

Traccia prova d'esame

Dato un albero binario *A* in input, stampare *True* se *A* soddisfa le seguenti proprietà, *False* altrimenti:

- Ogni nodo ha un attributo che può essere *True* oppure *False*;
- l'attributo della radice è *True*;
- ogni foglia ha come attributo *True*;
- se un nodo ha come attributo *False*, allora esistono entrambi i figli ed essi hanno attributo *True*;
- per ogni nodo, tutti i cammini da quel nodo alle rispettive foglie contengono lo stesso numero di nodi con attributo *True*.

Struttura dell'input

L'input consiste in un certo numero di righe. Ogni nodo dell'albero viene etichettato nell'input tramite un numero intero. La prima riga è nel formato *(x) VAL* dove *x* è il numero intero che etichetta la radice dell'albero e *VAL* può essere *True* oppure *False*. Le successive righe sono nel formato *(x) VAL - y d* dove *x* rappresenta un nuovo nodo da inserire nell'albero, *VAL* rappresenta il suo valore (*True* oppure *False*), *y* rappresenta il nodo a cui inserire *x* come figlio e *d* è la direzione in cui inserire, rappresentata dal simbolo *<* oppure *>* indicante, rispettivamente, la direzione sinistra oppure destra. Per esempio, la riga *(10) True - 4 <* indica che il nodo 10, con valore *True*, deve essere inserito come figlio sinistro del nodo 4. L'input è da considerarsi sempre corretto.

Struttura dell'output

L'output della soluzione consiste in una sola riga, contenente la stringa *True* se *A* soddisfa le proprietà di cui sopra, *False* altrimenti.

Esempio input – output

(1) False (2) True - 1 < (3) True - 1 > (4) True - 2 < (5) True - 2 > (6) True - 3 < (7) False - 3 >	False
--	-------

Regole e istruzioni

- Si può scegliere se usare C++ o Java; in entrambi i casi, si presuppone che lo studente sappia compilare il codice sorgente e avviare l'eseguibile ottenuto tramite terminale/prompt dei comandi.
- Si può assumere che l'input sia sempre corretto.
- Il vostro programma deve **leggere da stdin** e **scrivere su stdout**. La lettura da file o l'hardcoding di un input nel vostro programma non rappresenta una soluzione corretta.
- Per effettuare le (vostre) varie prove, potete creare dei file testuali contenenti input di prova e
 - o Scrivere o copiare riga per riga il vostro input al programma,
 - o Reindirizzare il contenuto del file di input al vostro programma (consigliato).
- Il vostro programma verrà valutato su vari input utilizzando il secondo metodo.

Come posso reindirizzare su stdin?

- Supponendo di utilizzare C++ e di essere su Linux/OS X

```
cat input.txt | ./programma
```

dove input.txt è un file testuale contenente un input e programma è l'eseguibile ottenuto dalla compilazione del vostro codice sorgente (input.txt e programma devono essere nella stessa cartella)

- Supponendo di utilizzare C++ e di essere su Windows

```
type input.txt | programma.exe
```

dove input.txt è un file contenete un input e programma.exe è l'eseguibile ottenuto dalla compilazione del vostro codice sorgente (input.txt e programma.exe devono essere nella stessa cartella).