**红黑树：**

特性

1.节点是红或者黑

2.根节点为黑

3.每个叶子节点都为黑，null节点

4.每个红节点两个子节点都是黑色

5.任意一节点到每一个叶子节点所有路径包含相同的黑色节点

插入情况

当前节点为根节点：

根节点直接变黑

当前节点的父节点是黑：

什么都不做

当前节点的父节点、叔父节点都是红色的：

1.父节点、叔父节点都变为黑色

2.祖父节点变为红色

3.指针移动到祖父节点，继续判断

当前节点父节点为红，叔父节点为黑，当前节点靠近叔父节点：

1.指针移动到父节点

2.以父节点为支点，向叔父节点相反方向旋转

当前节点父节点为红，叔父节点为黑，当前节点远离叔父节点：

1.指针移向父节点，父节点变为黑色

2.指针移向祖父节点，祖父节点变为红色

3以祖父节点为支点，向叔父节点方向旋转

最后一步，总能保证最上层节点为黑色。所以最多经历三步变化总能保证红黑树特性