

balanceDelete(TREE T , TREE t)

$t.color \leftarrow \text{RED}$ // colore il nodo da inserire di rosso

while ($t \neq T$) **and** ($t.color == \text{BLACK}$) **do**

 TREE $p \leftarrow t.parent$ // riferimento al padre

if $t == p.left$ **then**

 TREE $f \leftarrow p.right$ // riferimento al fratello

 TREE $ns \leftarrow f.left$ // riferimento al nipote sinistro

 TREE $nd \leftarrow f.right$ // riferimento al nipote destro

if $f.color == \text{RED}$ **then**

$p.color \leftarrow \text{RED}$

$f.color \leftarrow \text{BLACK}$

 rotateLeft(p)

 // t viene lasciato inalterato, quindi si ricade nei casi 2, 3, 4

else

if $ns.color == nd.color == \text{BLACK}$ **then**

$f.color == \text{RED}$

$t \leftarrow p$ // passo il problema al padre

else if ($ns.color == \text{RED}$) **and** ($nd.color == \text{BLACK}$) **then**

$ns.color \leftarrow \text{BLACK}$

$f.color \leftarrow \text{RED}$

 rotateRight(f)

 // t viene lasciato inalterato, quindi si ricade nel caso 4

else if $nd.color == \text{RED}$ **then**

$f.color == p.color$

$p.color \leftarrow \text{BLACK}$

$nd.color \leftarrow \text{BLACK}$

 rotateLeft(p)

$t \leftarrow T$

else

 // casi speculari