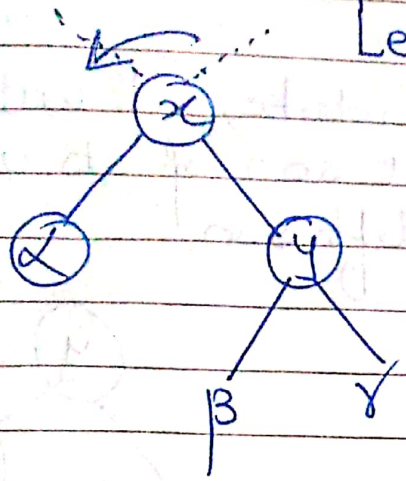
z ← parent of x.

Left Rotate

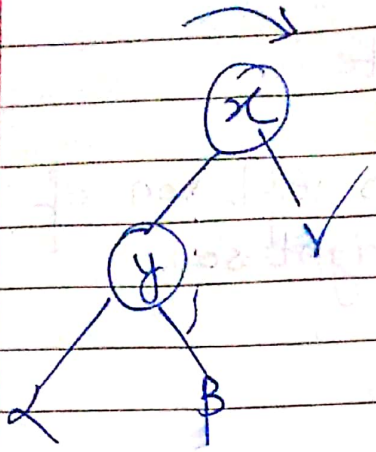


x becomes left son of its current right son.

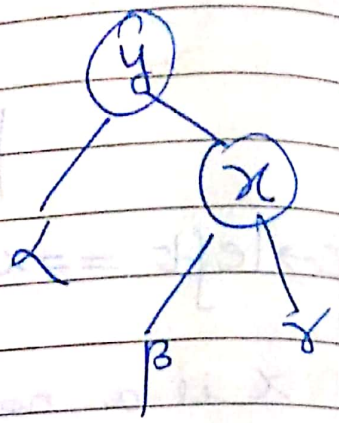
z→left == x or z→right == x.

x is a node which we r rotating.

```
Left_rotate(T, x)
{
    y = x→right;          Notedown x parent and x right son.
    y = y→left;
    z = x→parent;

x→right = v               // Linking y son v to x as right son
v→parent = x

~~y→left = x;~~

y→parent = z;
if z == NULL
    root = y.
else if z→left == x
    z→left = y
                          else
                          z→right = y
                          y→left = x
                          x→parent = y;
}
```



Right rotate, it will become right son of its current left son.

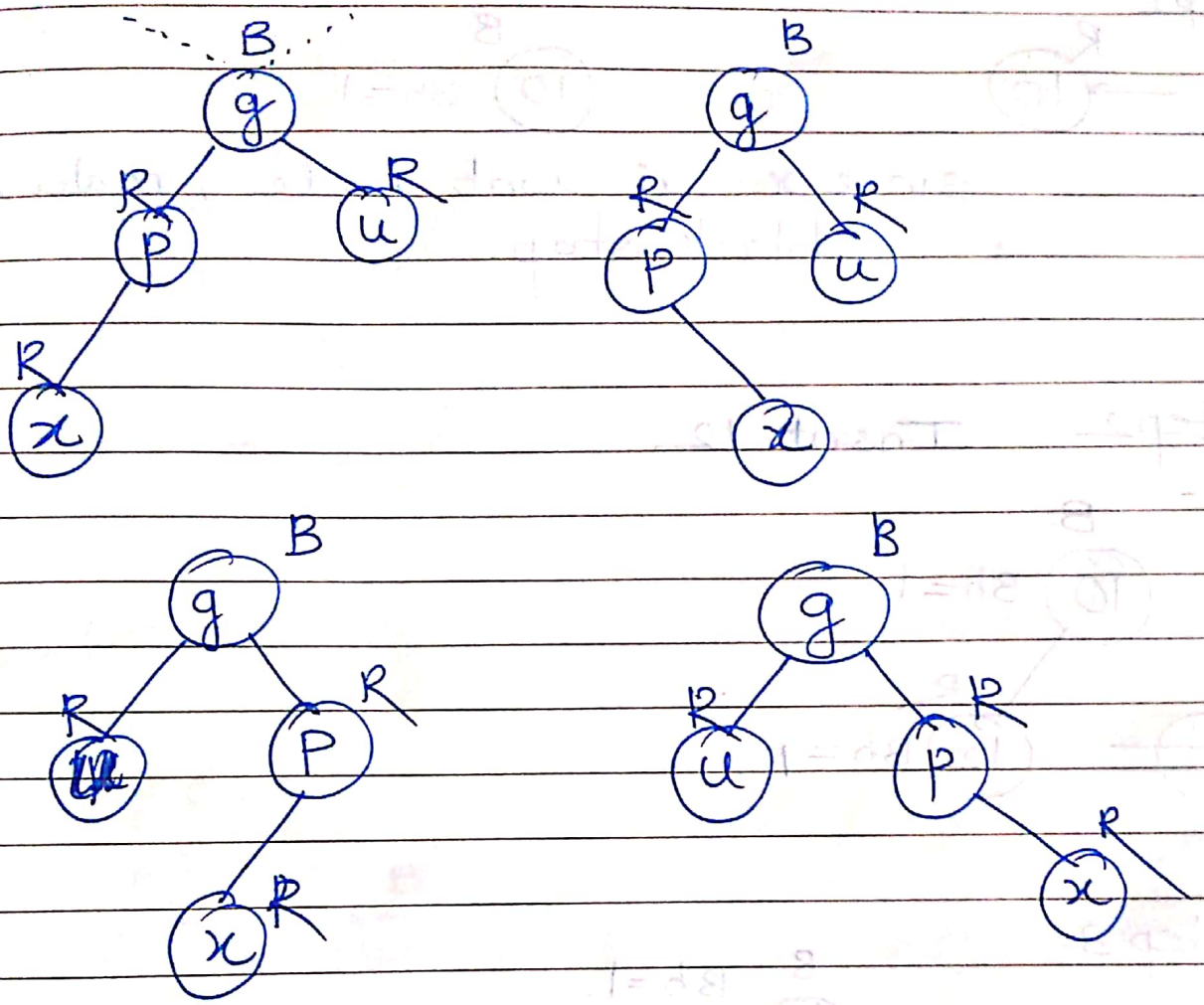



```
Right_rotate (T, x)
{
    z = x->parent
    y = x->left;
    v = y->right;

    x->left = v
    v->parent = x.

    y->parent = z.
    if z == NULL
        root = y
    else if z->left == x
        z->left = y
    else
        z->right = y

    y->right = x
    x->parent = y
}
```

x is a node which is to be adjusted/ inserted.

Violation to Red Rule (x parent is Red)

Case 1 : Uncle is Red.



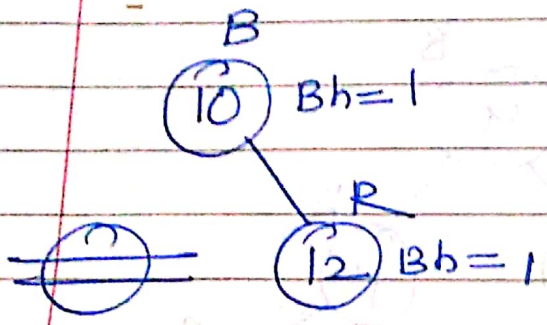
Change Uncle & Parent color to black
change grandparent to red.
If Grandparent is root make it black
& stop.
If grandparent is not root then node to be adjusted is g (recursive step).

Eg 10, 12, 8, 11

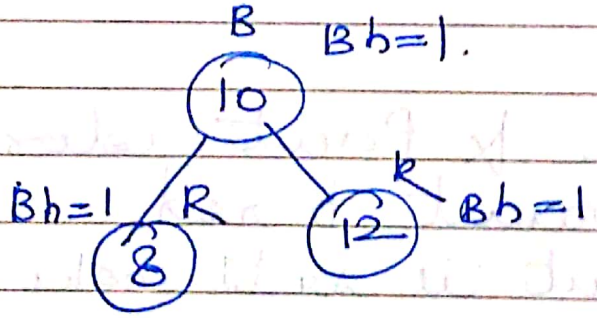
step1

since x is root node, make it black step.

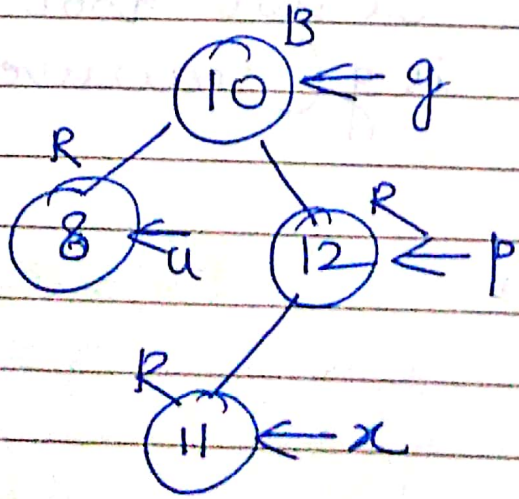
step2 Insert 12



step 3.

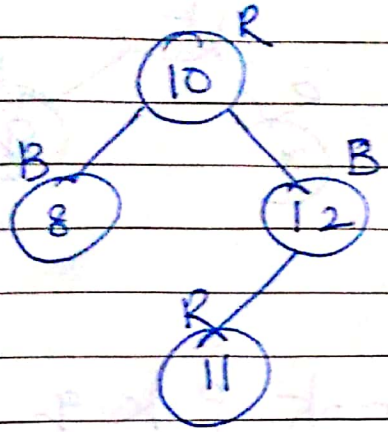


step4.



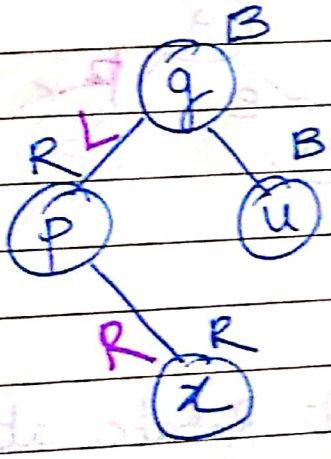
Violation of Red Rule
∵ uncle is Red.

make uncle & parent color black, & grandparent color red.

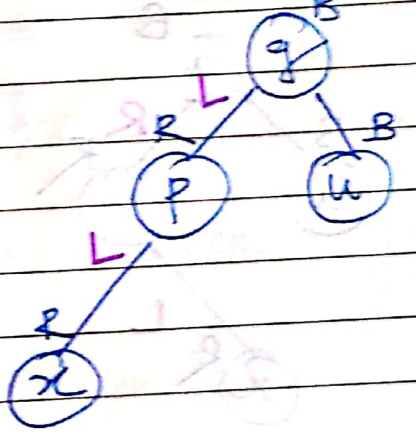


Root color is red, make it black stop.

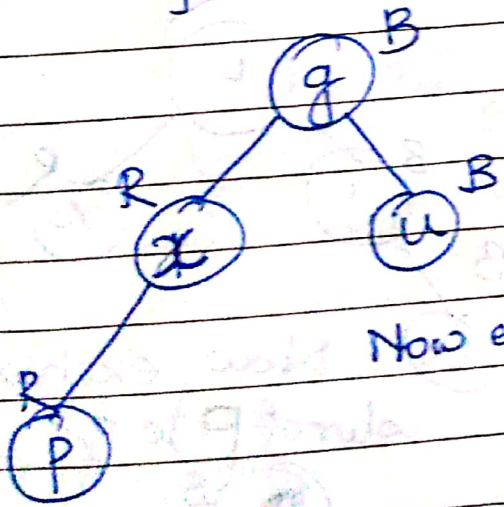
case 2 : Uncle is black (parent is Red)



case 3

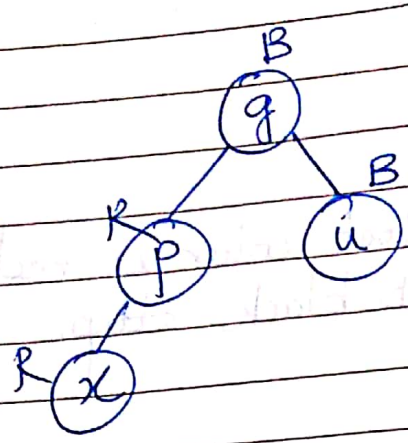


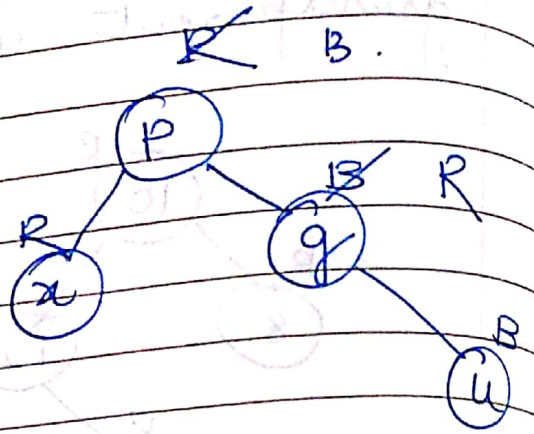
→ Left Rotate at parent



Now exchange roll of x & p.
so it will be transformed to case 3

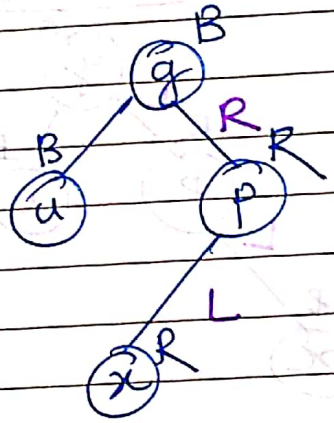
In case 3 Right rotate at grandparent





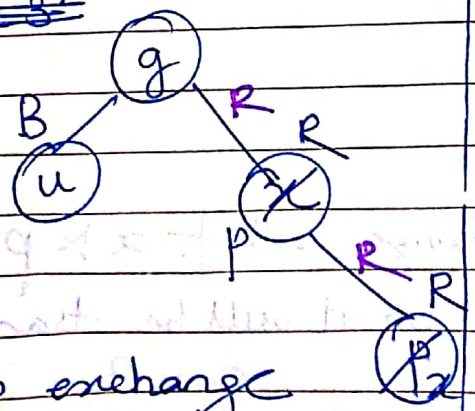
exchange color of p & g

case 2'



~~left~~ Right Rotate p.

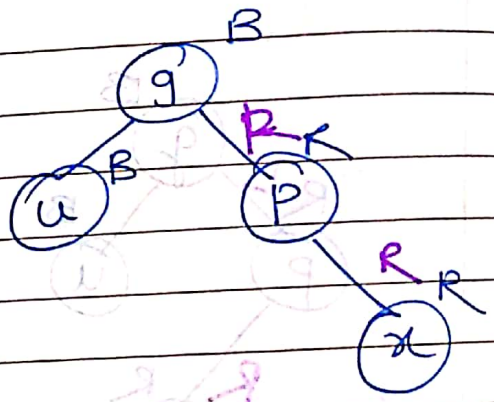
~~case 3'~~



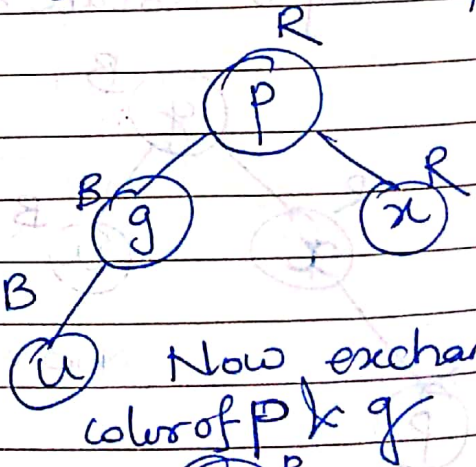
Now exchange ~~col~~ color of x & p.

∴ case 2' transformed into case 3'

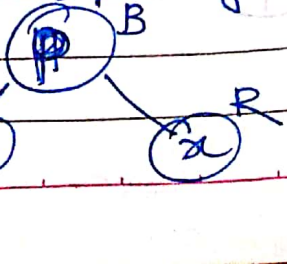
case 3'



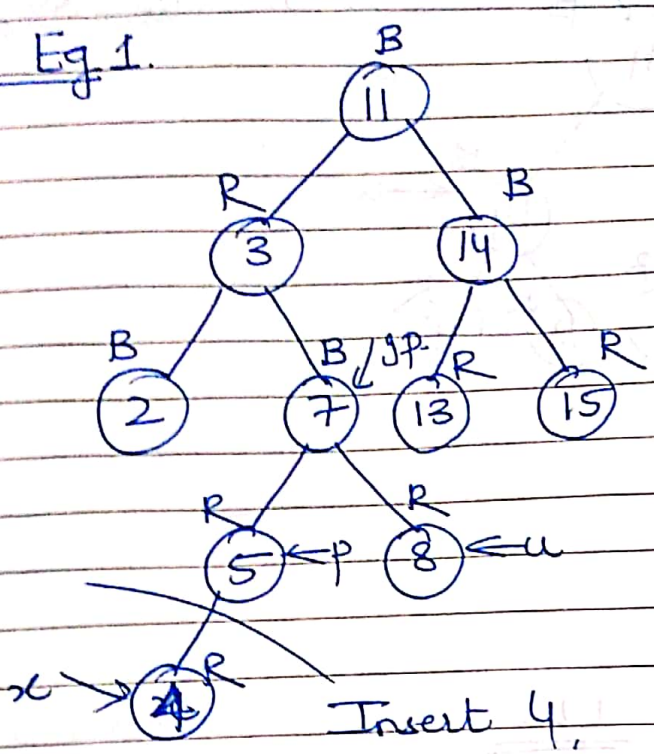
Left Rotate at grand parent



Now exchange color of p & g

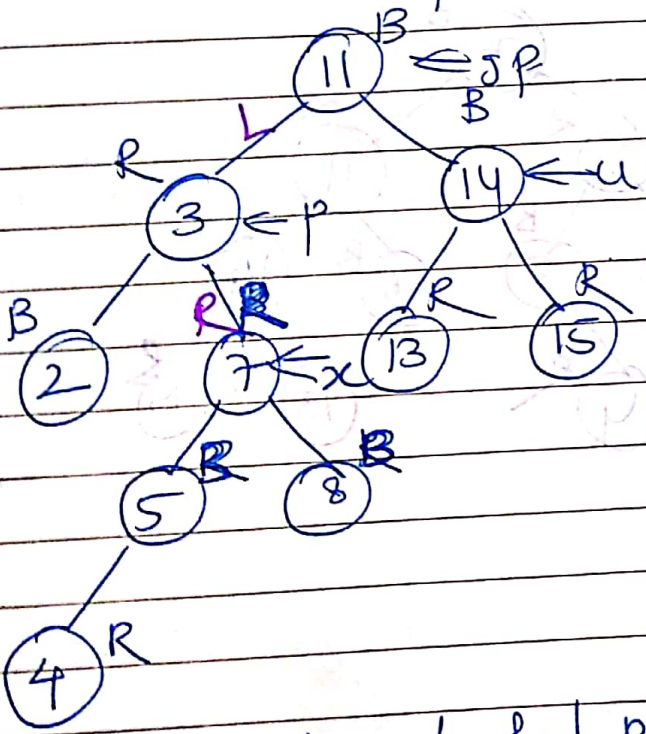


Eg 1.



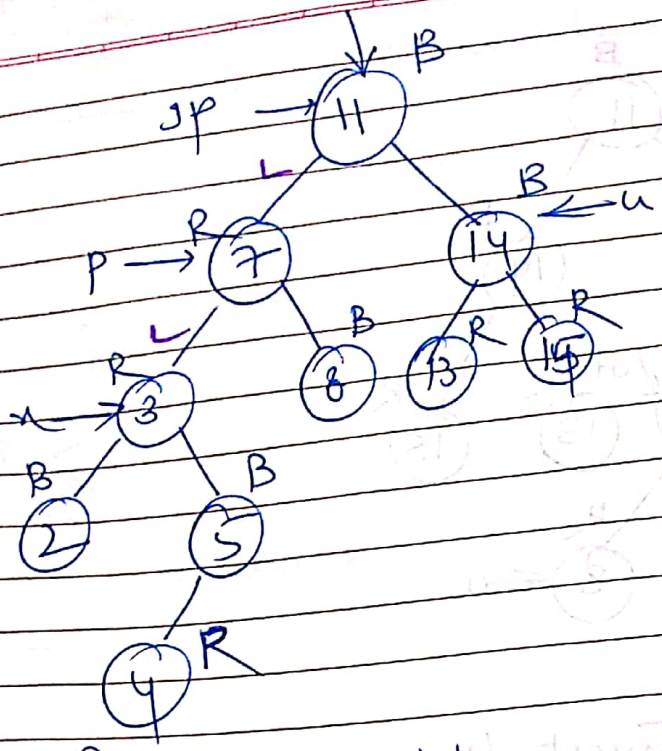
Violation of Red rule & uncle is Red

case 1 — Make p & u is black & grandparent R



Now again violation of Red Rule, uncle is black.

case 2 / Left rotate at parent.



RT case 3 LL

Right rotate gp & grand parent
& exchange color of gp & p.