

# Esercizio

Si definisca la funzione C di prototipo

```
int hLSbil(TreePtr t);
```

che restituisce il numero dei nodi sbilanciati a sinistra in altezza, cioè dei nodi X tali che l'altezza del sottoalbero sinistro di X è maggiore di quella del sottoalbero destro.

# Soluzione

```
int hSbil(TreePtr tPtr)
```

```
/*postc: restituisce il numero dei nodi sbilanciati a sinistra in altezza */
```

```
{int hsbil=0;
```

```
hSbilAus(tPtr,&hsbil);return hsbil;}
```

```
int hSbilAus(TreePtr tPtr,int *ris)
```

```
/* postc:restituisce l'altezza di t e calcola in ris il numero dei nodi sbilanciati a sinistra in altezza */
```

```
{int lh,rh;
```

```
if (!tPtr) return -1; /* l'albero vuoto ha altezza -1*/
```

```
lh = hSbilAus(tPtr -> lPtr, ris);
```

```
rh = hSbilAus(tPtr -> rPtr,ris );
```

```
if (lh > rh) {(*ris)++;return lh + 1;} else return rh+1;
```