

## <용어체크>

### B-트리

이진 트리를 확장해 하나의 노드가 가질 수 있는 자식 노드의 최대 숫자가 2보다 큰 트리 구조이며 하나의 노드에 많은 수의 데이터가 배치되어 대용량의 파일을 효율적으로 검색하고 갱신하기 위해 고안된 자료구조이다.

### B-트리에서 삽입

키  $x$ 를 삽입할 리프 노드  $r$ 을 찾아 노드  $r$ 에 공간의 여유가 있으면 키를 삽입하고 끝내고, 노드  $r$ 에 여유가 없으면 형제 노드를 살펴 공간의 여유가 있으면 형제 노드에 키를 하나 넘기고 끝내지만 형제 노드에 여유가 없으면 가운데 키를 부모 노드로 넘기고 노드를 두 개로 분리한다.

### B-트리에서 삭제

삭제할 키  $x$ 를 갖고 있는 노드를 찾아 이 노드가 리프 노드가 아니면  $x$ 의 직후 원소  $y$ 를 가진 리프 노드  $r$ 을 찾아  $x$ 와  $y$ 를 바꾸고 리프 노드에서  $x$ 를 제거하며 만약  $x$  제거 후 노드에 언더플로우가 발생하면 이를 적절히 해소한다.

## <학습내용>

### B-트리

#### B-트리에서 삽입

#### B-트리에서 삭제

## <학습목표>

B-트리의 도입 동기를 이해하고 특징을 설명할 수 있다.

B-트리에서 삽입 작업의 원리를 설명할 수 있다.

B-트리에서 삭제 작업의 원리를 설명할 수 있다.

Q. 방대한 양의 저장된 자료를 검색해야 하는 경우 검색어와 자료를 일일이 비교하는 방식은 비효율적입니다. 이럴 때 사용할 수 있는 검색 방법에는 어떤 것이 있을까요?

: 많은 데이터들을 저장하는 데이터베이스나 파일 시스템은 대량의 데이터를 효율적으로 검색하는 방법이 필요합니다. B-트리는 대용량의 파일을 효율적으로 검색하고 갱신하기 위해 고안된 트리 형태의 자료구조입니다. 또한 이진 트리를 확장해 하나의 노드가 가질 수 있는 자식 노드의 최대 숫자가 2보다 큰 트리 구조이기도 합니다. B-트리는 자료를 정렬된 상태로 보관하고 삽입 및 삭제를 대수 시간으로 할 수 있습니다.

## B-트리

이진 트리를 확장해 하나의 노드가 가질 수 있는 자식 노드의 최대 숫자가 2보다 큰 트리 구조이다.

하나의 노드에 많은 수의 데이터가 배치된다.

대용량의 파일을 효율적으로 검색하고 갱신하기 위해 고안된 자료구조이다.

최악의 경우에도  $O(\log n)$ 의 검색 성능을 보인다.

## B-트리에서 삽입

$x$ 를 삽입할 리프 노트  $r$ 을 찾는다.

노드  $r$ 에 공간의 여유가 있으면 키를 삽입하고 끝낸다.

노드  $r$ 에 여유가 없으면 형제 노드를 살펴 공간의 여유가 있으면 형제 노드에 키를 하나 넘기고 끝낸다.

형제 노드에 여유가 없으면 가운데 키를 부모 노드로 넘기고 노드를 두 개로 분리한다.

분리 작업은 부모 노드에서의 삽입 작업을 포함한다.

## B-트리에서 삭제

$x$ 를 키로 갖고 있는 노드를 찾는다.

이 노드가 리프 노드가 아니면  $x$ 의 직후 원소  $y$ 를 가진 리프 노드  $r$ 을 찾아  $x$ 와  $y$ 를 바꾼다.

리프 노드에서  $x$ 를 제거한다.

$x$  제거 후 노드에 언더플로우가 발생하면 적절히 해소한다.