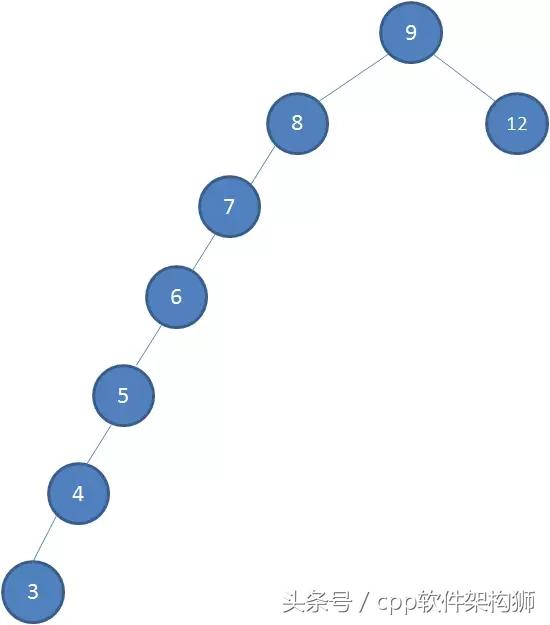
* **红黑树**（是一种二叉查找树）

二叉查找树使用了二分法，方便查找元素

但是如果有频繁的插入删除，

要调整的频率较高，

且会遇到右图这种情况，

所以引入了红黑树的概念

* **二叉查找树的基本规则**

时间复杂度是O(N)

二叉查找树又叫二叉搜索树、二叉排序树

有如下几个特点：

节点的左子树小于节点本身

节点的右子树大于节点本身

左右子树同样为二叉搜索树

* **红黑树的基本规则**

1. 根节点是**黑**色的
2. 一个点非**红**即**黑**
3. 红色节点的两个子节点为**黑（**从每个叶子到根的所有路径上不能有两个连续的红色节点**）**
4. 空的叶子节点（nil）都为**黑**
5. 从任一节点到其每个叶子的所有简单路径都包含相同数目的黑色节点（简称黑高）

* **红黑树的操作**

当我们在对红黑树进行插入和删除等操作时，对树做了修改，那么可能会违背红黑树的性质。

所以我们通过以下几种方式解决不平衡的问题

1. 变色
2. 左旋
3. 右旋

具体操作看这个网站

<https://my.oschina.net/u/3272058/blog/1914452>

旋转动图↓

https://blog.csdn.net/qq\_39879264/article/details/89248664?utm\_medium=distribute.pc\_relevant.none-task-blog-2%7Edefault%7EBlogCommendFromMachineLearnPai2%7Edefault-1.control&depth\_1-utm\_source=distribute.pc\_relevant.none-task-blog-2%7Edefault%7EBlogCommendFromMachineLearnPai2%7Edefault-1.control