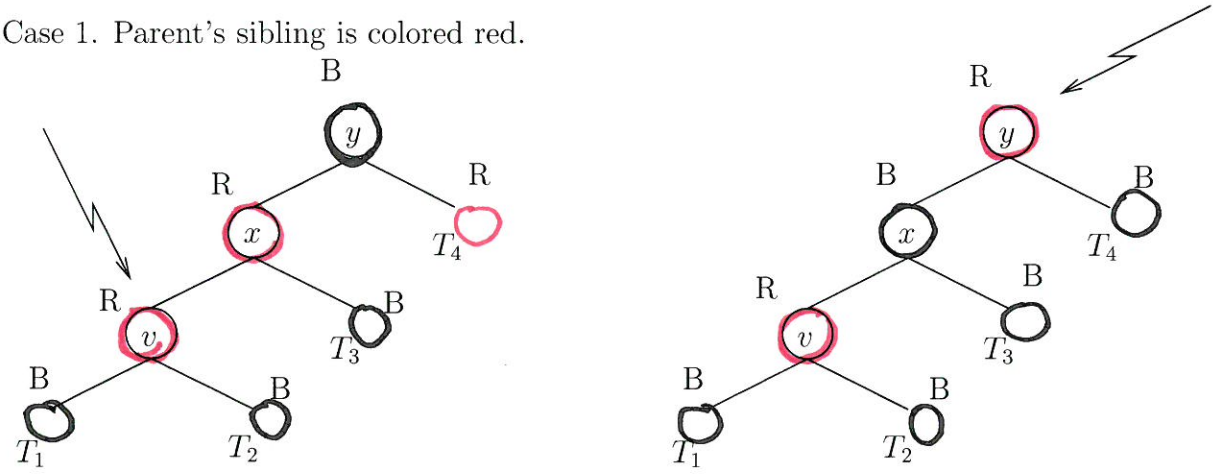
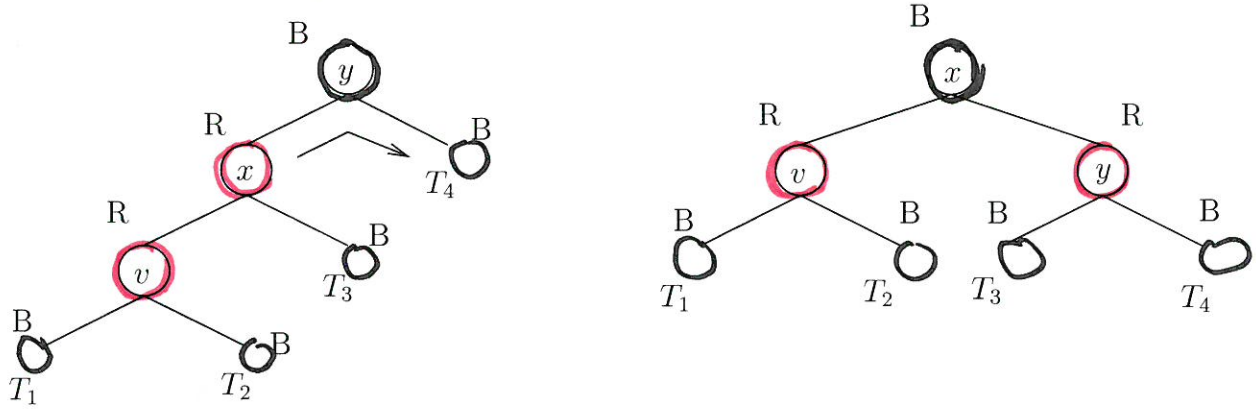
Repair(v): Repair algorithm

Precondition: v and v.parent are Red.

Case 1. Parent's sibling is colored red.



Case 2. Parent's sibling is colored black, v is aligned with grandparent.



Case 3. Parent's sibling is colored black, v is not aligned with grandparent.

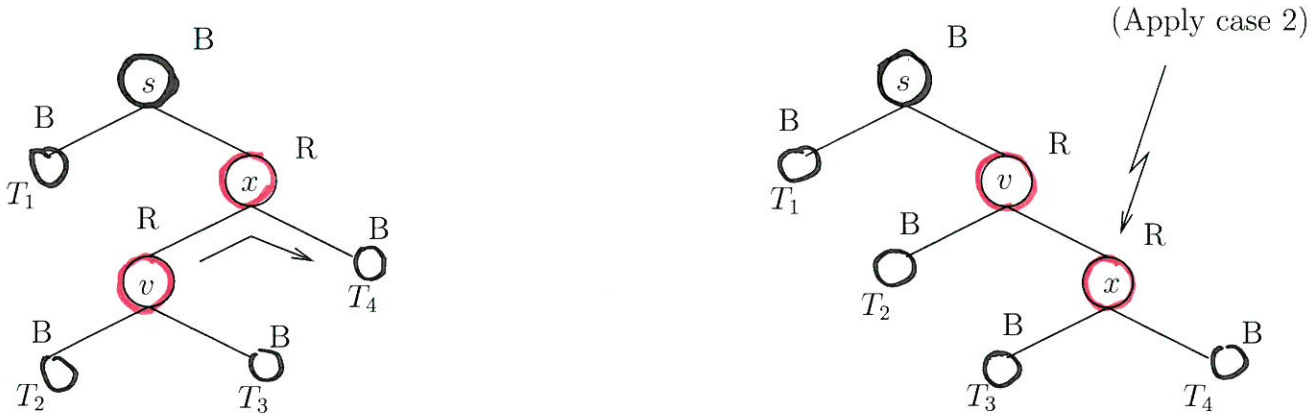
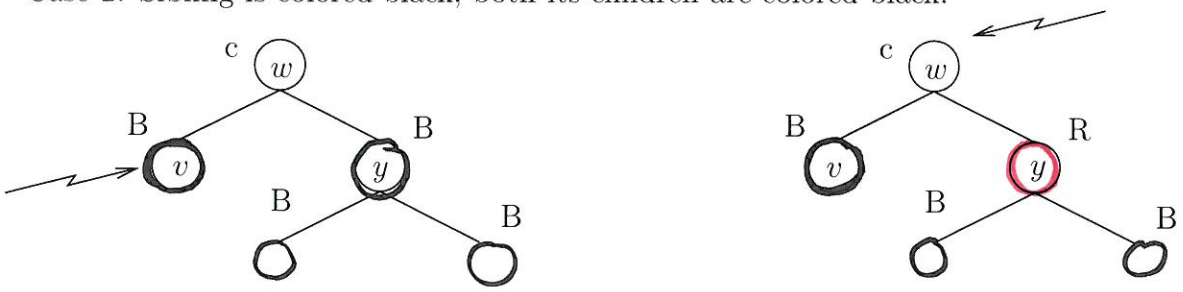


Figure 3: Repairing a Red-Black tree after insertion — 3 cases.

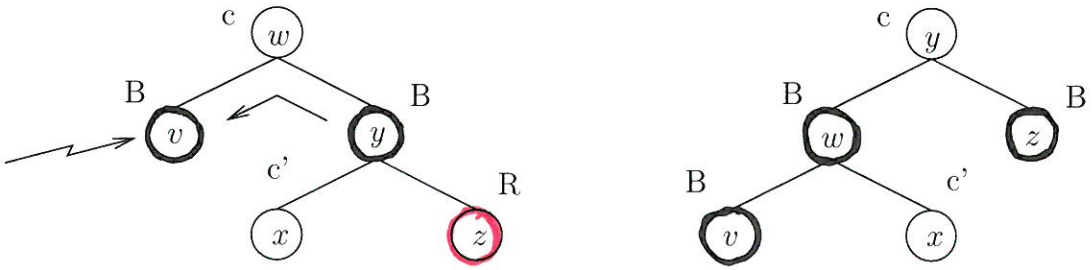
Fix (v): Precondition: v's black height is deficient by 1.

Case 1: v is colored red $\rightarrow$ change its color to black.

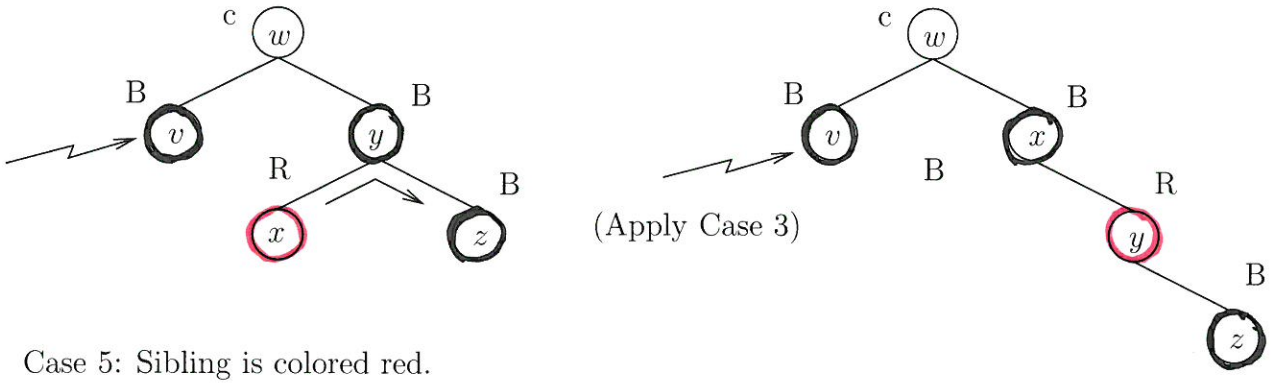
Case 2: Sibling is colored black, both its children are colored black.



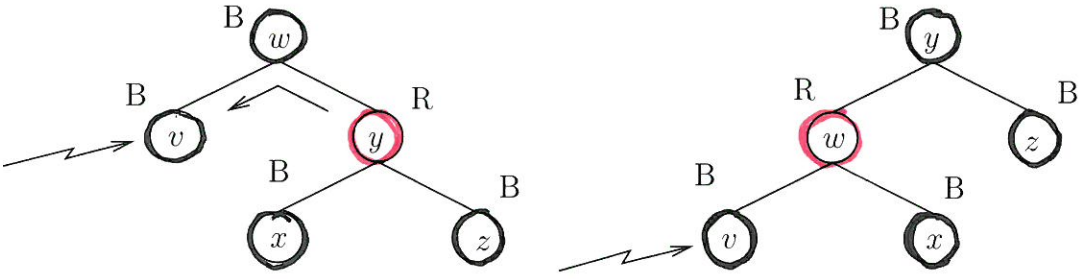
Case 3: Sibling is colored black, its aligned child is colored red.



Case 4: Sibling is colored black, its unaligned child is colored red.



Case 5: Sibling is colored red.



(Apply one of cases 2–4)

Figure 4: Fixing a Red-Black tree after deletion — 5 cases.