

实验-5-查找-参考程序

王新宇

计算机科学与技术系

1. 二叉排序树的建立、中序遍历、查找、插入、删除的实现方法与课堂所讲一致，此处不再重复给出代码。

2. 编写递归算法，从大到小输出关键字不小于x的数据元素。

方法：使用右子树、根、左子树的遍历顺序寻找并输出关键字不小于x的数据元素。

```
template <class DataType>
void PrintNLT(const BinarySortTree<DataType> &tree, const BTreeNode<DataType> *p,
const DataType x) {
    DataType e;
    if(p) {
        PrintNLT(tree, tree.GetRightChild(p), x); //-----补充定义GetRightChild函数
        tree.GetElem(p, e); //-----补充定义GetElem函数
        if(e < x) return;
        cout << e << " ";
        PrintNLT(tree, tree.GetLeftChild(p), x); //-----补充定义GetLeftChild函数
    }
}
```

//在BinaryTree.h中补充定义GetElem、GetRightChild、GetLeftChild函数

```
template <class DataType>
```

```
Status BinaryTree<DataType>::GetElem(const BTreeNode<DataType> *p,
```

```
DataType &e) const { //用e返回节点p的数据元素
```

```
    if (!p) return FAILED;
```

```
    e = p->data;
```

```
    return SUCCESS;
```

```
}
```

```
BTreeNode<DataType> *GetLeftChild(const BTreeNode<DataType> *p) const
```

```
{ return p->lChild; } //返回p的左孩子指针
```

```
BTreeNode<DataType> *GetRightChild(const BTreeNode<DataType> *p) const
```

```
{ return p->rChild; } //返回p的右孩子指针
```