

BTree

1、定义

在定义之前先介绍几个概念：

阶：树的阶其实就是树种可以最多有多少个子节点，比如2叉树的阶就是2，阶一般是人们自己定义的。

度：树的度就是树的最大深度，从根节点到叶子节点的计数

下面开始定义一个m阶B数的属性：

- 1、每一个节点最多拥有m个子节点
- 2、每一个非叶子节点（除根节点）最少有 $\lceil m/2 \rceil$ （向上取整）个子节点
- 3、如果跟节点不是叶子节点，也就是说整棵树不只有一个根节点的情况下，那么根节点至少有2个子节点。
- 4、有k个子节点的非叶子节点拥有k-1个键。
- 5、所有叶子节点在同一层。

上面这些属性将会在下面示意图中展示出来，没理解可以多对照示意图看看。

每一个内部节点的键将节点的子树分开。例如，如果一个内部节点有3个子节点（子树），那么它就必须有两个键： a_1 和 a_2 。左边子树的所有值都必须小于 a_1 ，中间子树的所有值都必须在 a_1 和 a_2 之间，右边子树的所有值都必须大于 a_2 。

内部节点

内部节点是除叶子节点和根节点之外的所有节点。它们通常被表示为一组有序的元素和指向子节点的指针。每一个内部节点拥有最多 U 个，最少 L 个子节点。元素的数量总是比子节点指针的数量少一（元素的数量在 $L-1$ 和 $U-1$ 之间）。 U 必须等于 $2L$ 或者 $2L-1$ ；因此，每一个内部节点都至少是半满的。 U 和 L 之间的关系意味着两个半满的节点可以合并成一个合法的节点，一个全满的节点可以被分裂成两个合法的节点（如果父节点有空间容纳移来的一个元素）。这些特性使得在B树中删除或插入新的值时可以调整树来保持B树的性质。

根节点

根节点拥有的子节点数量的上限和内部节点相同，但是没有限下。例如，当整个树中的元素数量小于 $L-1$ 时，根节点是唯一的节点并且没有任何子节点。

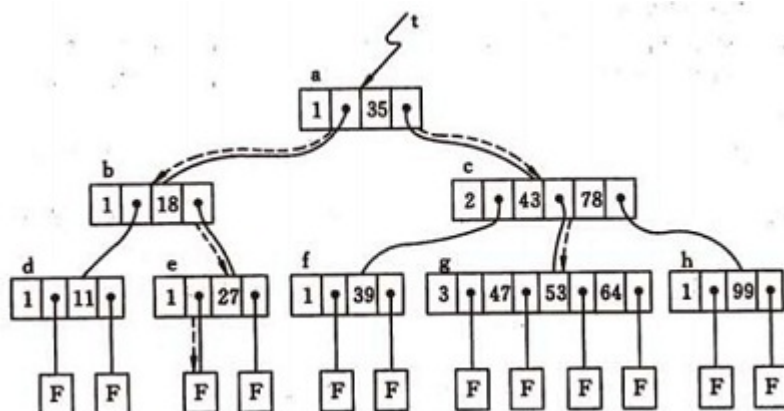
叶子节点

叶子节点对元素的数量有相同的限制，但是没有子节点，也没有指向子节点的指针。

2、BTree操作

查找

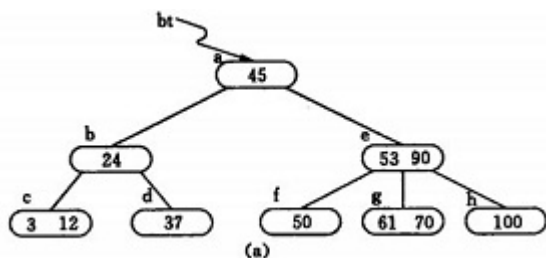
B数的每个节点是多键码的有序表，到达一个节点时候，现在节点内查找（可以采用二分查找），若找到就返回，否则继续在其子节点查找，若到达子节点时候还没找到元素，则说明元素不存在。下面借用一张图来说明下查找47的过程：（注：图中每个节点的第一个槽位表示节点中有多少个键，最后一层便于理解加了一层空层）



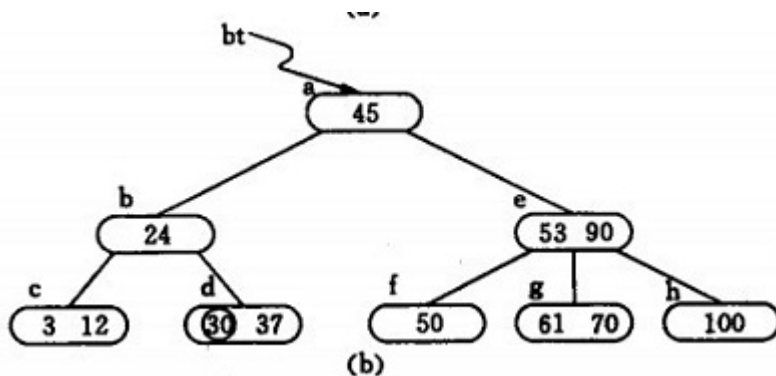
(1) 首先进入跟节点a，47大于根节点中的键35，所以去35右边指针指向的子节点c查找 (2) 节点c中，47在键43和78之间，所以继续沿着指针找到其子节点g (3)在g节点中成功找到47，返回。

插入

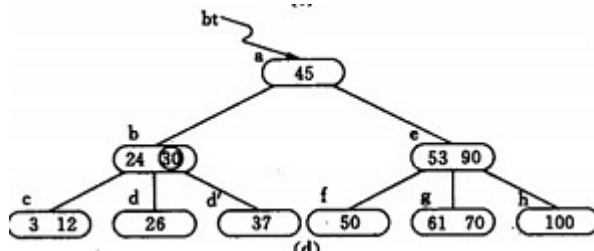
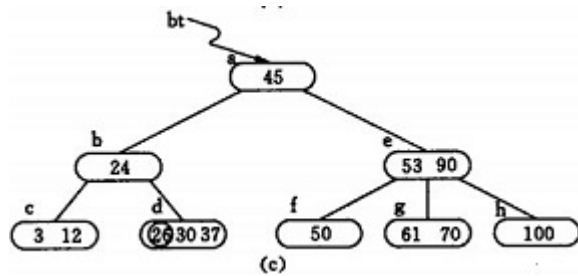
如下图，在一个3阶B树种插入关键字30、26、85。



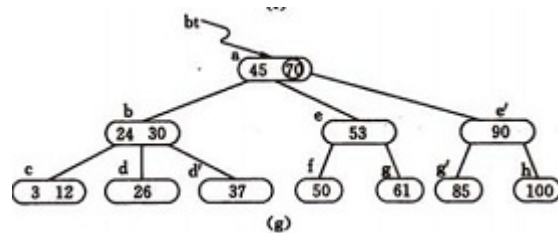
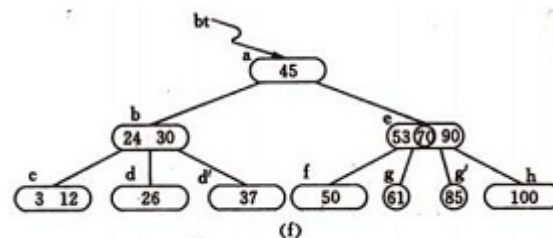
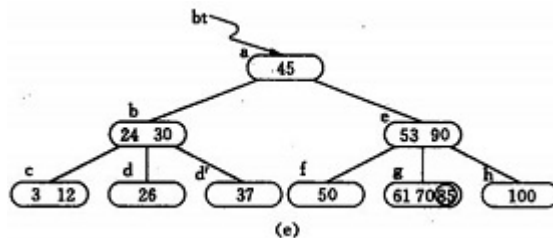
(1) 首先查找a节点，由于30小于45，故去45的左子树，到达b节点以后，30大于24，故去24的右子树d，由于d没有子树，且d中键的数量小于m-1 (2)，所以将30插入d节点。第一个关键字插入完成，结果如下图



(2) 同样的过程，26应该插入节点d，由于d中再插入数据的话键的数量不小于m-1，故将d中30放入父节点b，添加子节点d'在b节点中30的右边，d'中放入30的原先在d中的右边元素。结果如下图所示。



(3) 紧接着放入85，根据搜索85应该放入节点g中，但是放入节点g中以后g的键的数量大于m-1，故将g节点进行拆分，将g中的70往父节点c提升，然后将70的左右邻居键作为70的左右子树，由于70上升到c节点后，c节点的键数量也大于m-1，故将70再往c的父节点a提升，且将70在c中的左右节点作为其左右子树，元素插入完成，结果如下图所示。



总结一下，所有的插入都从根节点开始。要插入一个新的元素，首先搜索这棵树找到新元素应该被添加到的叶子节点。将新元素插入到这一节点中的步骤如下：1、如果节点拥有的元素数量小于最大值，那么有空间容纳新的元素。将新元素插入到这一节点，且保持节点中元素有序。2、否则的话这一节点已经满了，将它平均地分裂成两个节点：(1) 从叶子节点的元素和新的元素中选择出中位数 (2) 小于这一中位数的元素放入左边节点，大于这一中位数的元素放入右边节点，中位数作为分隔值 (3) 分隔值被插入到父节点中，这可能会造成父节点分裂，分裂父节点时可能又会使它的父节点分裂，以此类推。如果没有父节点（这一节点是根节点），就创建一个新的根节点（增加了树的高度）。

如果分裂一直上升到根节点，那么一个新的根节点会被创建，它有一个分隔值和两个子节点。这就是根节点并不像内部节点一样有最少子节点数量限制的原因。每个节点中元素的最大数量是 $U-1$ 。当一个节点分裂时，一个元素被移动到它的父节点，但是一个新的元素增加了进来。所以最大的元素数量 $U-1$ 必须能够被分成两个合法的节点。如果 $U-1$ 是奇数，那么 (阶) $U=2L$ ，总共有 $2L-1$ 个元素，一个新的节点有 $L-1$ 个元素，另外一个有 L 个元素，都是合法的节

点。如果 $U-1$ 是偶数，那么（阶） $U=2L-1$ ，总共有 $2L-2$ 个元素。一半是 $L-1$ ，正好是节点允许的最小元素数量。示例图如下

