

Pseudocodes der RB-Tree-Funktionen nach Cormen, Algorithms, MIT Press 2nd ed, 2001

RB-Insert(T, z) und RB-Insert-Fixup(T, z)

RB-Delete(T, z) und RB-Delte-Fixup(T, x)

RB-INSERT(T, z)

```

1   $y \leftarrow \text{nil}[T]$ 
2   $x \leftarrow \text{root}[T]$ 
3  while  $x \neq \text{nil}[T]$ 
4      do  $y \leftarrow x$ 
5          if  $\text{key}[z] < \text{key}[x]$ 
6              then  $x \leftarrow \text{left}[x]$ 
7              else  $x \leftarrow \text{right}[x]$ 
8   $p[z] \leftarrow y$ 
9  if  $y = \text{nil}[T]$ 
10     then  $\text{root}[T] \leftarrow z$ 
11     else if  $\text{key}[z] < \text{key}[y]$ 
12         then  $\text{left}[y] \leftarrow z$ 
13         else  $\text{right}[y] \leftarrow z$ 
14   $\text{left}[z] \leftarrow \text{nil}[T]$ 
15   $\text{right}[z] \leftarrow \text{nil}[T]$ 
16   $\text{color}[z] \leftarrow \text{RED}$ 
17  RB-INSERT-FIXUP( $T, z$ )

```

RB-INSERT-FIXUP(T, z)

```

1  while  $\text{color}[p[z]] = \text{RED}$ 
2      do if  $p[z] = \text{left}[p[p[z]]]$ 
3          then  $y \leftarrow \text{right}[p[p[z]]]$ 
4              if  $\text{color}[y] = \text{RED}$ 
5                  then  $\text{color}[p[z]] \leftarrow \text{BLACK}$  ▷ Case 1
6                   $\text{color}[y] \leftarrow \text{BLACK}$  ▷ Case 1
7                   $\text{color}[p[p[z]]] \leftarrow \text{RED}$  ▷ Case 1
8                   $z \leftarrow p[p[z]]$  ▷ Case 1
9              else if  $z = \text{right}[p[z]]$ 
10                 then  $z \leftarrow p[z]$  ▷ Case 2
11                 LEFT-ROTATE( $T, z$ ) ▷ Case 2
12                  $\text{color}[p[z]] \leftarrow \text{BLACK}$  ▷ Case 3
13                  $\text{color}[p[p[z]]] \leftarrow \text{RED}$  ▷ Case 3
14                 RIGHT-ROTATE( $T, p[p[z]]$ ) ▷ Case 3
15             else (same as then clause
                    with “right” and “left” exchanged)
16   $\text{color}[\text{root}[T]] \leftarrow \text{BLACK}$ 

```

RB-DELETE(T, z)

```

1  if left[z] = nil[T] or right[z] = nil[T]
2      then y ← z
3      else y ← TREE-SUCCESSOR(z)
4  if left[y] ≠ nil[T]
5      then x ← left[y]
6      else x ← right[y]
7  p[x] ← p[y]
8  if p[y] = nil[T]
9      then root[T] ← x
10     else if y = left[p[y]]
11         then left[p[y]] ← x
12         else right[p[y]] ← x
13 if y ≠ z
14     then key[z] ← key[y]
15     copy y's satellite data into z
16 if color[y] = BLACK
17     then RB-DELETE-FIXUP(T, x)
18 return y

```

RB-DELETE-FIXUP(T, x)

```

1  while x ≠ root[T] and color[x] = BLACK
2      do if x = left[p[x]]
3          then w ← right[p[x]]
4              if color[w] = RED
5                  then color[w] ← BLACK                ▷ Case 1
6                  color[p[x]] ← RED                    ▷ Case 1
7                  LEFT-ROTATE(T, p[x])                  ▷ Case 1
8                  w ← right[p[x]]                        ▷ Case 1
9              if color[left[w]] = BLACK and color[right[w]] = BLACK
10                 then color[w] ← RED                    ▷ Case 2
11                 x ← p[x]                                ▷ Case 2
12             else if color[right[w]] = BLACK
13                 then color[left[w]] ← BLACK            ▷ Case 3
14                 color[w] ← RED                          ▷ Case 3
15                 RIGHT-ROTATE(T, w)                     ▷ Case 3
16                 w ← right[p[x]]                         ▷ Case 3
17                 color[w] ← color[p[x]]                  ▷ Case 4
18                 color[p[x]] ← BLACK                     ▷ Case 4
19                 color[right[w]] ← BLACK                 ▷ Case 4
20                 LEFT-ROTATE(T, p[x])                    ▷ Case 4
21                 x ← root[T]                             ▷ Case 4
22             else (same as then clause with "right" and "left" exchanged)
23 color[x] ← BLACK

```