

## 问题：BinaryTree 的设计

课本提供的代码中，给出的头文件 `BinarySearchTree.h`, `AvlTree.h`, `RedBlackTree.h` 和 `SplayTree.h` 分别实现了对应的功能。由于是教学代码，每一个头文件都是独立的。现在，请根据它们各自的用途，整理它们的逻辑关系，重构全部代码，并用继承关系予以表达。

### 1 BinaryTree 类

`BinaryTree` 类是一个模板类，用于实现二叉树的基本结构。它定义了一个受保护的结构 `BinaryNode` 和一个指向 `BinaryNode` 的指针 `root`。此外，它还声明了一系列虚拟函数，以便在派生类中进行具体实现。

#### 1.1 BinaryNode

可以看到 `BinarySearchTree` 和 `SplayTree` 的派生类中的节点均使用了结构体 `BinaryNode`, `AvlTree` 和 `RedBlackTree` 的节点需要对于 `BinaryNode` 中的成员进行了扩充，所以可以用继承的关系处理这两种节点。

#### 1.2 虚拟函数

给出派生类的共同函数

```
virtual const Comparable& findMin() const = 0;
virtual const Comparable& findMax() const = 0;
virtual bool contains(const Comparable& x) const = 0;
virtual bool isEmpty() const { return root == nullptr; }
virtual void printTree(ostream& out = cout) const = 0;
virtual void makeEmpty() = 0;
virtual void insert(const Comparable& x) = 0;
virtual void insert(Comparable&& x) = 0;
virtual void remove(const Comparable& x) = 0;
```

### 2 实现

给出 `TestBinaryTree.cpp`, 代码包含原有的 `TestBinarySearchTree.h`, `TestAvlTree.h`, `TestRedBlackTree.h` 和 `TestSplayTree.h`, 只要能够完成运行并且“Finish”或者“Complete”即表该树运行成功。

#### 2.1 疑点

按照原有的 `TestSplayTree.h` 测试 `SplayTree` 时，我发现代码无法成功运行，但是当我把 `NUMS4=30000` 改成 `NUMS4=3000` 就能成功运行了。