

Lab 12 (5/19) B-Tree

Due : ~2022.05.24.(Tue) 23:59, Late Submission : ~2022.05.25.(Wed) 23:59

- **B-Tree Implementation - Insert, find, PrintInorder.**
- 아래의 사진과 같이 [input].txt 파일을 입력 받아 커맨드를 수행하고, 결과값을 [output].txt 파일에 출력하여 저장한다.
- **Implement the function that creates and edits B-tree by receiving the [input].txt file as input commands as below, and the output result is stored in the [output].txt file.**

```

3
i 10
i 40
i 70
i 4
p
i 92
i 54
i 32
f 5
f 32
p
i 29
f 10
p

4 10 40 70
finding error : key 5 is not in the tree!
key 32 found
4 10 32 40 54 70 92
key 10 found
4 10 29 32 40 54 70 92

```

- **<input> :**
각 line마다 Command 가 주어짐.(**Order, insert, find, print**)
Commands are given line by line (Order, insert, find, print)
- **<output> :**
각 Command에 맞는 result 출력.
Appropriate output messages for each command.

<Order Setting>

- 위 사진에서처럼 첫줄에 자연수로 입력됩니다.
- Natural number will be given as the first line in the example above.

<Insert>

- 위의 사진에서처럼 “**i [insert할 key값]**”로 표현합니다.
- “i 1”의 경우 “1” key를 B-tree에 insert 해주시면 됩니다.
- B-Tree 안에 이미 존재하는 Key를 insert 하는 경우, 에러 메시지를 출력해주셔야 합니다.
- key 중복시 에러메시지 형식은 “**insert error : key [key값] is already in the tree!**”입니다.
- “**i [keyValue to be inserted]**”.
- In the case of “i 1”, “1” key should be inserted into the B-tree.
- when inserting the key value that already exists in the tree, the error message should be printed out as the format of “**insert error : key [keyValue] is already in the tree!**”.

<Find>

- 위의 사진처럼 "f [search할 key값]" 로 표현합니다.
- "f 2" 의 경우, B-tree안에 "2"가 존재하는지 존재하지 않는지 출력해주시면 됩니다.
- 출력 형식은 key 값이 존재할 경우 "key key값 found"
key 값이 존재하지 않는 경우 "finding error : key key값 is not in the tree!" 입니다.
- 위의 사진처럼 "f [keyValue to be searched]" 로 표현합니다.
- In the case of "f 2", the existence of the key value in the tree should be printed out.
- When the key value exists in the tree, "key keyValue found",
when the key value does not exist in the tree, "finding error : key keyValue is not in the tree!" should be printed out.

<Print>

- 위 사진처럼 "p" 로 표현 합니다.
- "p"의 경우, inorder 로 B-tree 안에 있는 Key 값을 출력해주셔야 합니다.
- Tree가 비어있는 경우 에러 메시지를 출력해주셔야 합니다.
- 에러메시지 형식은 "print error : tree is empty!"입니다.
- "p"
- In the case of "p", all key values in the tree should be printed out in inorder traversal.
- When printing an empty tree, the error message should be printed in the format of "print error : tree is empty!"

<Structure & Function Format>

Structure

```
typedef struct BNode* BNodePtr;
struct BNode{
    int order;
    int size      /* number of children */
    BNodePtr *child; /* children pointers */
    int *key;      /* keys */
    int is_leaf;
}BNode;
```

Function

```
BNodePtr CreateTree(int order);
void Insert(BNodePtr root, int key);
int Find(BNodePtr root, int key);
void PrintTree(BNodePtr root);
void DeleteTree(BNodePtr root);
```

- 위 사진과 같은 Struct 구조체를 사용하셔야 합니다.
- 위 사진과 같은 함수들을 형식에 맞게 구현해주시면 됩니다.
- Struct format above should be used for implementation
- Functions should be implemented in appropriate format as above.

<File Name Format>

- [StudentID].c ex) 20XXXXXXXXXX.c

<Execution>

- gcc 20XXXXXXXXXX.c -o 20XXXXXXXXXX
- ./20XXXXXXXXXX [input_file_name] [output_file_name]
- !!! 꼭 제공되는 testcase로 실행시켜보시기 바랍니다!!!!,
- !!! Run your solution code with the provided test case above and check whether it works properly !!!

<|issue>

- 코드 작성시 주석을 적어주시기 바랍니다. 주석이 없는경우 Cheating으로 간주될 수 있습니다.
 - 제공된 testcase는 채점 case에 포함됩니다. 모두 알맞게 나오는지 확인해보시기 바랍니다.
 - 파일 입출력은 `argv[]` 를 사용하여 구현해주시기 바랍니다.
 - 제출 마감 시간 이전의 가장 최신 버전의 commit을 기준으로 채점할 예정입니다.
 - 제출 파일과, 폴더 naming 은 꼭 지정된 형식으로 해주셔야 합니다.
 - Please write down the detailed comments when writing the code. If there is no comment, it might be considered cheating.
 - Provided test case is included in the test case for grading. Please check to see if it makes a proper result.
 - Do not use a fixed file name when inputting and outputting files, but implement it using `argv[]` as in skeleton code.
 - Scoring will be based on the latest version of commit before the deadline.
 - The names of the .c file and directory should be named in proper format.
 -
 -
 - 출력시, 위 사진의 예시와 같은 형식으로 출력해주시면 됩니다. 모든 공백은 띄어쓰기 한칸입니다. 모든 출력 메시지의 알파벳은 소문자만 사용하여 출력합니다.
 - -> 출력시 위 사진과 같이 커맨드에 알맞는 메시지 출력후 줄바꿈.
 - All the messages must be printed out according to the appropriate format as shown in the example above. All spaces are one space. Only lowercase letters should be used for the alphabets in output messages.
 - -> newline(\n) after each output message.

<Directory Format>

- 아래와 같이 git 프로젝트 폴더에 "lab12" 폴더 생성후, "lab12" 폴더 안에 "20XXXXXXXXXX.c" 파일을 위치시키시면 됩니다.
 - After creating the "lab12" directory in the git-project-directory as below, place the "20XXXXXXXXXX.c" file in the "lab12" directory.

2022_CSE2010_20XXXXXXX/ (GitLab project directory)

Fig. 1.

---lab11/

-----2020XXXXXX.c

... . . .

---lab12/

-----2020XXXXXX.c