Project3

Kim, Seongjun

2010년 7월 22일

Goal

Make B+-tree index.

The key is 'int' type. Its size is 4 or 8 bytes (depends on machine). The data is pointer type and its size is also 4 or 8 byte. The block's size is 4096 bytes. If B+-tree is on 32-bit machine, leaf node has $\lfloor (4096 - 4)/4 \rfloor = 1023$ keys and pointers.

Insert and remove operations are trivial. But search operation is subtle. Because this B+-tree allows duplicate keys, search operation returns a iterator that points leftmost data of duplicate key. You can access all data by using this iterator. If it can't find key, it return null iterator.

Following 'BPTree.h', implement all public methods of BPTree class in 'BPTree.cc'.

Warning: Don't change declaration of public method of BPTree class. If you need, first let me know why it is need and how to change by e-mail.

How to compile & run

```
$ make
$ ./bptree_test
```

References

Chapter 10