



数据结构与算法(五)

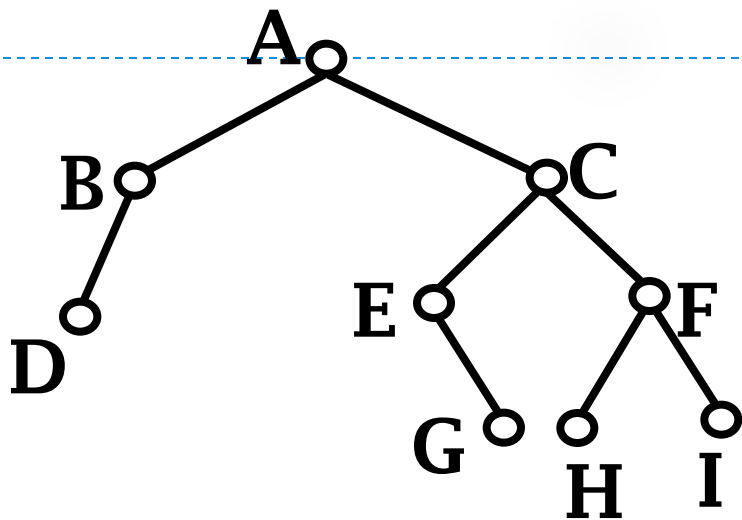
张铭 主讲

采用教材：张铭，王腾蛟，赵海燕 编写
高等教育出版社，2008. 6（“十一五”国家级规划教材）



第五章 二叉树

- 二叉树的概念
- 二叉树的抽象数据类型
 - 深度优先搜索
 - 宽度优先搜索
- 二叉树的存储结构
- 二叉搜索树
- 堆与优先队列
- Huffman树及其应用



5.4 二叉搜索树

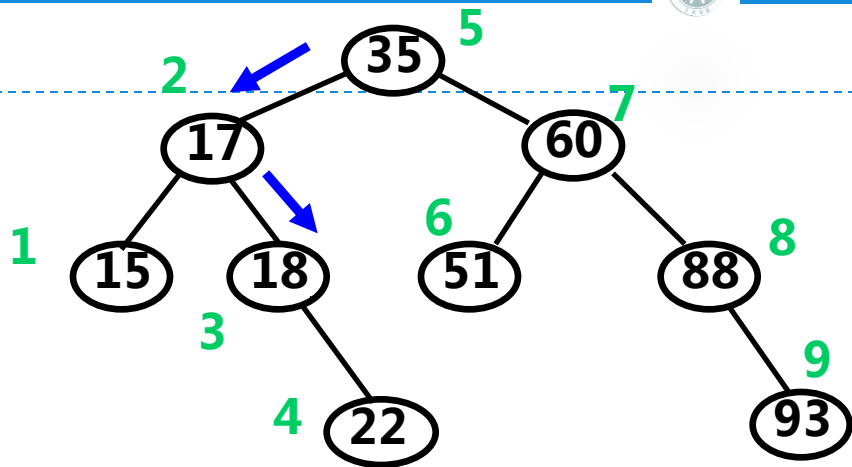
二叉搜索树

• Binary Search Tree (BST)

- 或者是一棵空树；
- 或者是具有下列性质的二叉树：

- 对于任何一个结点，设其值为K
- 则该结点的 **左子树**(若不空)的任意一个结点的值都 **小于 K**；
- 该结点的 **右子树**(若不空)的任意一个结点的值都 **大于 K**；
- 而且它的左右子树也分别为BST

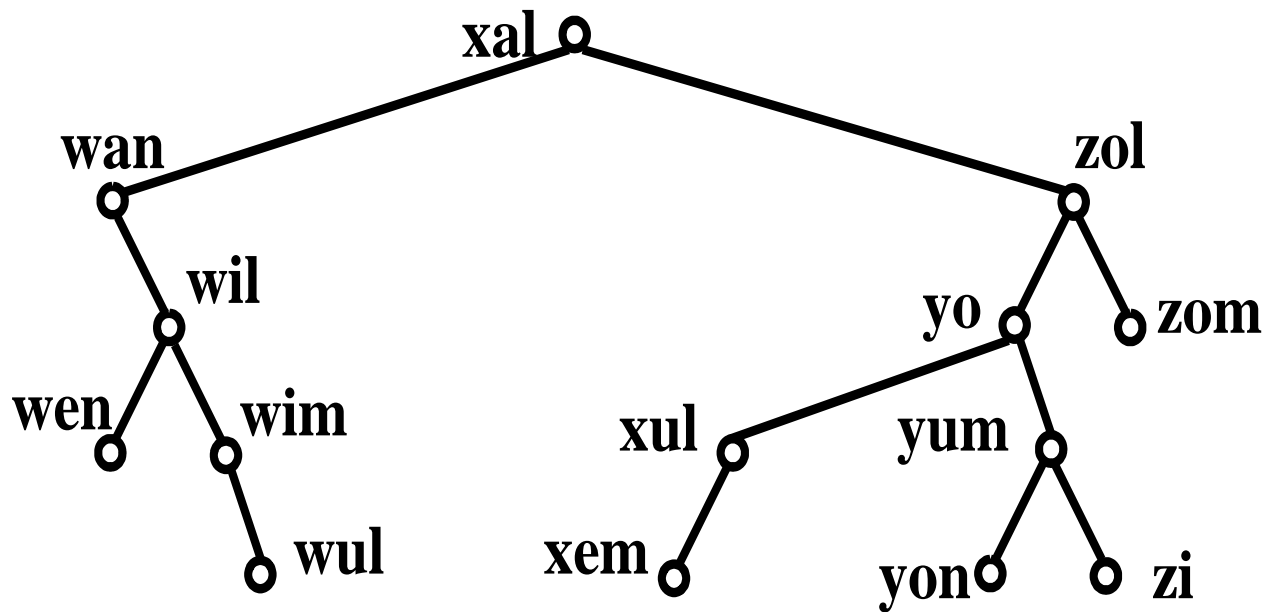
- 性质: **中序遍历是正序的** (由小到大的排列)





5.4 二叉搜索树

BST示意图

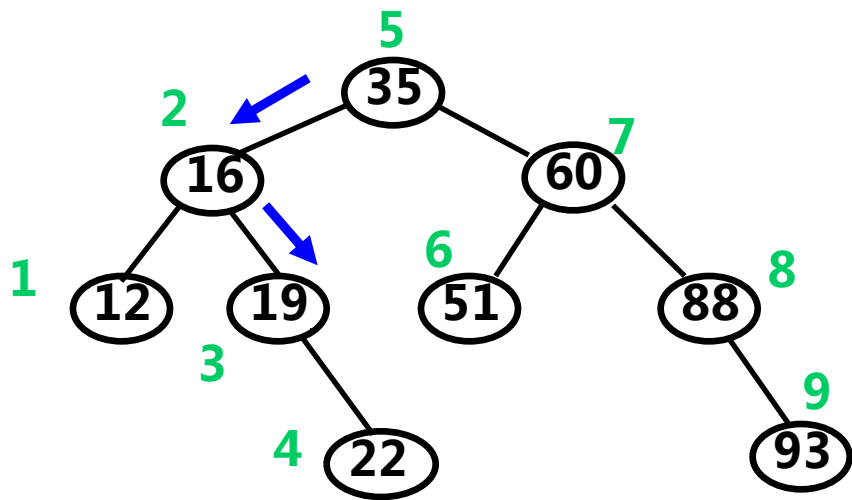


检索 19

□ 只需检索二个子树之一

□ 直到 K 被找到

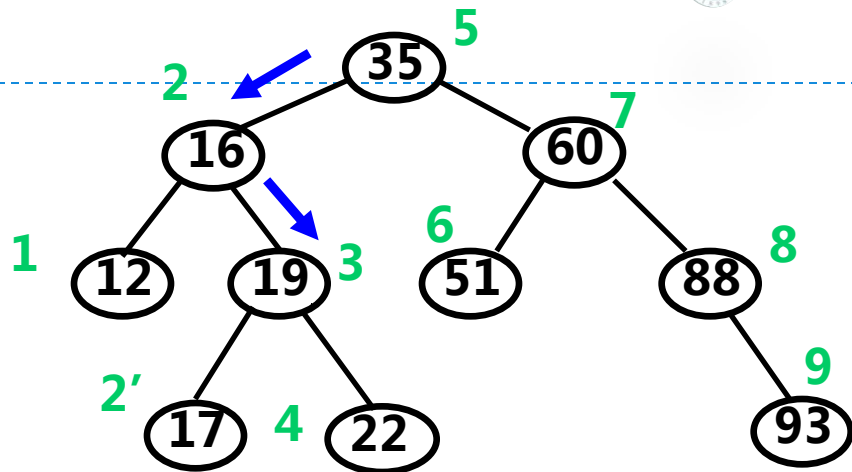
□ 或遇上树叶仍找不到，则不存在

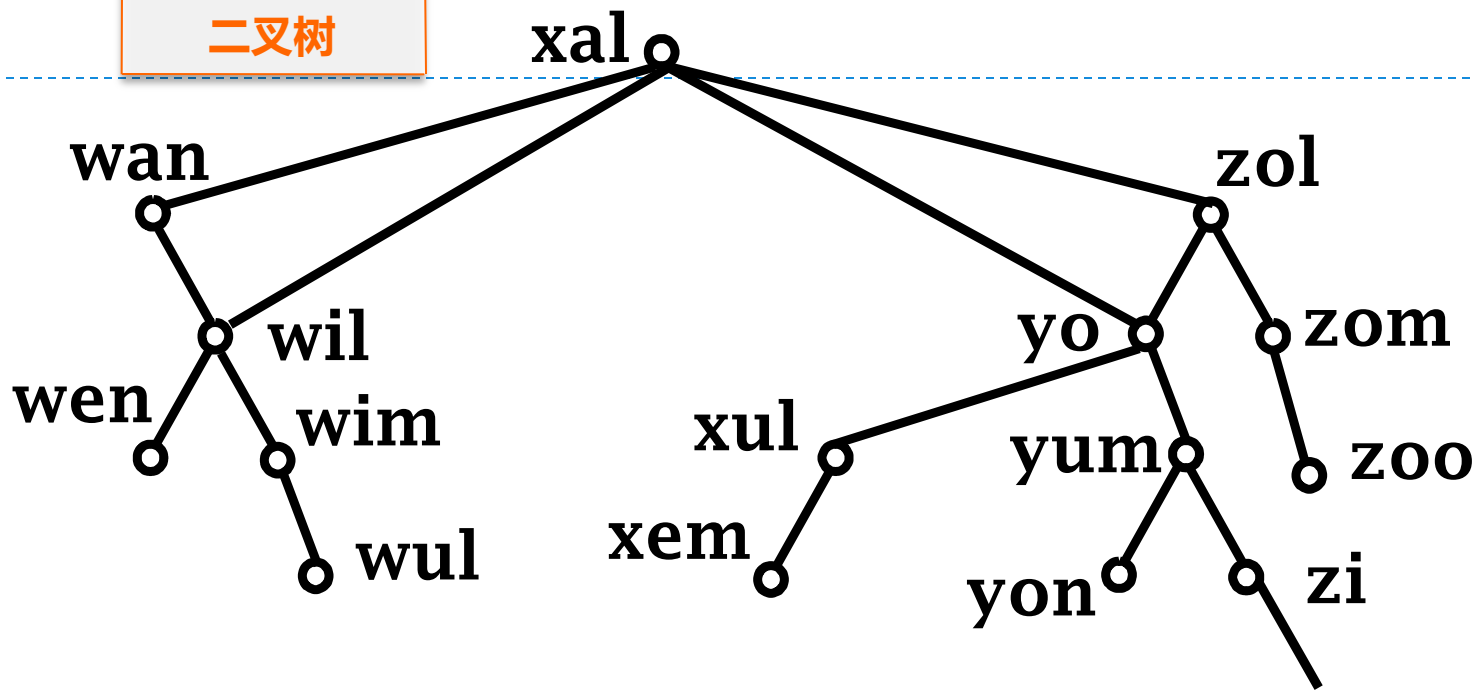


5.4 二叉搜索树

插入17

- 首先是检索，若找到则不允许插入
- 若失败，则在该位置插入一个新叶
- 保持BST性质和性能！





□ 删除 wan

□ 删除 zol

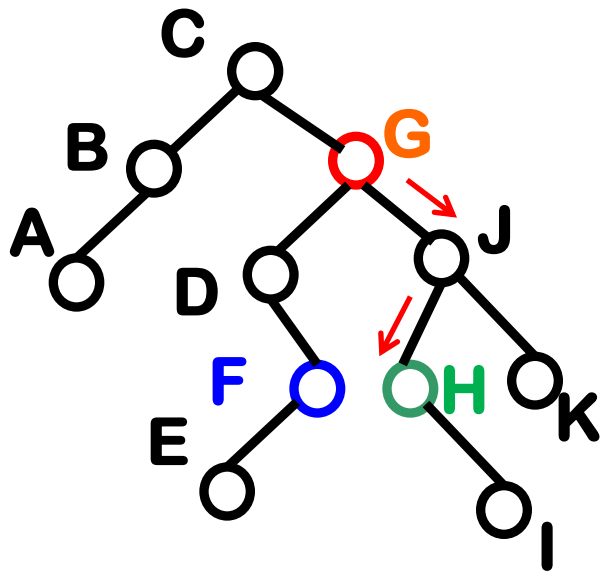


BST删除(值替换)

```

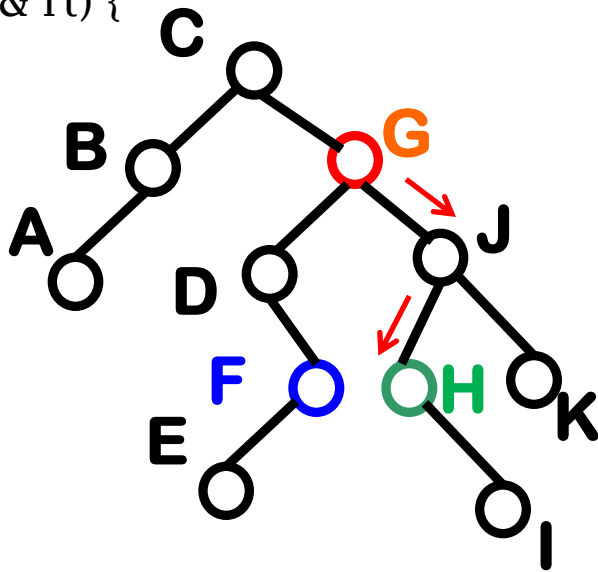
void BinarySearchTree<T>::removehelp(BinaryTreeNode <T> *& rt, const T val) {
    if (rt==NULL) cout<<val<<" is not in the tree.\n";
    else if (val < rt->value())
        removehelp(rt->leftchild(), val);
    else if (val > rt->value())
        removehelp(rt->rightchild(), val);
    else {
        // 真正的删除
        BinaryTreeNode <T> * temp = rt;
        if (rt->leftchild() == NULL) rt = rt->rightchild();
        else if (rt->rightchild() == NULL) rt = rt->leftchild();
        else {
            temp = deletemin(rt->rightchild());
            rt->setValue(temp->value());
        }
        delete temp;
    }
}

```



找rt右子树中最小结点，并删除

```
template <class T>
BinaryTreeNode* BST::deletemin(BinaryTreeNode <T> *& rt) {
    if (rt->leftchild() != NULL)
        return deletemin(rt->leftchild());
    else { // 找到右子树中最小，删除
        BinaryTreeNode <T> *temp = rt;
        rt = rt->rightchild();
        return temp;
    }
}
```



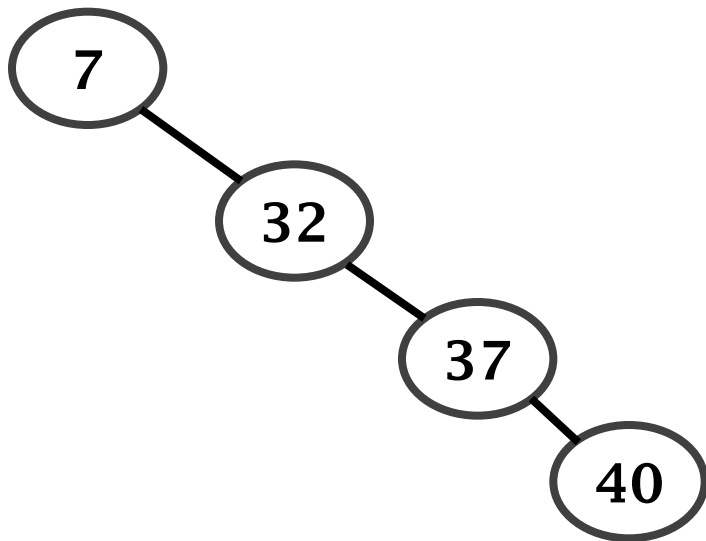


二叉搜索树总结

- 组织内存索引
 - 二叉搜索树是适用于内存存储器的一种重要的树形索引
 - 常用红黑树、伸展树等，以维持平衡
 - 外存常用B/B+树
- 保持性质 vs 保持性能
 - 插入新结点或删除已有结点，要保证操作结束后仍符合二叉搜索树的定义

思考

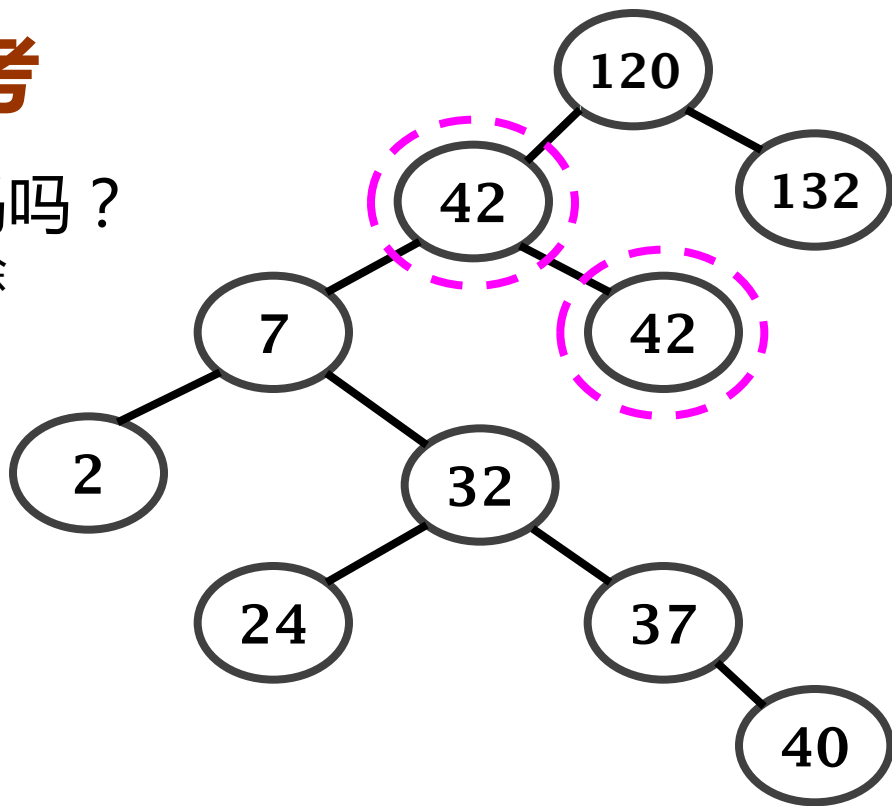
- 怎样防止BST退化为线性结构？



5.4 二叉搜索树

思考

- 允许重复关键码吗？
 - 插入、检索、删除





张铭《数据结构与算法》



数据结构与算法

谢谢聆听

国家精品课 “数据结构与算法”

<http://www.jpk.pku.edu.cn/pkujpk/course/sjig/>

张铭，王腾蛟，赵海燕

高等教育出版社，2008. 6。 “十一五” 国家级规划教材