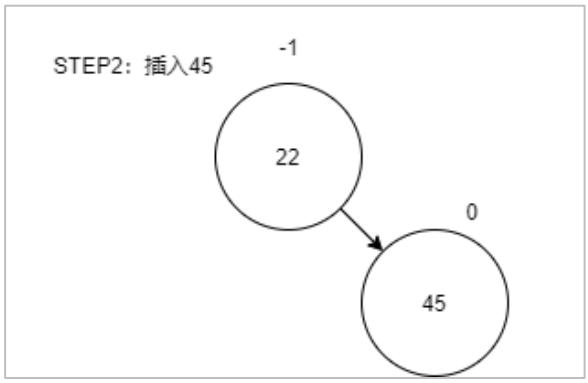
1、 请对长度为 10 的表：(22, 45, 56, 12, 33, 57, 88, 94, 44, 11)画出构造平衡二叉树的过程。

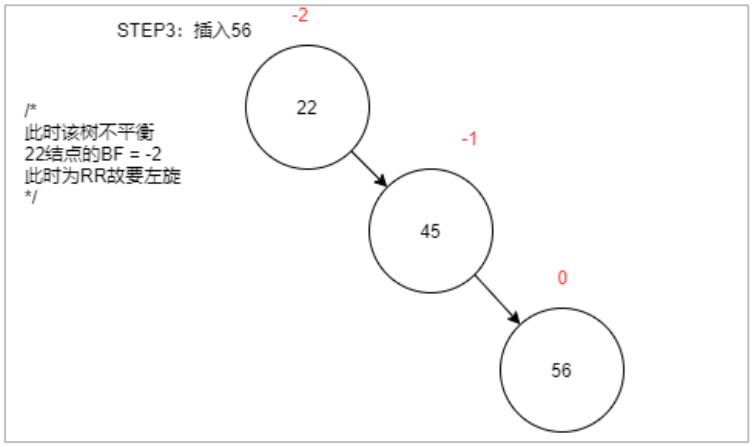
解：

第一步：插入 22

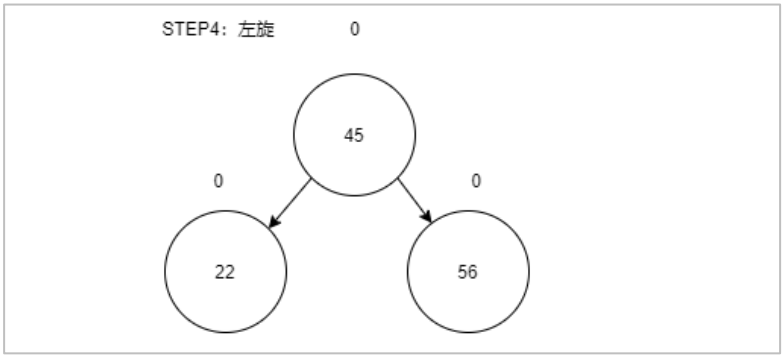
第二步：插入 45



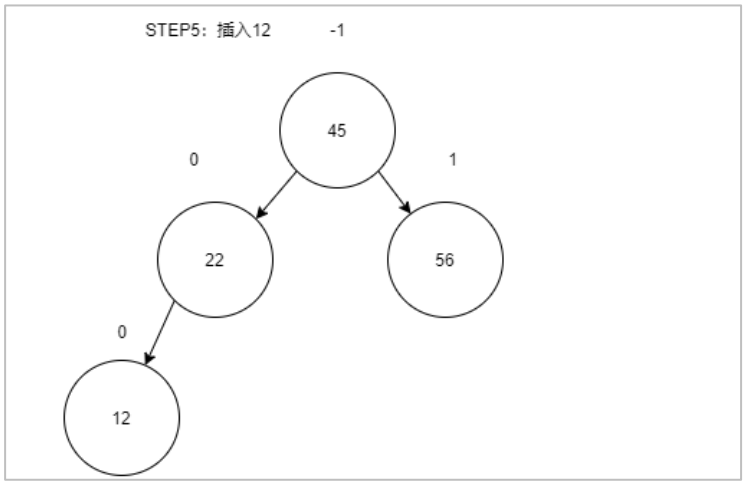
第三步：插入 56（此时出现了不平衡）



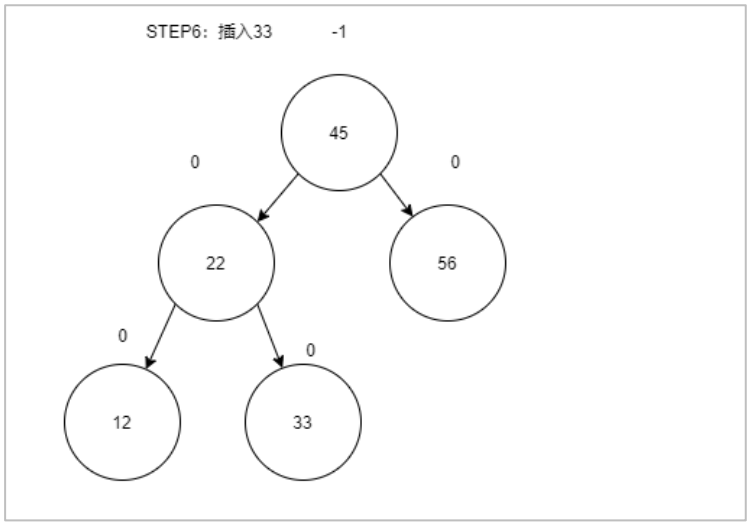
第四步：左旋 22



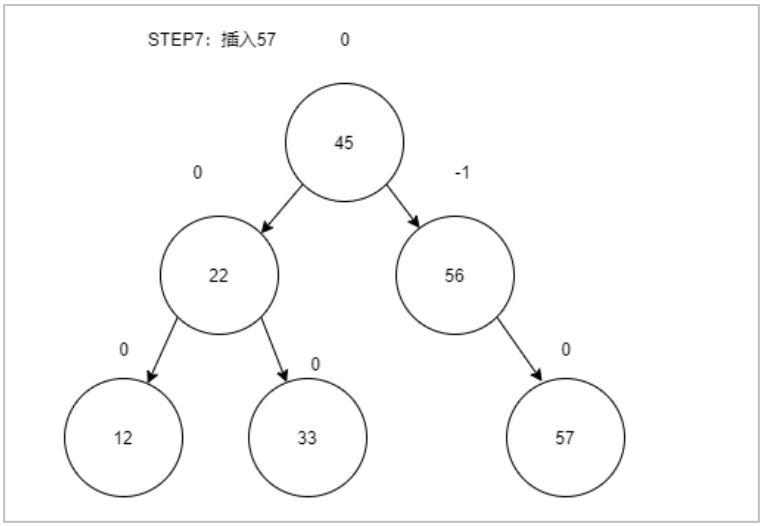
第五步：插入 12



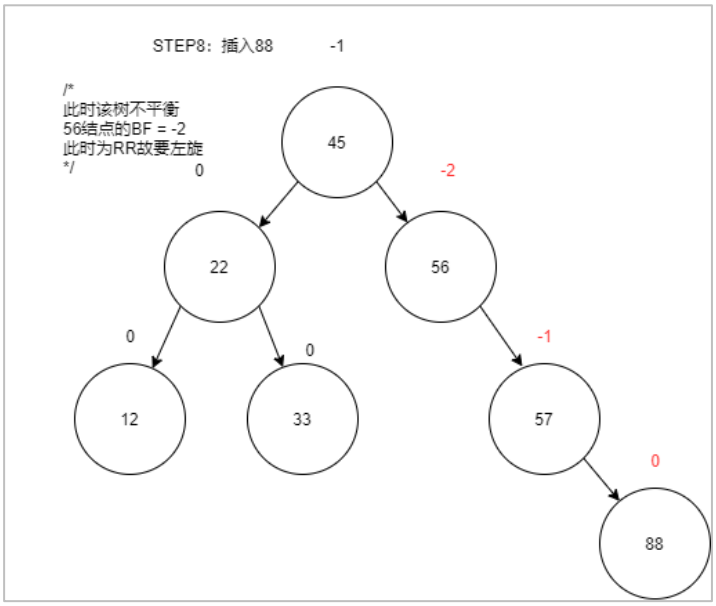
第六步：插入 33



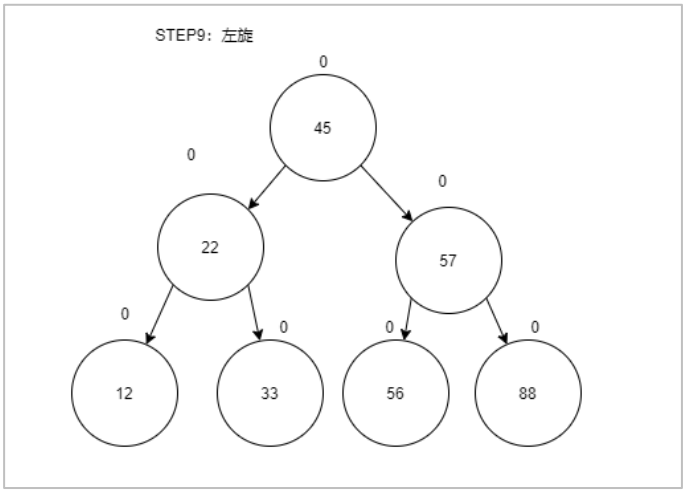
第七步：插入 57



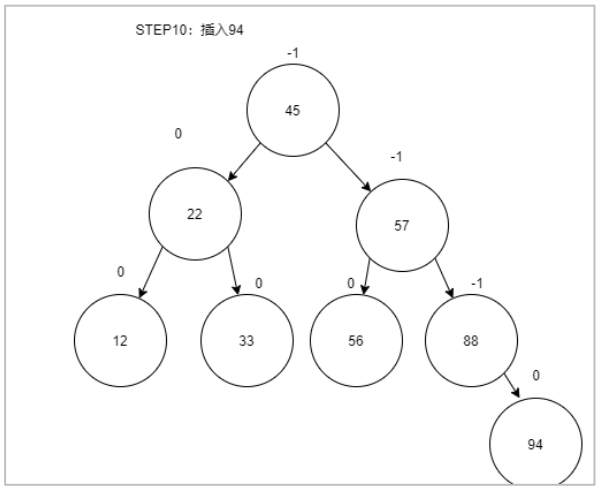
第八步：插入 88（此时出现了不平衡）



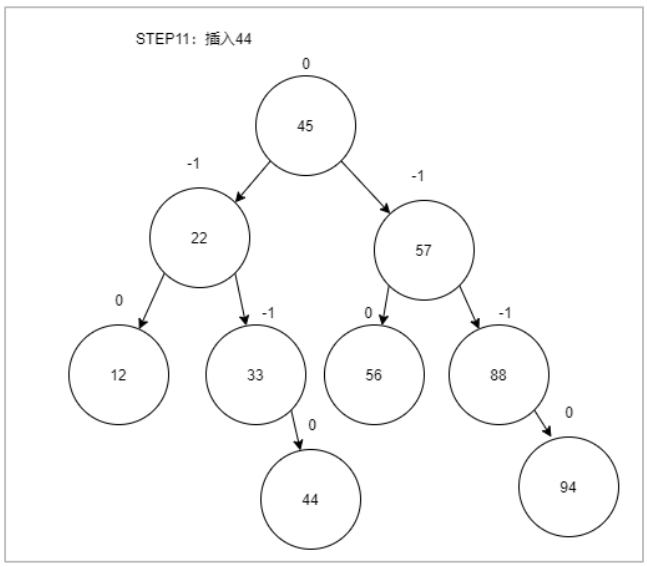
第九步：左旋



第十步：插入 94



第十一步：插入 44



第十二步：插入 11

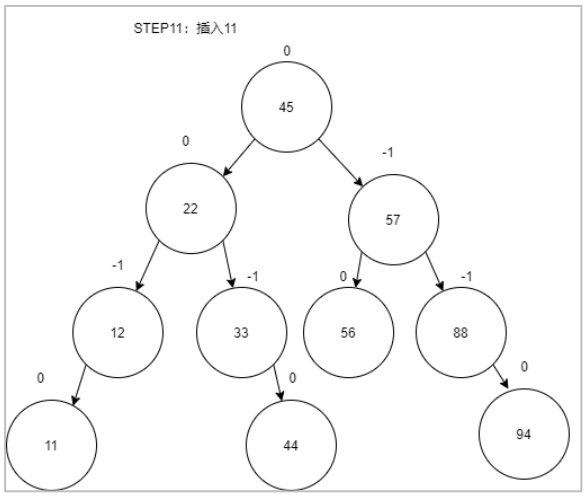


Figure1. 构造平衡二叉树的过程

2、 假设一棵平衡二叉树的每个结点都标明了平衡因子 b，设计一个算法，求平衡二叉树的高度。

解：

注：本来想把整个平衡二叉树排序流程都给实现的，但是在旋转结点的时候出现了问题并且找不到方法解决（指针的指针都解决不了）

求高度的方法：
从根节点开始，每个结点的 bf 值若为 1 则左边的树深度更深，为-1 则右边的树深度更深，为 0 则两边一样深度，只需要一直往深的一边前进，直到为空停止即可。

```
// 求平衡二叉树的高度
int AvlTreeHeight(AvlTree root)
{
    int height = 0;
    if (root->left == nullptr && root->right == nullptr) // 或root->bf == 0也行
    {
        return 1; // 若只有根节点则返回高度为1
    }

    while (root != nullptr) // 一直为空才停止
    {
        if (root->bf == 1 && root->bf == 0)
        {
            root = root->left;
            height++;
        }
        else
        {
            root = root->right;
            height++;
        }
    }
    return height;
}
```

Figure2 函数`AvlTreeHeight`代码快照

```
// 求平衡二叉树的高度代码
int AvlTreeHeight(AvlTree root)
{
    int height = 0;
    if (root->left == nullptr && root->right == nullptr) // 或 root->bf
== 0 也行
    {
        return 1; // 若只有根节点则返回高度为 1
    }

    while (root != nullptr) // 一直为空才停止
    {
        if (root->bf == 1 && root->bf == 0)
        {
            root = root->left;
            height++;
        }
        else
        {
            root = root->right;
            height++;
        }
    }
    return height;
}
```

出问题的函数：

```
1.void insetAvlNode(AvlTree &root, int data)
2.void rotate(AvlNode **node, const char *type)
```

3、 设哈希函数 H（K）=3 K mod 11，哈希地址空间为 0～10，对关键字序列（32，13，49，24，38，21，4，12），按下述两种解决冲突的方法构造哈希表，并分别求出等概率下查找成功时和查找失败时的平均查找长度 ASLsucc 和 ASLunsucc。

① 线性探测法；

② 链地址法。

解：

将 32，13，49，24，38，21，4，12 带入 H（K）中算得

H（32）= 8；H（13）= 6；H（49）= 3；H（24）= 6；

H（38）= 9；H（21）= 8；H（4）= 1；H（12）= 9；

易得:`38`与 `12`冲突， `13` 与`24`冲突， `21`与 `32`冲突。

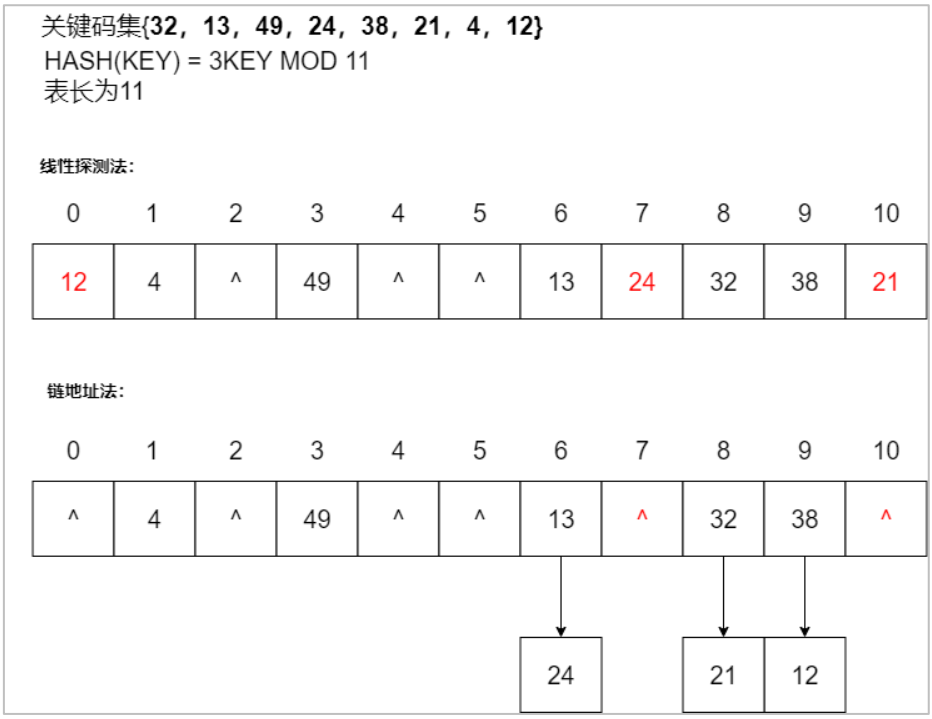


Figure3 使用线性探测法和链地址法解决冲突的的哈希表

线性探测法装填因子 α = 8/11：

$$\mathbf{ASLsucc} = \frac{1}{2}(1 + 1/(1 - a)) = 2.33$$

$$\mathbf{ASLunsucc} = \frac{1}{2}(1 + 1/(1 - a)^{\wedge}2) = 7.224$$

链地址法装填因子 α = 5/11：

$$\mathbf{ASLsucc} = 1 + a/2 = 1.227$$

$$\mathbf{ASLunsucc} = a + e^{\wedge}(a) = 1.482$$

----------END----------