# 哈希

https://blog.csdn.net/zgwangbo/article/details/52204893

解决哈希冲突

1：线性探测

2：二次探测

3：拉链法

# 红黑树:

https://blog.csdn.net/sun\_tttt/article/details/65445754

性质一：节点是红色或者是黑色；

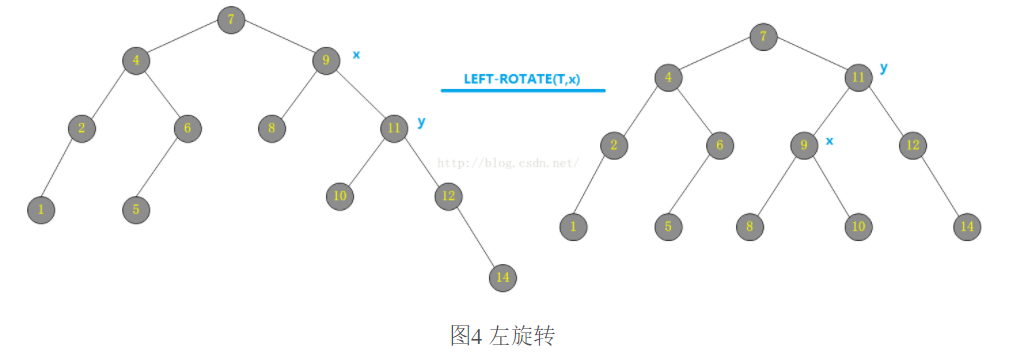
性质二：根节点是黑色；

性质三：每个叶节点（NIL或空节点）是黑色；

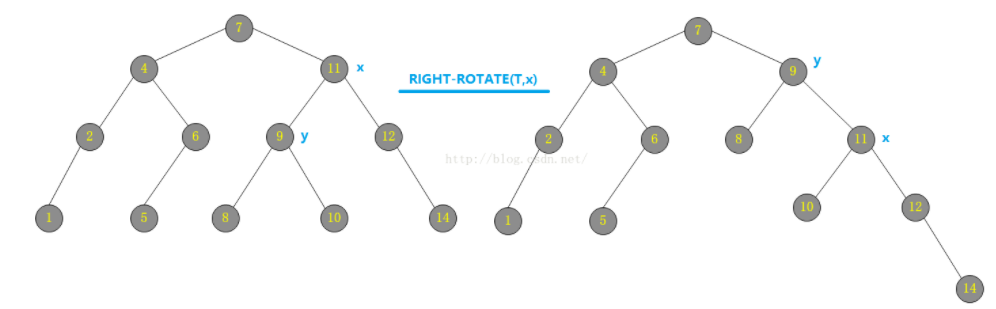
性质四：每个红色节点的两个子节点都是黑色的（也就是说不存在两个连续的红色节点）；

性质五：从任一节点到其没个叶节点的所有路径都包含相同数目的黑色节点；

左旋：对某个结点x做左旋操作时，假设其右孩子为y而不是nil：以x到y的链为“支轴”进行。使y成为该子树新的根结点，x成为y的左孩子，y的左孩子成为x的右孩子



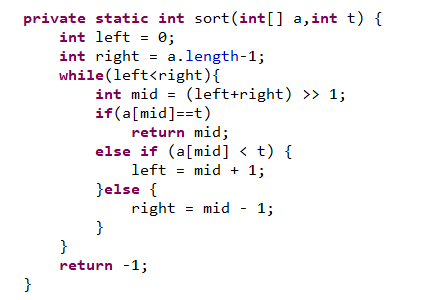
右旋: 对某个结点x做右旋操作时，假设其左孩子为y而不是nil：以x到y的链为“支轴”进行。使y成为该子树新的根结点，x成为y的右孩子，y的右孩子成为x的左孩子。



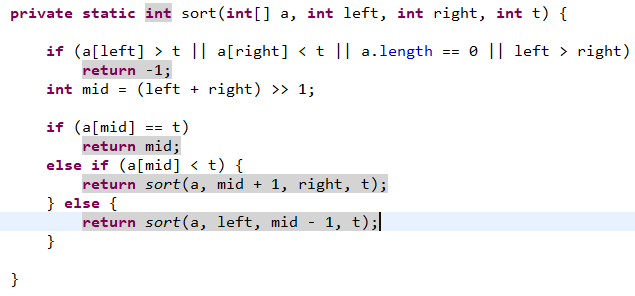
# 查找算法：

## 二分查找 O=logn

循环实现



递归实现



# 递归

如何分析递归行为复杂度?