

源代码目录

名称	修改日期	类型	大小
 btree_main.cpp	2020/9/10 2:23	C++ Source	17 KB
 Makefile	2020/9/10 1:51	文件	1 KB

此题目仅实现了非图形版

实现思路说明：

B 树的插入和删除都采用简单方法。

但是由于细节上的操作差异（用红色字体标出），和其他同学输出的树形有时会不同。（下面说明）

插入过程分为定位和插入两个步骤

1) 定位：

查找插入该关键字的位置，即最底层中的某个非叶子节点（规定一定是插入在最底层的某个非叶子节点内）

2) 插入：

若插入后，不破坏 3 阶二叉树的定义，即插入后节点关键字个数在【1，2】之间，直接插入。
若插入后，关键字数量大于 2，则对插入后的节点进行分裂操作

分裂：

插入后的节点中间位置关键字 2 并入父结点中，关键字 1 留在原来节点，关键字 3 放入新节点中，若并入父节点后，父节点关键字超出范围，继续向上分裂，直到符合要求为止

删除过程

只删除叶子节点，如果删除的不是叶子节点，就找右子树的最小关键字进行替代删除。

1) 直接删除

若被删除关键字所在节点关键字个数大于 1，直接删除

2) 兄弟够借

若被删除关键字所在节点关键字总数为 1，且与此节点临近的兄弟节点的关键字大于 1，则需要从兄弟节点接一个关键字，此过程需要调整该节点、双亲节点和兄弟节点的关键字。

先从右兄弟借、后从左兄弟借

3) 兄弟不够借

若被删除关键字所在节点关键字总数为 1，并且与此节点相邻的兄弟节点的关键字个数为 1，则删除关键字，并与一个不够借的兄弟节点和双亲节点中两兄弟子树中的关键字合并。

合并后若双亲节点因减少一个节点导致不符合定义，则继续执行 2、3 步骤。

使用方法

./btree --input in.dat --output out.dat