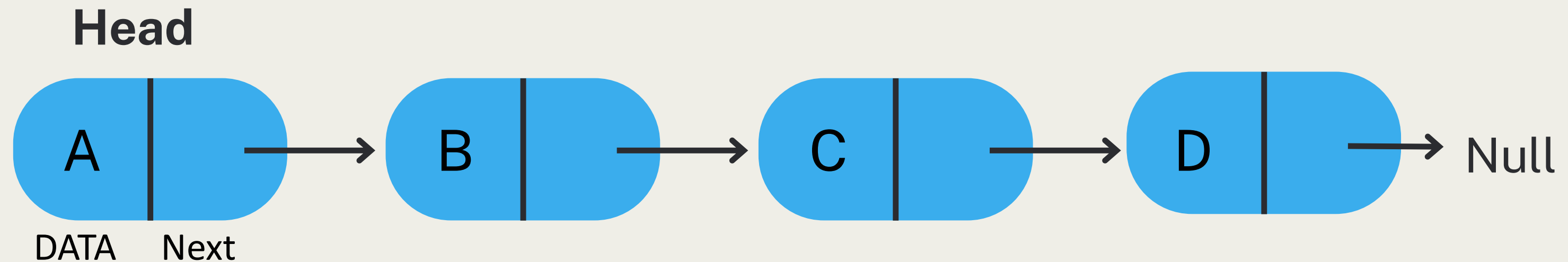
Procedure

삭제와 삽입 작업에서 Multi-Thread 도입



Process and complements

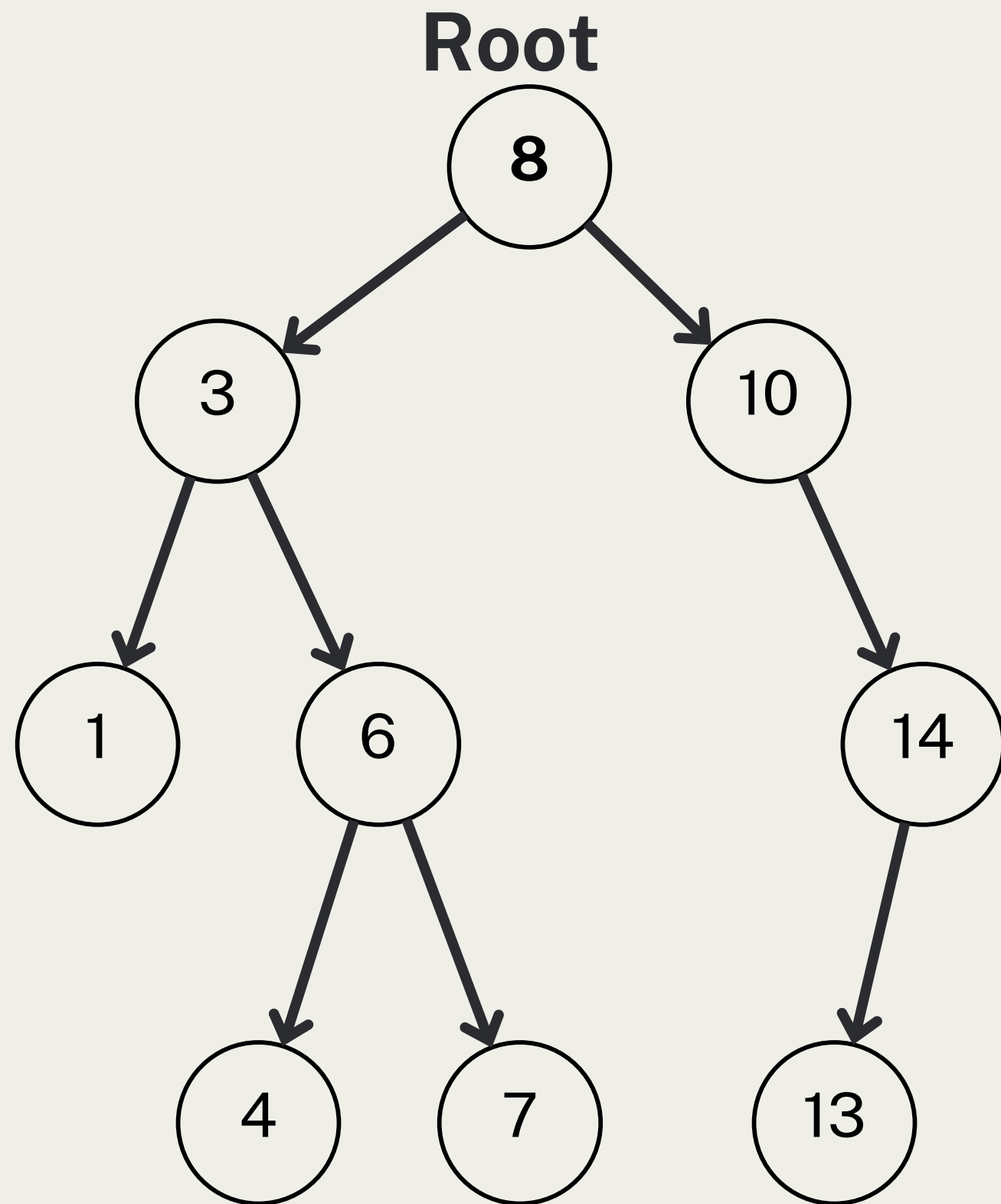
Linked List



활용

- 동적 메모리 할당(Dynamic Memory Allocation)
- 데이터 삽입과 삭제
- 데이터 구조의 유연성
- 메모리 효율성
- 재귀적 자료구조

Binary Search Tree



활용

- 데이터 검색과 정렬
- 허프만 코딩, cladogram
- 데이터의 계층 표현
- 메모리 효율성

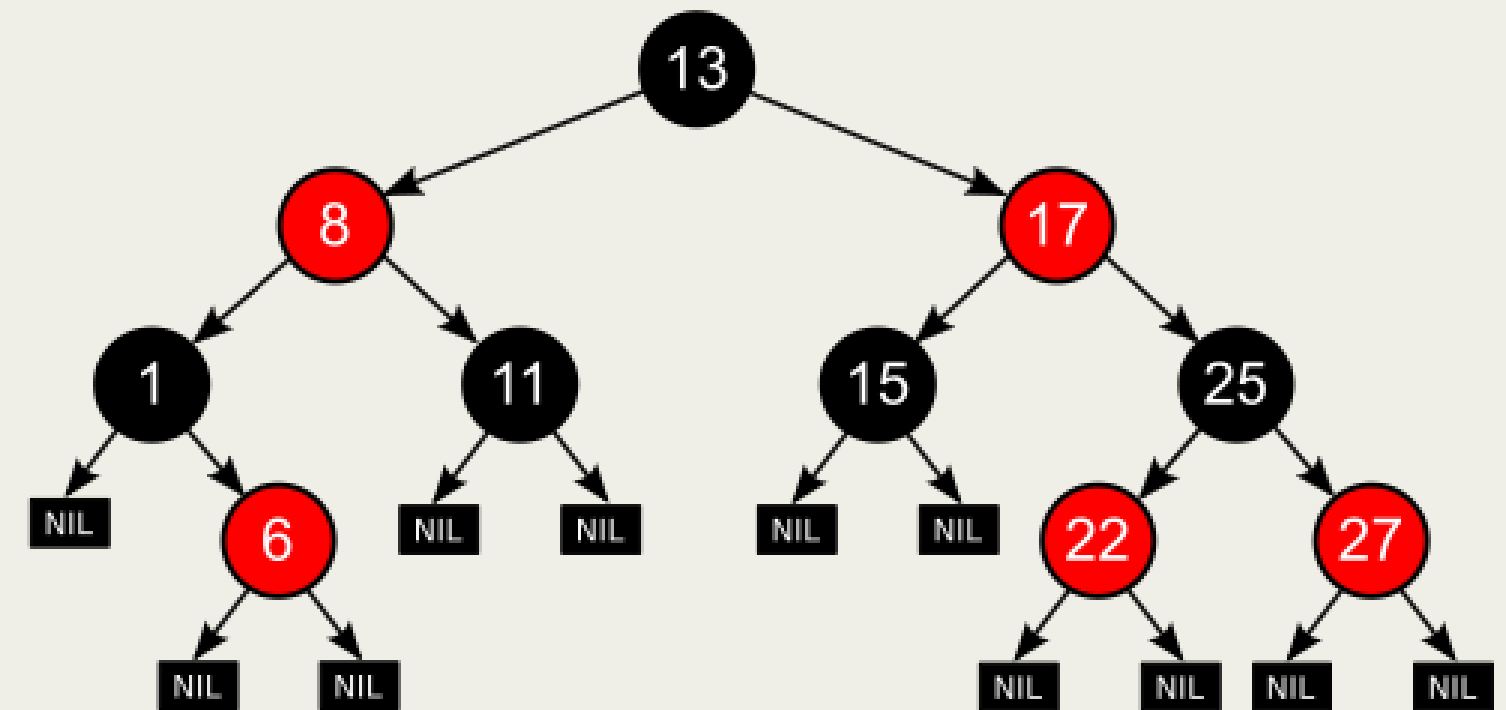
Red Black Tree

레드-블랙 트리

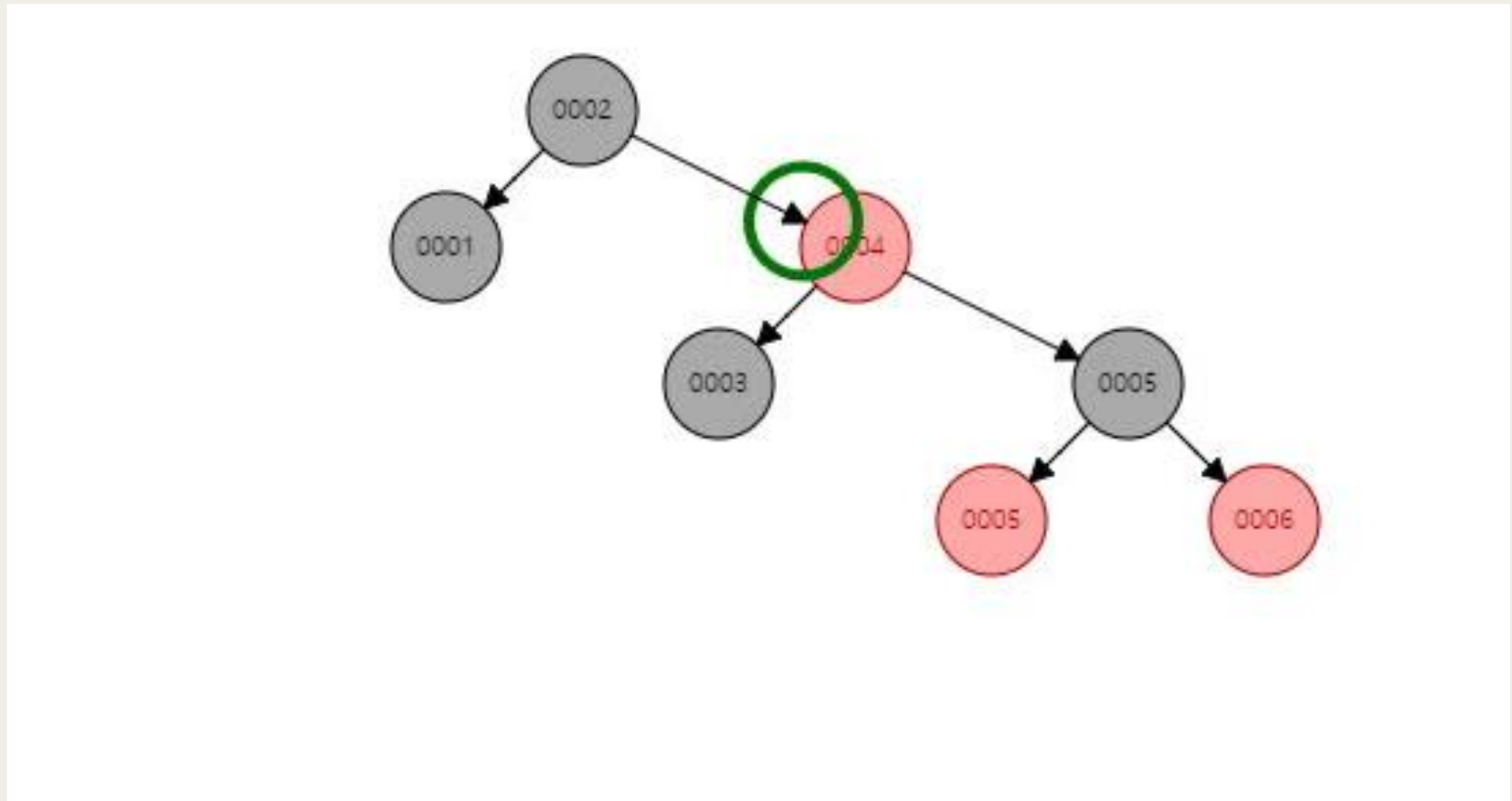
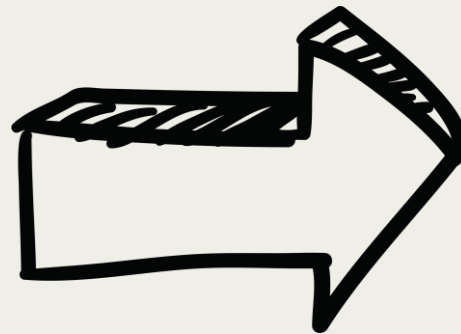
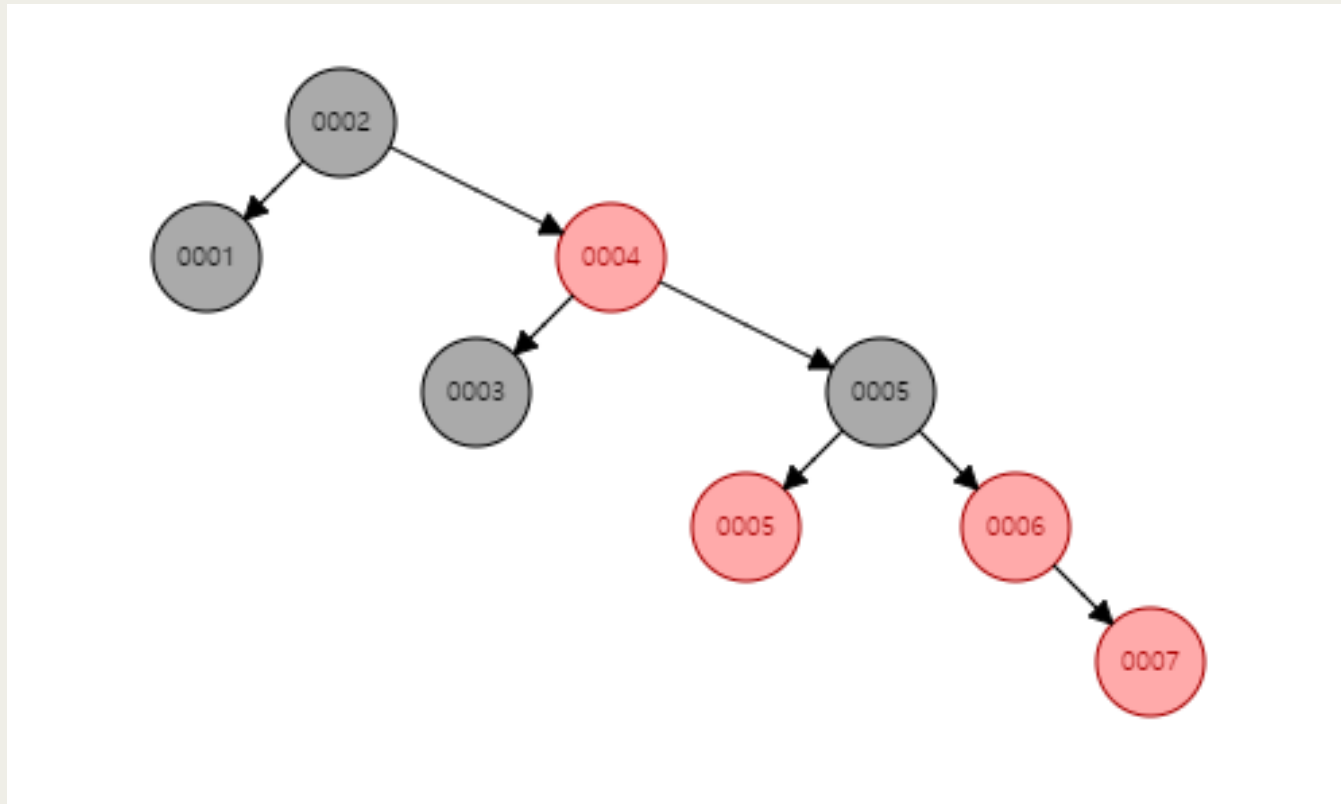
이진 탐색 트리(Binary Search Tree)의 일종으로, 균형을 유지하면서, 데이터를 저장하고 탐색하는 자료구조

레드-블랙 트리의 조건

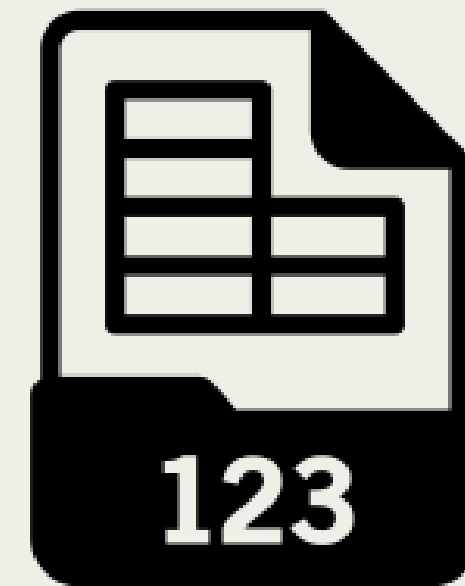
1. 노드는 레드 혹은 블랙 중의 하나이다.
2. 루트 노드는 블랙이다.
3. 모든 리프 노드들(NIL)은 블랙이다
4. 레드 노드의 자식노드 양쪽은 언제나 모두 블랙이다.
5. 어떤 노드에서 리프 노드까지의 모든 경로에는 같은 수의 블랙 노드가 있다.



Red Black Tree



Red Black Tree



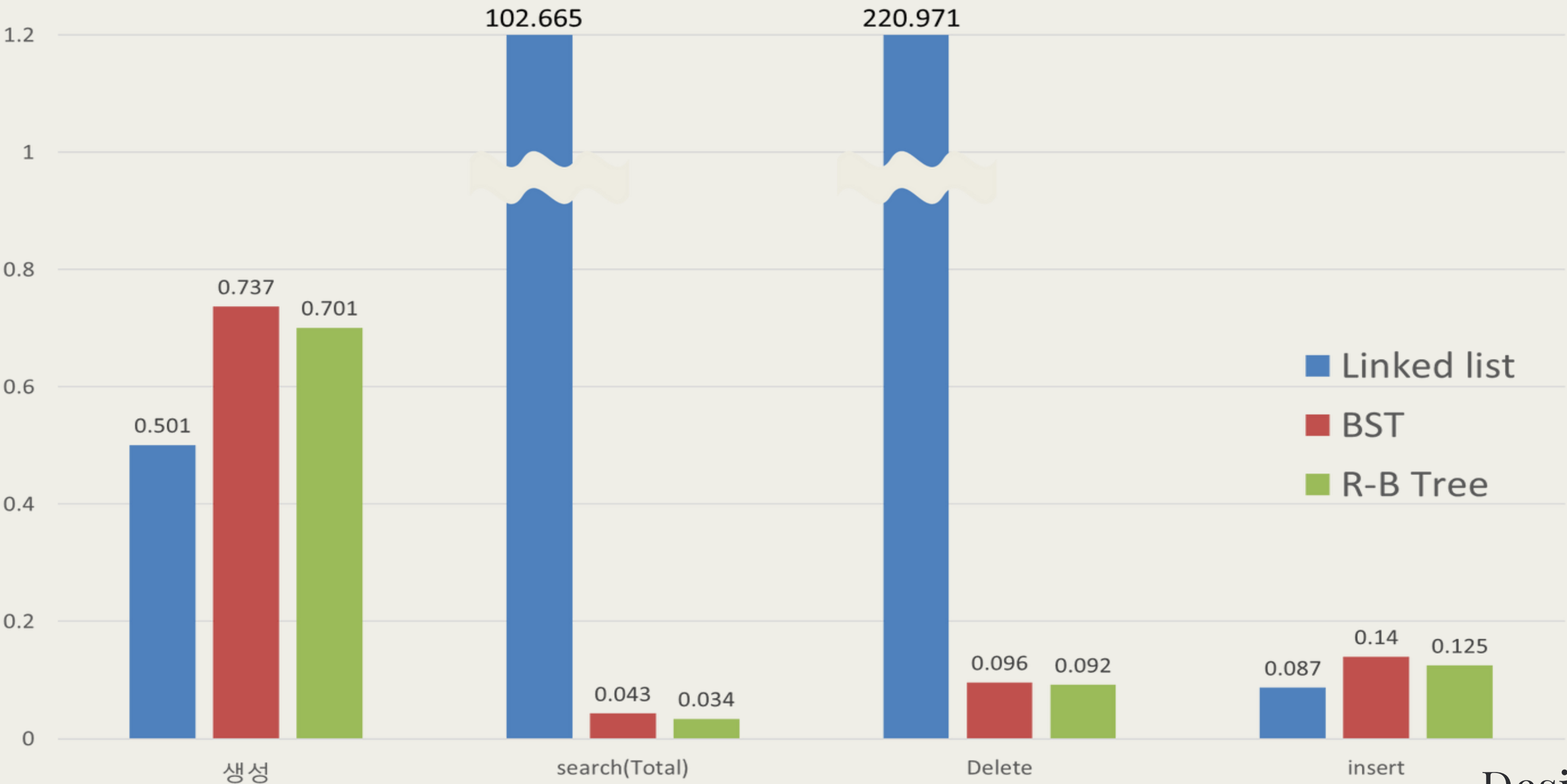
활용

- 데이터의 효율적인 저장 및 탐색
- STL 구현 - map container in Standard Template Library(STL)
- 데이터베이스 인덱스

Compare Time by Data Structure in Single Thread

각 자료구조 당

- 초기입력 데이터 수 : 40만, - 데이터 탐색횟수 : 10만, - 데이터 삭제횟수 : 20만, - 데이터 추가횟수 : 20만



Multithread

멀티스레드(Multithread)

하나의 프로세스 내에서 둘 이상의 스레드가 동시에 작업을 수행하는 방법

종류

Concurrency (동시성)	Parallelism (병렬성)
동시에 실행되는 것처럼 보이는 것	실제로 동시에 실행되는 것
논리적인 개념	물리적인 개념
싱글코어, 멀티코어에서 가능	멀티코어에서 가능

Concurrency

동시성제어(Concurrency Control)

여러 스레드가 동시에 실행되는 상황에서 데이터의 일관성과 정확성을 유지하기 위한 기술과 접근 방법

동시성 제어의 목표

- 데이터 일관성
- 데이터 무결성
- 데드락 회피
- 성능 최적화

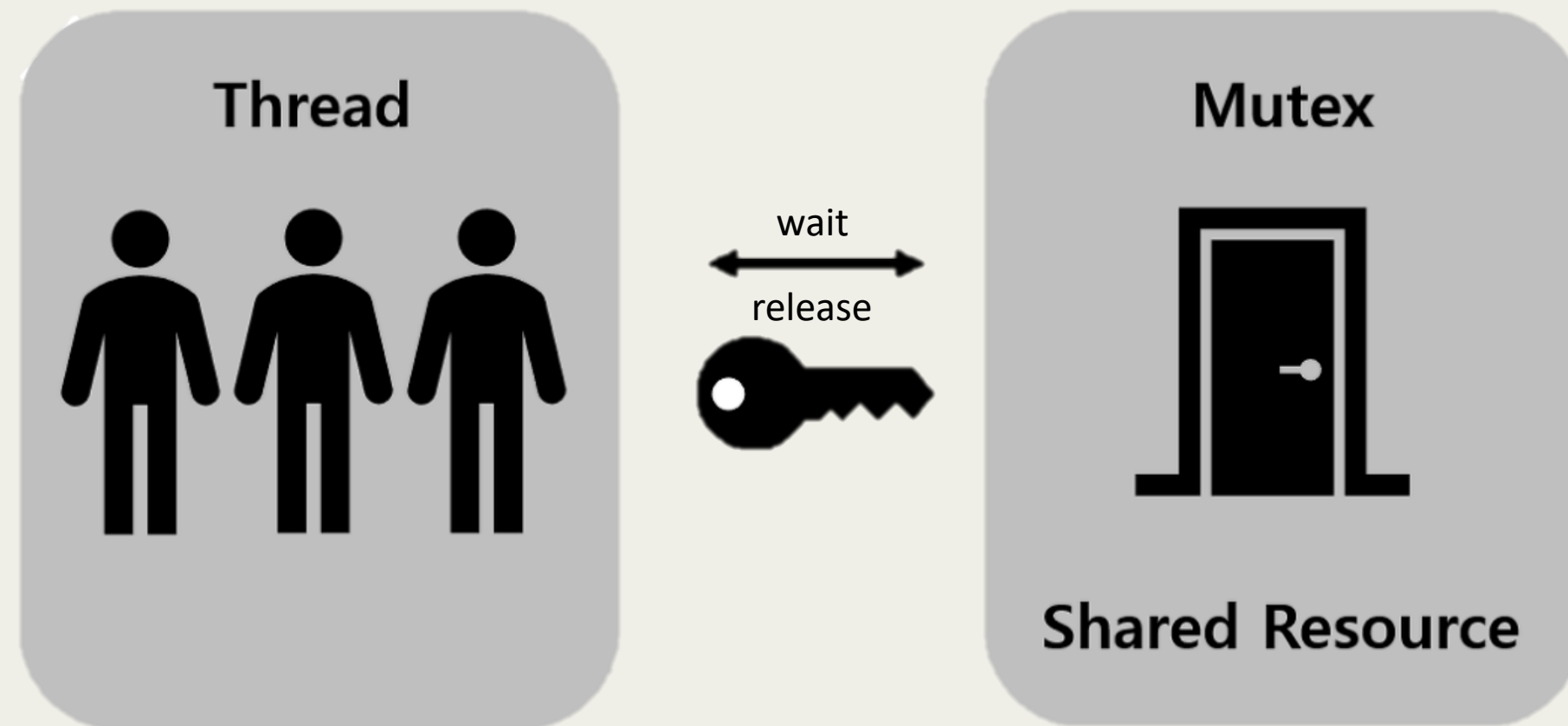
동시성 제어 기법

- Locking
- 트랜잭션
- MVCC(Multi-Version Concurrency Control)
- 동시성 제어 알고리즘

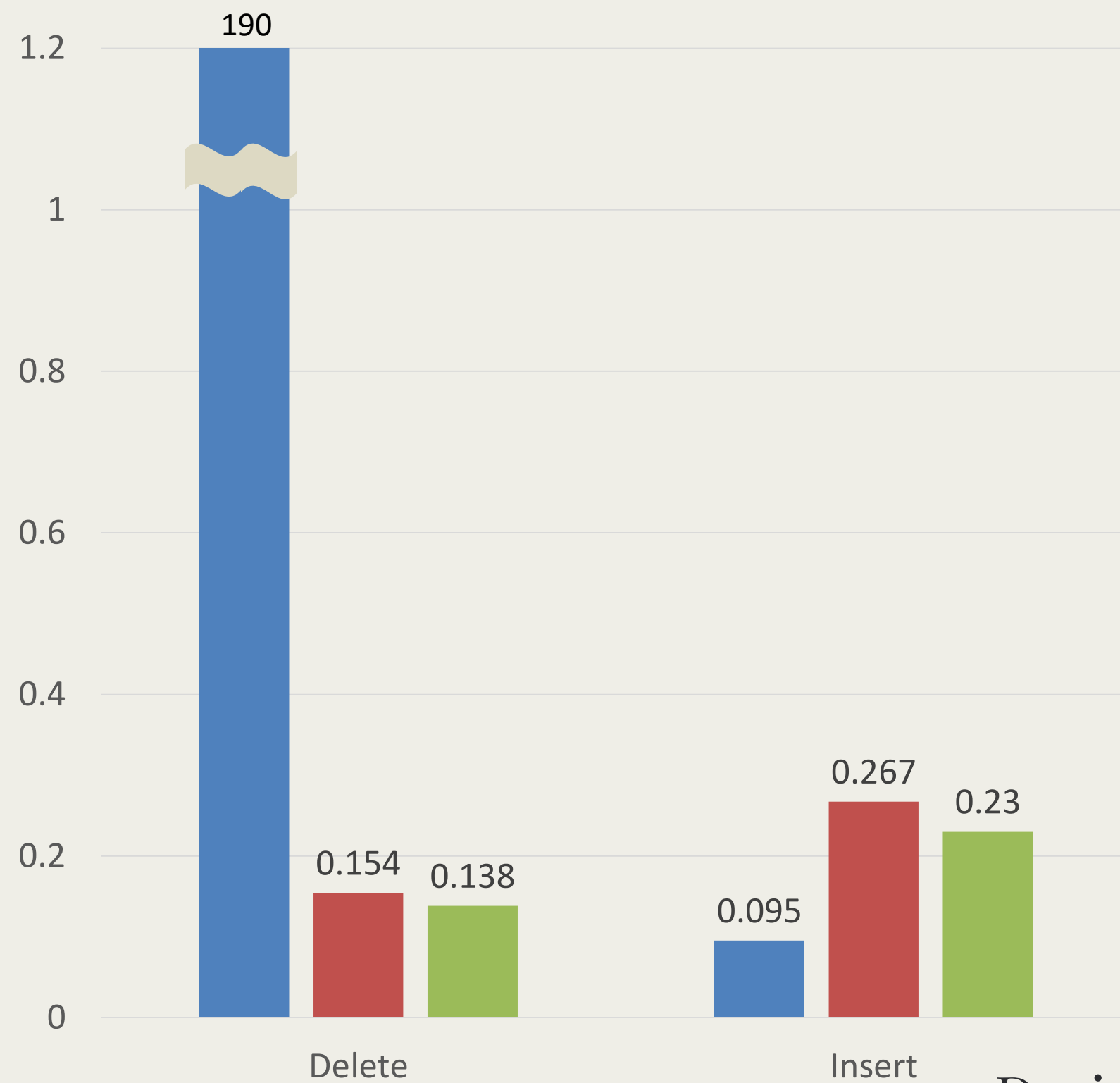
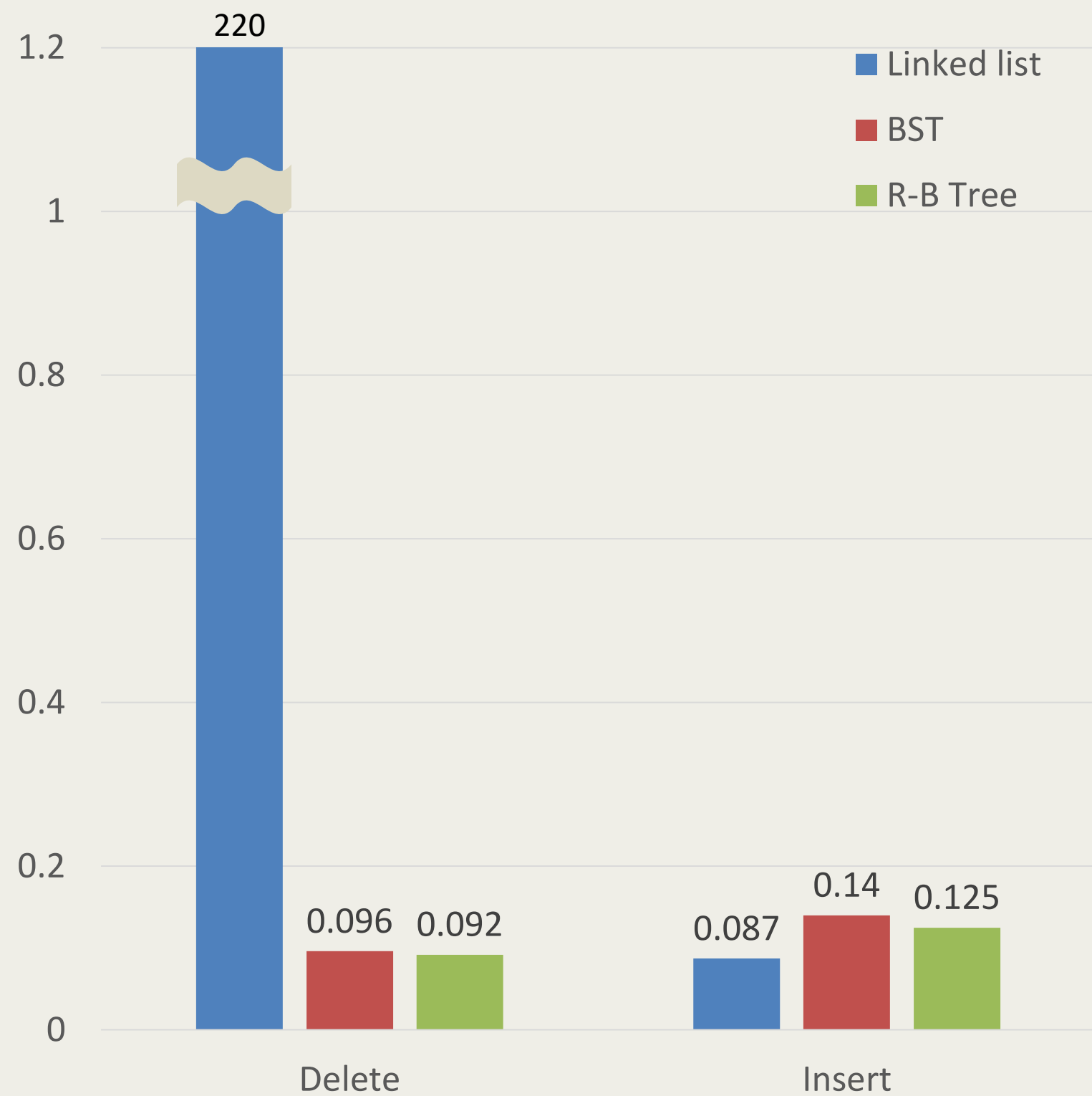
Concurrency - MUTEX

뮤텍스(MUTEX)

멀티스레드를 활용할 때 상호 배제를 위한 동기화 기법.
여러 스레드가 공유 자원에 동시 접근하지 못하도록 제어



Compare Time Between Single Thread and Multi thread



Value of this project

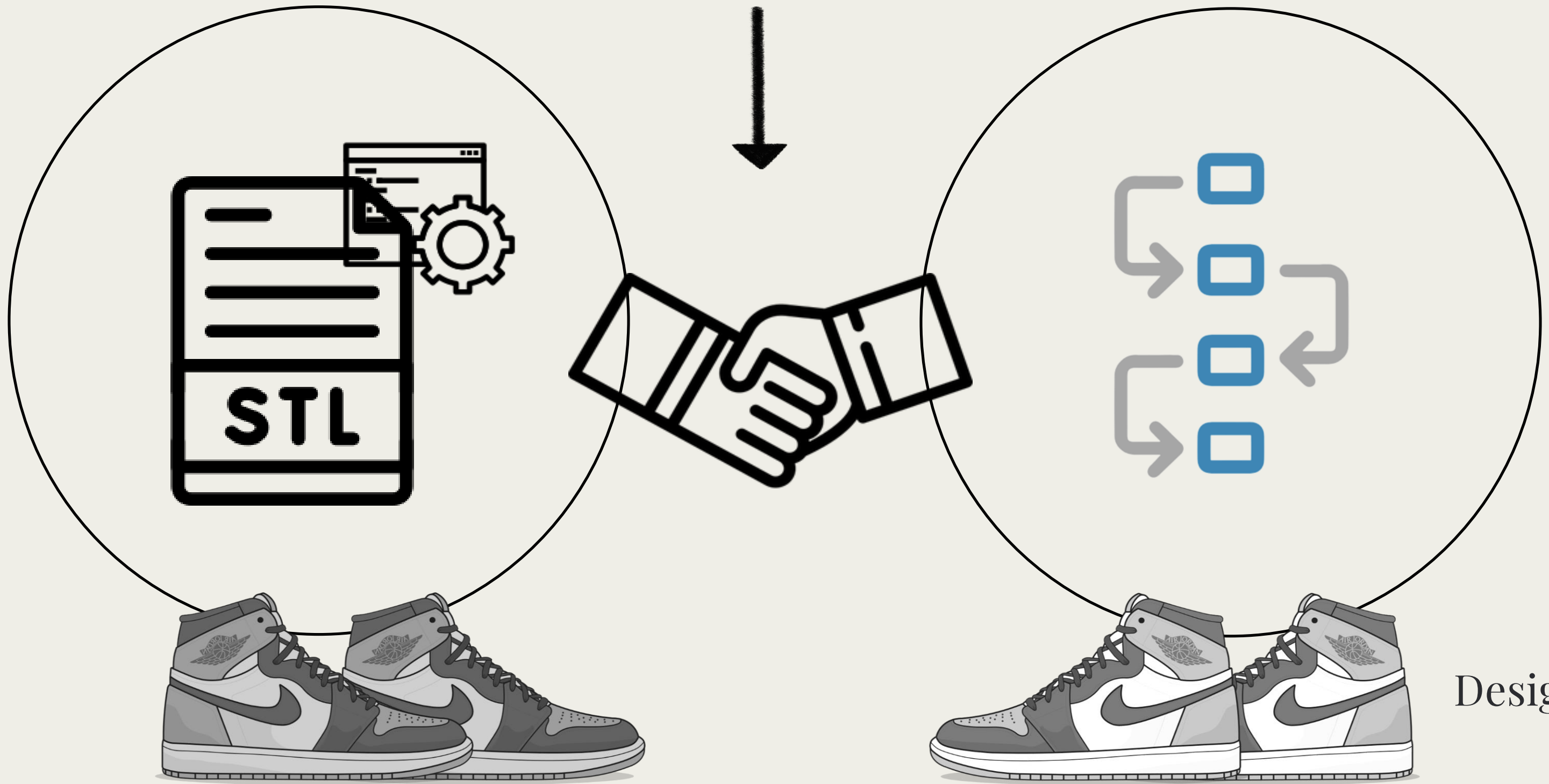
Value of this project

AI model 개발 적용시 자원을 효율적으로 활용



Value of this project

Standard Template Library(STL)에 concurrency 도입



Thank you!

FOR LISTENING TO OUR PROJECT

성재모 (mentor)

김규용

구교진

이병현

조준형

송우민