

Red-black tree deletion steps

RB-DELETE

- Pick one {
- If the node we deleted has 2 NIL children, its replacement x is NIL.
 - If the node we deleted has 1 NIL child and 1 non-NIL child, its replacement x is the non-NIL child.
 - If the node we deleted has 2 non-NIL children, set x to the replacement's right child before the replacement is spliced out.
-

- Pick one {
- If the node we deleted is red and its replacement is red or NIL, we are done.
 - If the node we deleted is red and its replacement is black and non-NIL, color the replacement red and proceed to the appropriate case.
 - If the node we deleted is black and its replacement is red, color the replacement black. We are done.
 - If the node we deleted is black, its replacement is black, and x is the root of the tree, we are done.
 - If the node we deleted is black, its replacement is black, and x is not the root of the tree, proceed to the appropriate case.
-

RB-DELETE-FIXUP

Cases

0. Node x is red
1. Node x is black and its sibling w is red
2. Node x is black, its sibling w is black, and both of w 's children are black
3. Node x is black, its sibling w is black, and
 - If x is the left child, w 's left child is red and w 's right child is black
 - If x is the right child, w 's right child is red and w 's left child is black
4. Node x is black, its sibling w is black, and
 - If x is the left child, w 's right child is red
 - If x is the right child, w 's left child is red

Case 0: Node x is red

1. Color x black. We are done.
-

Case 1: Node x is black and its sibling w is red

1. Color w black
 2. Color $x.p$ red
 3. Rotate $x.p$
 - a. If x is the left child do a left rotation
 - b. If x is the right child do a right rotation
 4. Change w
 - a. If x is the left child set $w = x.p.right$
 - b. If x is the right child set $w = x.p.left$
 5. With x and our new w , decide on case 2, 3, or 4 from here.
-

Case 2: Node x is black, its sibling w is black, and both of w 's children are black

1. Color w red
 2. Set $x = x.p$
 - a. If our new x is red, color x black. We are done.
 - b. If our new x is black and is the root of the tree, we are done.
 - c. If our new x is black and is not the root of the tree, decide on case 1, 2, 3, or 4 from here. Note that we have a new w now.
-

Case 3: Node x is black, its sibling w is black, and

- If x is the left child, w 's left child is red and w 's right child is black
 - If x is the right child, w 's right child is red and w 's left child is black
1. Color w 's child black
 - a. If x is the left child, color $w.left$ black

- b. If x is the right child, color $w.right$ black
 - 2. Color w red
 - 3. Rotate w
 - a. If x is the left child do a right rotation
 - b. If x is the right child do a left rotation
 - 4. Change w
 - a. If x is the left child set $w = x.p.right$
 - b. If x is the right child set $w = x.p.left$
 - 5. Proceed to case 4.
-

Case 4: Node x is black, its sibling w is black, and

- If x is the left child, w 's right child is red
 - If x is the right child, w 's left child is red
- 1. Color w the same color as $x.p$
 - 2. Color $x.p$ black
 - 3. Color w 's child black
 - a. If x is the left child, color $w.right$ black
 - b. If x is the right child, color $w.left$ black
 - 4. Rotate $x.p$
 - a. If x is the left child do a left rotation
 - b. If x is the right child do a right rotation
 - 5. We are done.