

balanceInsert(TREE  $t$ )

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

//  $t == \text{nil}$  è la condizione di fine ciclo

**while**  $t \neq \text{nil}$  **do**

```
(0)   TREE  $p \leftarrow t.parent$  // riferimento al padre
      TREE  $n \leftarrow \text{iif}(p \neq \text{nil}, p.parent, \text{nil})$  // riferimento al nonno
      TREE  $z \leftarrow \text{iif}(n == \text{nil}, \text{nil}, \text{iif}(n.left == p, n.right, n.left))$  // riferimento allo zio

(1)   if  $p == \text{nil}$  then
      |    $t.color \leftarrow \text{BLACK}$ 
      |    $t \leftarrow \text{nil}$  // fine

(2)   else if  $p.color == \text{BLACK}$  then
      |    $t \leftarrow \text{nil}$  // fine

(3)   else if  $z.color == \text{RED}$  then
      |    $p.color \leftarrow z.color \leftarrow \text{BLACK}$ 
      |    $n.color \leftarrow \text{RED}$ 
      |    $t \leftarrow n$  // passo il problema al nonno

      else

4a)   |   if  $(t == p.right) \text{ and } (p == n.left)$  then
      |   |   rotateLeft( $p$ )
      |   |    $t \leftarrow p$  // passo il problema al padre

4b)   |   if  $(t == p.left) \text{ and } (p == n.right)$  then
      |   |   rotateRight( $p$ )
      |   |    $t \leftarrow p$  // passo il problema al padre

      |   else

5a)   |   |   if  $(t == p.left) \text{ and } (p == n.left)$  then
      |   |   |   rotateRight( $n$ )
      |   |   else if  $(t == p.right) \text{ and } (p == n.right)$  then
      |   |   |   rotateLeft( $n$ )

5b)   |   |    $p.color \leftarrow \text{BLACK}$ 
      |   |    $n.color \leftarrow \text{RED}$ 
      |   |    $t \leftarrow \text{nil}$  // fine
```