

RED-BLACK TREE

DELETIONS

RULES

①

case 0: If node to be deleted say x is red, delete x and add external node make it black.

case 1: If node to be deleted say x is black, and its sibling(w) is black and both of sibling's children are black

→ color w as red

→ set x as w 's parent (i.e., make parent of x as double node)

→ if new x is red, recolor to black

→ if new x is black, apply any of

the cases 1, 2, 3 or 4

case 2: If node to be deleted say x is black, and its sibling say w is black and

- If x is left child, w 's left child is

- red and w 's right child is black

- If x is right child, w 's right child is

- red and w 's left child is black

→ color w 's child black

→ i) If x is Left child, color w . left
block

ii) If x is right child, color w . right
block

\xrightarrow{b} color w as red

\xrightarrow{c} rotate w

\xrightarrow{i} If x is left child, do right rotation

\xrightarrow{ii} If x is right child, do left rotation

\xrightarrow{d} change w

\xrightarrow{i} If x is the left child, set/make
w as/to right child of x's parent

\xrightarrow{ii} If x is right child, make w as
left child of x's parent

\xrightarrow{e} proceed to case 3.

case 3: If node to be deleted x is black and
its sibling w is black and

- If x is left child, w's right child is red
- If x is right child, w's left child is red

\xrightarrow{a} color w as same color of parent of x

\xrightarrow{b} color parent of x as black

\xrightarrow{c} w's child is black

\xrightarrow{i} If x is the left child, color right child
of w to black

\xrightarrow{ii} If x is the right child, color left child
of w to black

\xrightarrow{d} Rotate parent of x

→ If x is left child, do left rotation

→ If x is right child, do right rotation

case 4: If the node to be deleted is black, and its sibling w is red

a → color w as black

b → color parent of x as red

c → Rotate parent of x

i → If x is left child, do left rotation

ii → If x is right child, do right rotation

d → change w

i → If x is left child, make w as right child of x 's parent

ii → If x is right child, make w as left child of x 's parent

e → With x and new w , apply any one of the above cases

case 5: If node to be deleted x is root apply any one of the above cases.