

```
// INSERIMENTO DI UN NODO
```

```
TREE insertNode(TREE T, ITEM k, ITEM v)
```

```
    TREE p  $\leftarrow$  nil // padre
```

```
    TREE u  $\leftarrow$  T // parto dalla radice
```

```
    // cerco posizione inserimento
```

```
    finché  $u \neq \text{nil}$  and  $u.key \neq k$  fai
```

```
    |   p  $\leftarrow$  u
```

```
    |   u  $\leftarrow$  iif( $k < u.key, u.left, u.right$ )
```

```
    se  $u \neq \text{nil}$  and  $u.key == k$  allora
```

```
    |   // la chiave è già presente, aggiornò il valore
```

```
    |   u.value  $\leftarrow$  v
```

```
    altrimenti
```

```
    |   // la chiave non è presente
```

```
    |   // creo un nodo coppia chiave-valore
```

```
    |   TREE new  $\leftarrow$  Tree( $k, v$ )
```

```
    |   // collegò il nodo creato
```

```
    |   link( $p, new, k$ )
```

```
    |   se  $p == \text{nil}$  allora
```

```
    |   |   T  $\leftarrow$  new // primo nodo ad essere inserito
```

```
    // restituisco l'albero non modificato o il nuovo nodo
```

```
    ritorna T
```