

해시 테이블 (Hash Table)

practice 자료구조 · 2024. 9. 26. 13:44

•공부할 내용

1.해시 함수란?

2.해시 테이블이란?

해시 충돌은 왜 생기고, 어떻게 해결하는가?

•임의의 입력값을 받아, 고정된 길이의 해시값을 출력하는 함수

•이때 해시값은 임의의 숫자 혹은 문자열

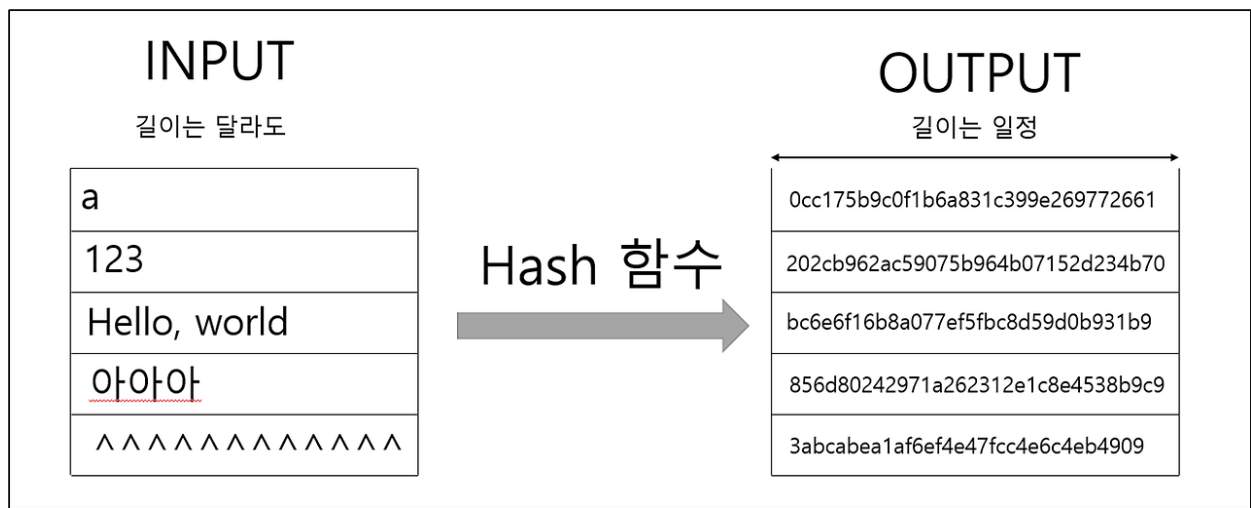
hash 함수 :

input : 아무거나 (숫자, 문자, 아무거나)

output: 숫자만 또는 문자 또는 섞여서 여러가지 해쉬값

hash함수: hash값으로 바꿔주는 함수 (숫자 , 문자로 바꿔주는 함수)

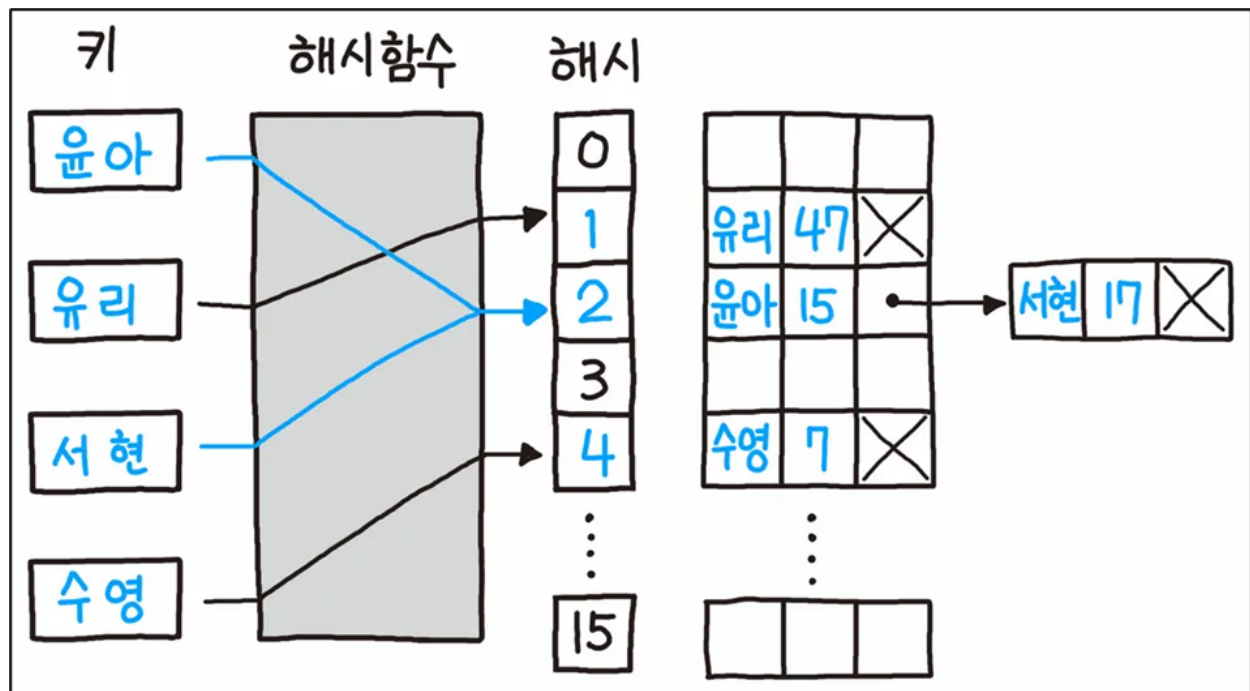
한방향: input에서 output은 가능, output에서 input은 불가능



- 다양한 해시 함수가 존재 (MD5, SHA, 등)
- 해시 값을 통해 입력값을 유추하는 것은 (거의) 불가능
- 이러한 해시 함수를 통해 효과적으로 데이터를 저장하고 불러오는 것이 가능
- Key와 Value 쌍이 있다고 할때, 주어진 Key에 대한 Value를 효율적으로 찾고 싶음
- 예를 들자면, 학생의 이름이 Key, 시험 점수를 Value라고 할때, 주어진 학생의 점수를 빠르게 찾고 싶음
- 가장 단순한 방법: 모든 Key-Value 쌍을 저장
- [(김지훈, 60), (김동현, 50), (김현우, 70), ...]
- 하지만 이 방법은 $O(n)$ 의 시간 복잡도를 가짐 (매우 비효율적)
- 보다 나은 방법: 해시 테이블
-
- List (혹은 Array)에서, 주어진 Index의 값을 찾는 것은 $O(1)$ 의 시간 복잡도를 가짐 (매우 효율적)

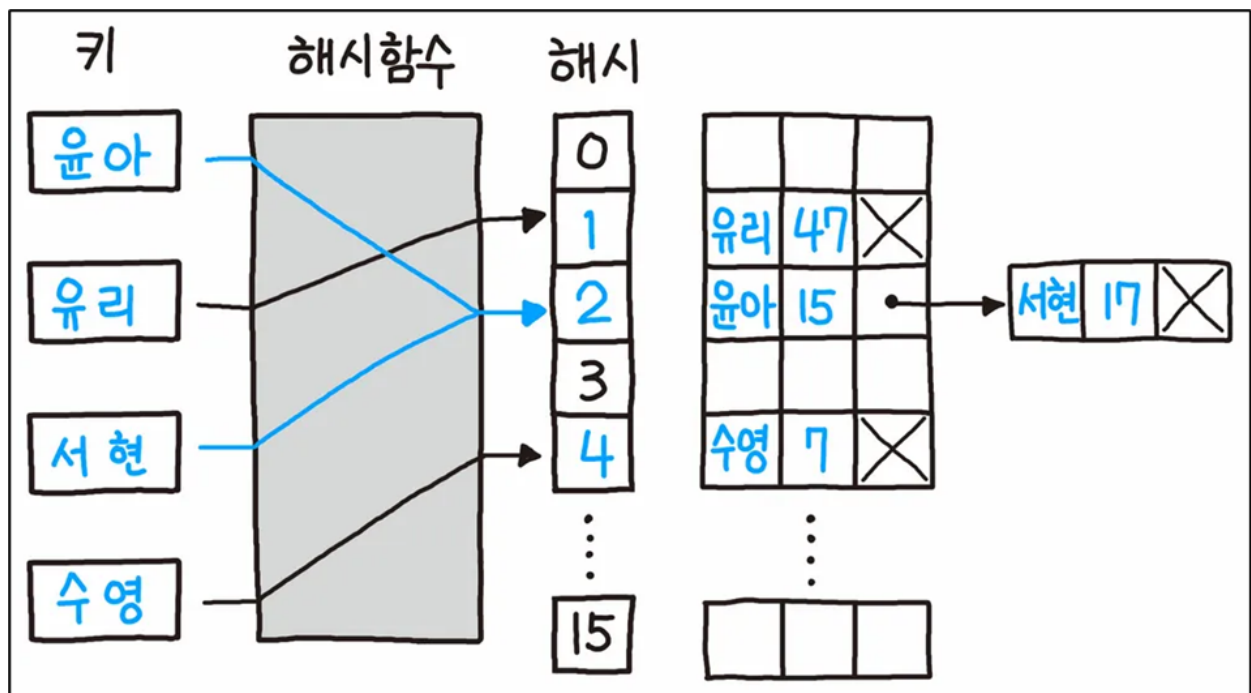
•또한, 해시 함수를 쓰면 임의의 Key를 숫자로 변환하는 것이 가능

주어진 Key에 대한 Value를 해시 함수로 변환된 숫자 Index에 저장하고 사용하자



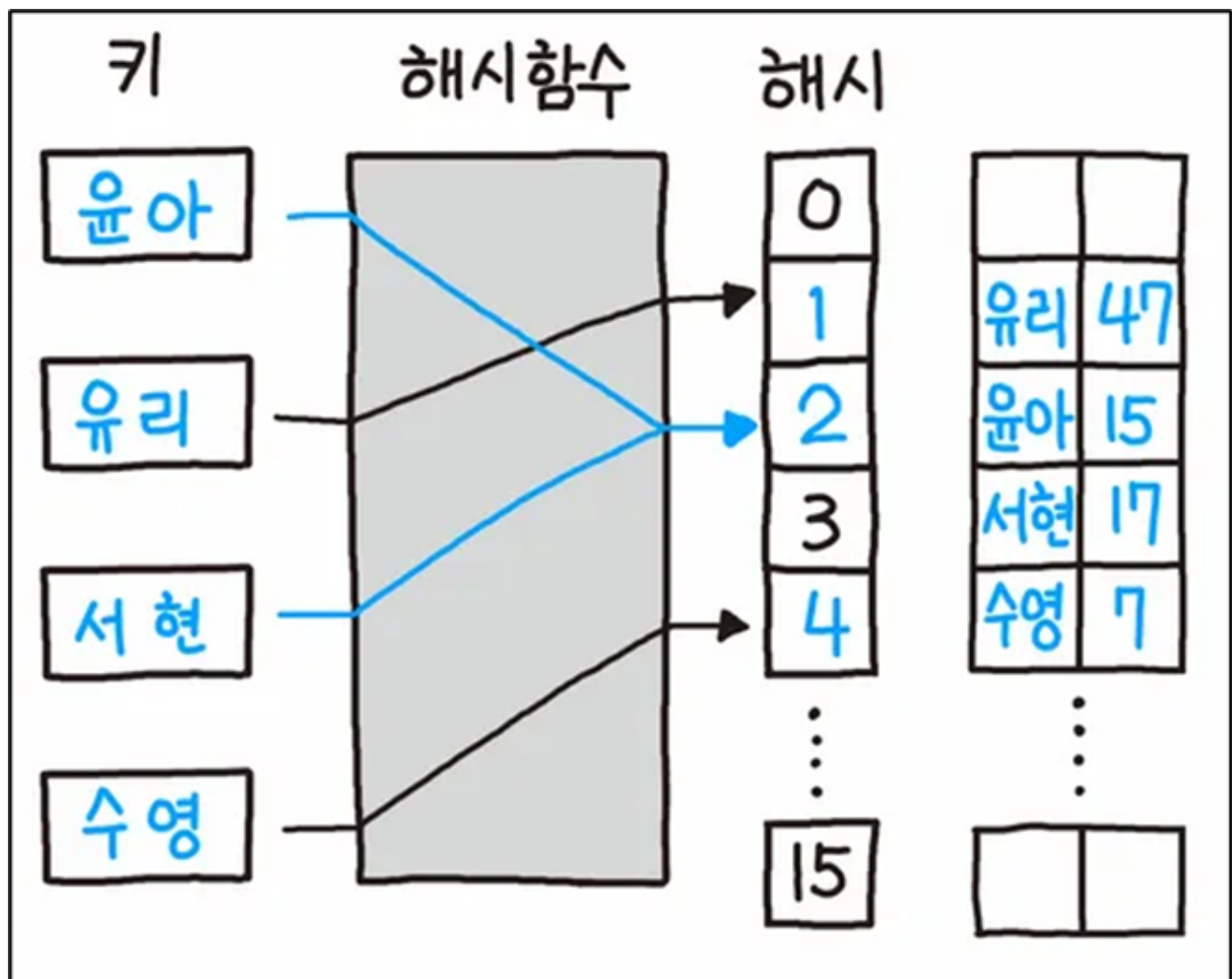
문제 1 – 해시 충돌

- 이때 여러 개의 Key가 같은 숫자 (Index)를 가질 수 있음 (충돌)
- 이를 해결하는 첫번째 방법: 체이닝 (Chaining)
- 값이 충돌하면, 단순히 Linked List로 새 Value를 저장하는 것



•두번째 방법: Open Addressing

•충돌이 발생할 경우, 주변에 있는 빈 슬롯을 찾아 Value를 저장하는 것



문제 1 – 모범 정답

- 해시 테이블은 키를 해시 함수로 변환하여 값을 저장하는 방법입니다.
- 해결 방법에는 주로 체이닝(Chaining)과 개방 주소법(Open Addressing)이 있습니다.
- 체이닝은 충돌이 발생한 키를 링크드 리스트로 저장하는 방법이며,
Open Addressing
은 빈 슬롯을 찾아서 저장하는 방법입니다

문제 1 – 추가 문제

- Open Addressing 방식은 테이블에 이미 많은 Value들이 저장되어 있을 경우, 처리 시간이 크게 늘어납니다. 왜 이런 현상이 발생할까요?

답변

- Open Addressing에서, 해시 충돌이 발생하면, 주변에 있는 빈 슬롯을 찾아 Value를 저장하게 됩니다.

- 하지만 만약 테이블에 이미 많은 값들이 저장되어 있을 경우, 빈 슬롯을 찾는 것이 매우 오래 걸리게 됩니다.

- 따라서 해시 테이블에 빈 공간이 별로 없는 경우, 큰 테이블로 옮기는 것이 (일반적으로) 필요합니다.

문제 2

- 이진 탐색 트리 (Binary Search Tree)의 주요 특징과 기본 연산에 대해 설명해 주세요.

- 공부할 내용

- 1.왜 이진 탐색 트리가 필요한가?
- 2.이진 탐색 트리란?
- 3.이진 탐색 트리에는 어떤 기본 연산들이 있는가?

이진 탐색 트리란?

- 하지만 값들이 이진 탐색 트리로 저장되어 있다면, 보다 효율적으로 이러한 작업을 진행할 수 있음 (모든 값을 다 확인할 필요가 없음)

- 특정 값 검색하기:

- Root부터 시작해서, 주어진 값이 작으면 왼쪽, 크면 오른쪽으로 이동하면 됨

- 특정 값 삽입하기:

- 비슷하게, Root부터 시작해 길을 따라가다 끝에서 새로운 노드 만들기

- 특정 값 삭제하기

- 길을 따라가다, 주어진 값을 만나면 삭제하면 됨

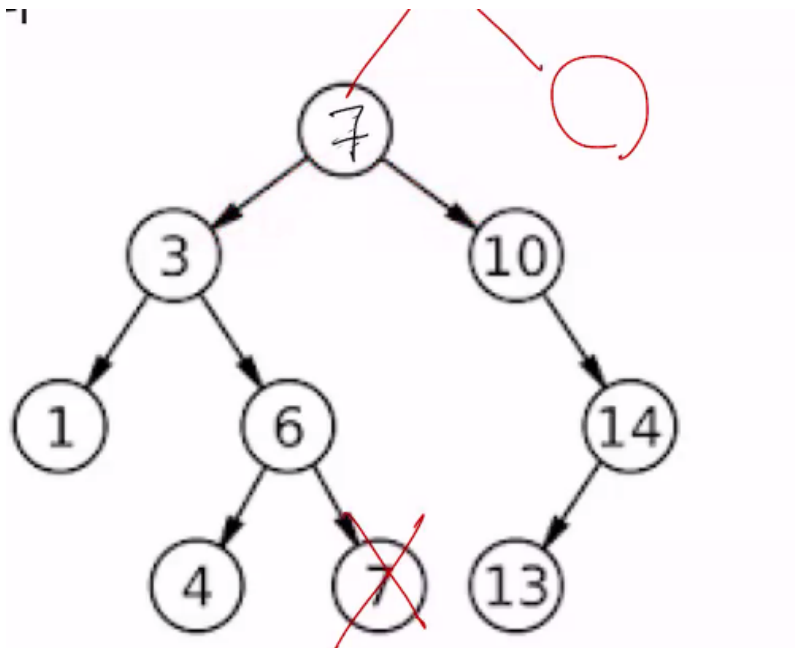
- 다만 이때 세가지 경우가 있음

-

- 1.주어진 값 노드가 자식이 없을 경우 -> 단순히 삭제

- 2.주어진 값 노드에 자식이 하나만 있을 경우 -> 자식을 현재 위치로 바꾸기

- 3.만약 주어진 값 노드에 두개의 자식이 있을 경우 -> 왼쪽 자식 트리에서 가장 큰 값과 교체



공감

구독하기

'practice_자료구조' 카테고리의 다른 글

P / NP (0)

2024.09.30

crossentropy (1)

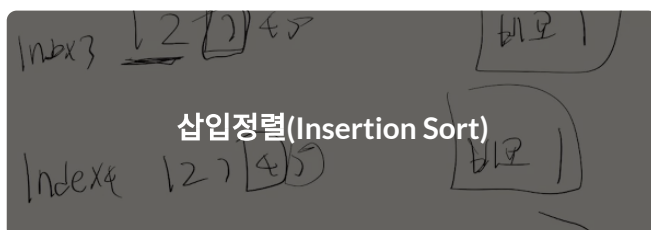
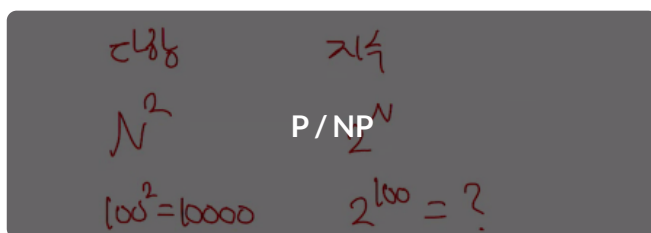
2024.09.20

삽입정렬(Insertion Sort) (0)

2024.09.15

관련글

관련글 더보기



자연어(NLP)

네이쳐2024 님의 블로그입니다.



구독하기 +

댓글 0



이름

비밀번호

내용을 입력하세요.



댓글
등록