### 实验三：红黑树插入算法

**实验要求：**

编码实现红黑树的插入算法，使得插入后依旧保持红黑性质。（即：实现教材p178页的RB-INSERT, RB\_INSERT\_FIXUP算法）

节点属性：(也可自行定义, 不做强制要求)

TNode = ｛

Color： red / black，

Key： int，

Left： TNode\*，

Right： TNode\*，

P： TNode\*

｝

**程序输入：**

文件名： insert.txt

文件格式：第一行为待插入数据的个数，第二行为待插入的数据（int类型， 空格分割）

注：1）初始时红黑树应为空。

2）按顺序插入， 例如，对于下图的数据，插入顺序应为 20，10，14



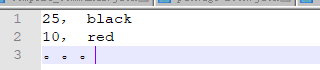
**程序输出：**

print输出：RB\_INSERT\_FIXUP算法历经的情况种类数，即代码注释中 case x 的数字，当 z.p==z.p.p.right时（即代码中被省略的else部分），情况种类数顺延为4,5,6。例：书上13-4图中情况种类数输出应为“1 2 3”。

文件输出：将插入完成后的红黑树进行 “层次遍历（Level-Order Traverse）” 并将相应的遍历序列输出到文件中。

文件名： LOT.txt 层次遍历序列结果

格式：每一行对应一个节点的信息（key， color）, nil 节点输出nil即可.



**实验报告要求：**

1）实验内容

2）实验目的

3）算法设计思路

4）源码 + 注释

5）算法正确性测试

6）实验过程中遇到的困难及收获

**提交方式：**

**提交内容：**

**1、源码**

**2、实验报告（压缩包命名方式为： 学号-姓名-第x次试验.zip**

**例如： SA21011001-张三-第三次试验.zip） （2021年12月1日 23：59：59 前）**