Database 기말과제

**Project2 Report**

# 학과 : 컴퓨터공학과 학번 : 12162065

# 이름 : 신동석

**Contents**

## What you’ve implemented and what you’ve NOT

1. Brief explanation of your implementation
2. How to compile and run
3. Talk about your experience of doing this project
4. Contact information
5. What you’ve implemented and what you’ve NOT

### Index creation

<BlockSize, root\_Bid, depth>로 이루어진 header를 입력하고 root block를 만들었다.

1. **Insertion**

sample로 주어진 “sample\_insertion\_input.txt” 파일의 정보를 tree의 index file에 input하였다. depth에 따라 성공과 실패가 나온 걸로 봐서 insert 함수가 제대로 구현하지 못하였다는 것을 알 수 있었으나 보완점을 찾는데 실패하였다. Depth가 1일 때만 정상적인 것으로 보아 완전히 잘못된 코드이다.

테이블이(가) 표시된 사진

자동 생성된 설명

### Point (exact) search

#### “sample\_search\_input.txt”를 열고 key값을 하나하나 받을때마다 search를 진행하여 key,id값을 찾아낸다.

#### 텍스트이(가) 표시된 사진 자동 생성된 설명

1. **Range search**

#### “sample\_range\_search..txt”를 열고 start\_key값,end\_key값을 받아 start\_key를 기준으로 leaf\_node까지 내려가서 search를 진행한다.

#### 텍스트이(가) 표시된 사진 자동 생성된 설명

1. **Print B+Tree structure**

depth가 1이하인 트리에 대해서는 모든 노드를 출력하고 1초과인 트리에 대해서는 level 0,1에 대해서만 출력한다. 정상적인 insert를 구현하지 못하여 depth가 1이상인 것에 대해 시행해보지 못하였다.

텍스트이(가) 표시된 사진

자동 생성된 설명

1. Brief explanation of your implementation
2. Insert

Insert는 전체적으로는 header로부터 root\_Bid를 받아서 해당 블락을 node배열에 복사하고 그 안의 key값들을 기준으로 leaf\_node까지 내려가서 insert를 한다. 그 후, split이 발생하면 타고 내려갔던 경로를 올라오며 block들에 추가할 수 있는 지 없는 지를 판별하여 root\_block까지 확인하는 알고리즘을 생각하였다. Split이 일어나면 bin파일의 가장 뒤에 해당 블락을 저장하고 그에 맞는 Bid를 부모 블락에 (leaf block의 경우 앞의 블럭에도) 기억시키는 방법을 생각했다.

1. Point search

완성된 b+tree라면 가지고 있는 key값을 들고 트리를 따라 내려가서 마지막에 leaf\_blok에서 확인하여 해당 id를 출력한다.

1. Range search

Point search와 같은 방법으로 start\_key를 기준으로 leaf block까지 내려가 lead block을 타고 range에 맞게 id들을 출력하였다.

1. Print

Print는 BFS의 개념을 이용하여 queue에 각 level에 해당하는 BID를 넣고 Block들을 호출해서 읽는 방법으로 구현하였다.

## How to compile and run

#### 사용한 개발 언어 : C++

사용한 개발 도구 : Microsoft Visual studio 2019

1. Microsoft Windows version
2. Index creation

bptree.exe c [bptree binary file] [block\_size]

#### Insertion

bptree.exe i [bptree binary file] [records data text file]

#### Point (exact) search

bptree.exe s [bptree binary file] [input text file] [output text file]

#### Range Search

bptree.exe r [bptree binary file] [input text file] [output text file]

#### Print B+\_Tree structure

bptree.exe p [bptree binary file] [output text file]

텍스트이(가) 표시된 사진

자동 생성된 설명

1. Talk about your experience of doing this project

이번 프로젝트는 project1에 비하여 굉장히 어렵게 느껴졌다. 일단, 파일을 입출력하는 과정이 익숙치 않았고 기본적으로 기말고사 범위였던 b+-tree를 열심히 공부하고 이론적으로 아는 것과 실제로 구현하는 것은 굉장한 차이가 있단 것을 느꼈다. Insert하는 과정에 있어서 트리 자체의 구현이 똑바로 되지 않으니 아래의 다른 구현 함수들이 전혀 활용될 수 없는 것을 보며 굉장히 답답함을 느꼈다. 이 답답함을 느끼며 구조를 구현하는 것이 모든 것의 시작이며 제대로 된 구조 없이는 기능은 의미가 없다는 것을 알았다. 동시에 교수님께서 첫 수업 때 악명높은 과제이지만 출분히 낼 가치가 있다고 말씀하신 것이 무엇을 의미하는지 알 수 있었다. 이제껏 해본 과제 중에서 가장 오랜 시간이 걸리고 완성도가 가장 떨어지는 제출물이다. 어떠한 구조를 구현하는데 있어서 항상 존재하는 라이브러리를 사용하는데 익숙했고 알고리즘을 스스로 공부하면서 쉬운 자료구조조차 혹시나 제대로 구현할 줄 모르며 쓰고있는 것은 아닐까라는 의문이 들었다. 비록 과제물은 완성도가 떨어지지만 앞으로의 공부나 학습 및 문제 풀이에 있어서 새로운 마음가짐을 가지게 만들어 주는 과제였던 동시에 수많은 고민을 하며 보완하고 하였기에 후회 없고 굉장히 재밌게 한 과제였다..

## Contact information

E-mail : [2006peter@naver.com](mailto:2006peter@naver.com)

Phone\_number : 010 – 9699-8073