

Algoritmica – Prova di Laboratorio

Corso A e B

Appello del 05/04/2016

Istruzioni

Risolvete il seguente esercizio prestando particolare attenzione alla formattazione dell'input e dell'output. La correzione avverrà in maniera automatica eseguendo dei test e confrontando l'output prodotto dalla vostra soluzione con l'output atteso. Si ricorda che è possibile verificare la correttezza del vostro programma su un sottoinsieme dei input/output utilizzati. I file di input e output per i test sono nominati secondo lo schema: `input0.txt output0.txt input1.txt output1.txt ...`. Per effettuare le vostre prove potete utilizzare il comando del terminale per la redirectione dell'input. Ad esempio

```
./compilato < input0.txt
```

effettua il test del vostro codice sui dati contenuti nel primo file di input, assumendo che `compilato` contenga la compilazione della vostra soluzione e che si trovi nella vostra home directory. Dovete aspettarvi che l'output coincida con quello contenuto nel file `output0.txt`. Per effettuare un controllo automatico sul primo file input `input0.txt` potete eseguire la sequenza di comandi

```
./compilato < input0.txt | diff - output0.txt
```

Questa esegue la vostra soluzione e controlla le differenze fra l'output prodotto e quello corretto.

Una volta consegnata, la vostra soluzione verrà valutata nel server di consegna utilizzando altri file di test non accessibili. Si ricorda di avvisare i docenti una volta che il server ha accettato una soluzione come corretta.

Esercizio

Il programma deve leggere una sequenza di N coppie *chiave* e *valore*. Le N chiavi sono stringhe distinte di lunghezza al più 100 caratteri, per le quali deve essere costruito un albero binario di ricerca **non** bilanciato. I valori sono interi positivi. Per l'inserimento delle coppie nell'albero si deve rispettare il loro ordine nella sequenza.

Al programma viene data in input una stringa s che si può assumere essere presente tra le chiavi della sequenza. Sia u il nodo dell'albero avente chiave s . Il programma deve indentificare l'intero maggiore tra tutti gli interi dei nodi che si trovano nel sottoalbero radicato in u , nodo u compreso.

L'input è formattato nel seguente modo. La prima riga contiene l'intero N . Seguono poi $2N$ righe, due righe per coppia. La prima riga della coppia contiene la chiave, mentre la seconda contiene il valore. L'ultima riga dell'input contiene la stringa s .

L'output è costituito da una singola riga contenente l'intero identificato dal programma.

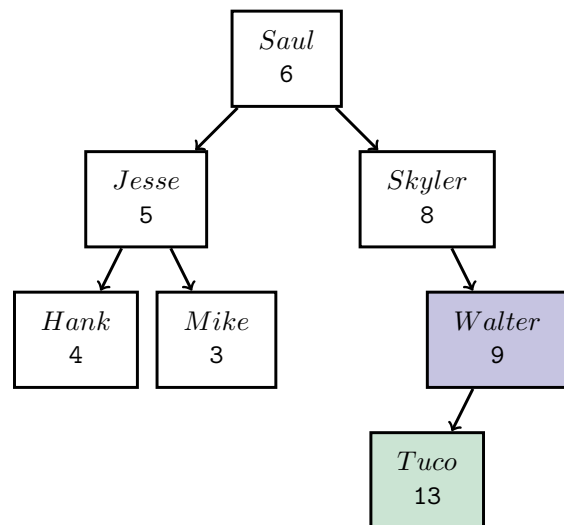
Esempio

Nota: il testo in verde è da intendersi come *commento* e dunque non fa parte dell'input. Nell'albero finale, il nodo relativo alla stringa *s* è evidenziato in blu, mentre quello relativo all'elemento da restituire in output è evidenziato in verde.

Input

```
7      # N
Saul   # Prima chiave
6      # Primo valore
Jesse  # Seconda chiave
5      # Secondo valore
Mike   # etc.
3
Hank
4
Skyler
8
Walter
9
Tuco
13
Walter # Stringa s
```

Albero finale



Output

```
13 # Valore associato a Tuco
```