

Relazione assegnamento di recupero

Francesco Sacco

1 Problemi con il primo test

Il file TTREE/DATI/out1.check.txt da comparare all'output in uscita di test1 è questo:

```
Test 1.1:
Albero vuoto
--> L0 (rr) 12
--> L1 (lf) 45
--> L2 (lf) 56
--> L3 (lf) 78
--> L3 (md) 56
--> L4 (md) 56
--> L3 (rg) 51
--> L1 (rg) 4
--> L2 (lf) 9
--> L3 (md) 9
--> L4 (md) 9
--> L3 (rg) 6
--> L2 (rg) -9
--> L3 (lf) 3
--> L4 (md) 3
--> L3 (rg) -22
Test 1.2:
```

In pratica prima stampa **Test 1.1:**, poi stampa prima un'albero vuoto e poi uno non vuoto attraverso la funzione scritta dal docente `stampa_albero()`, e infine stampa **Test 1.2:**.

Tuttavia la funzione `stampa_albero()` stampa ----- quando l'albero passato come argomento non è vuoto, quindi il file uscito dal **Test1** è così

```
Test 1.1:
Albero vuoto
--> L0 (rr) 12
--> L1 (lf) 45
--> L2 (lf) 56
--> L3 (lf) 78
--> L3 (md) 56
--> L4 (md) 56
--> L3 (rg) 51
--> L1 (rg) 4
```

```
--> L2 (lf) 9
--> L3 (md) 9
--> L4 (md) 9
--> L3 (rg) 6
--> L2 (rg) -9
--> L3 (lf) 3
--> L4 (md) 3
--> L3 (rg) -22
-----
Test 1.2:
```

Visto che c'è questa "linea in più" non mi faceva superare il test, quindi mi sono trovato costretto a modificare il file `TTREE/DATI/out1.check.txt` per metterci la linea mancante.

2 Funzione di scrittura

La funzione `scrivi_albero(FILE * f, nodo_t * root)` stampa una lista di numeri messi in colonna, prima stampa il valore della chiave contenuto nel nodo `*root` e poi agisce ricorsivamente sui suoi sottoalberi¹.

Ho scelto di farla così perché è simile alla funzione di stampa fornita dal docente e rende la funzione di lettura semplice.

Se si vuole sapere che forma ha l'albero basta scriverlo su carta leggendo un elemento della lista per volta e usando un'algoritmo identico alla funzione `inserisci (int x, nodo_t ** proot)`.

¹ l'ordine non è importante, ma comunque prima stampa il sottoalbero sinistro, poi quello centrale e poi quello destro