

Algoritmica - Prova di Laboratorio del 16/02/2011

Risolvete il seguente esercizio, prestando particolare attenzione alla formattazione dell'input e dell'output, in quanto la correzione è automatica. Per consegnare un elaborato dovete fornire il codice sorgente attraverso il comando `./consegna` che avete nella vostra home directory. Il comando deve essere utilizzato nel seguente modo:

`./consegna fileSorgente.c numEx`

ricordando che il percorso deve essere specificato a partire dalla vostra home directory e `numEx` deve essere un identificativo numerico (es. 1).

*Il comando **consegna** può essere utilizzato molteplici volte, per cui è possibile sovrascrivere la propria soluzione per un dato esercizio. Di tutte le consegne per un dato esercizio, viene corretta soltanto l'ultima.*

*Il file da consegnare deve contenere nelle prime righe un commento *C* che specifica il vostro Nome, Cognome e Numero di Matricola. Per esempio:*

```
/*
  Nome: Alan
  Cognome: Turing
  Matricola: 193700
*/
```

File non contenenti tali informazioni NON saranno ritenuti validi.

Nota: *E' possibile consultare i manuali (in inglese) contenenti la spiegazione di funzionamento e la sintassi di funzioni di libreria utilizzando il comando **man**, ad esempio:*

man strlen

Esercizio 1

Scrivere un programma che legga da tastiera la descrizione di un albero binario completo a sinistra e determini se si tratta di un albero binario di ricerca.

In un albero binario, il *livello* di un nodo è dato dalla sua distanza dalla radice. Per assunzione la radice si trova a livello 0. Un albero binario è *completo a sinistra* quando tutti i livelli tranne l'ultimo sono riempiti di nodi e l'ultimo è riempito da sinistra verso destra. In questo modo, ogni nodo (tranne uno) ha 0 o 2 figli.

Dato un albero binario completo a sinistra lo scopo del programma è controllare se si tratti di un albero binario di ricerca secondo la convenzione che gli elementi \leq del nodo corrente siano a sinistra, mentre quelli $>$ siano a destra.

L'input è formattato nel seguente modo. La prima riga contiene il numero N di nodi dell'albero binario. L'albero è poi descritto nello stesso modo degli heap, ovvero con una sequenza K di N interi. Si osservino anche la figure a pagina 3 per un esempio.

L'output **deve** contenere **solo e soltanto** un intero: 0 se l'albero non è un albero binario di ricerca, 1 altrimenti. L'output **deve** terminare con un a capo ($\backslash n$) dopo l'ultimo gruppo.

Esempi

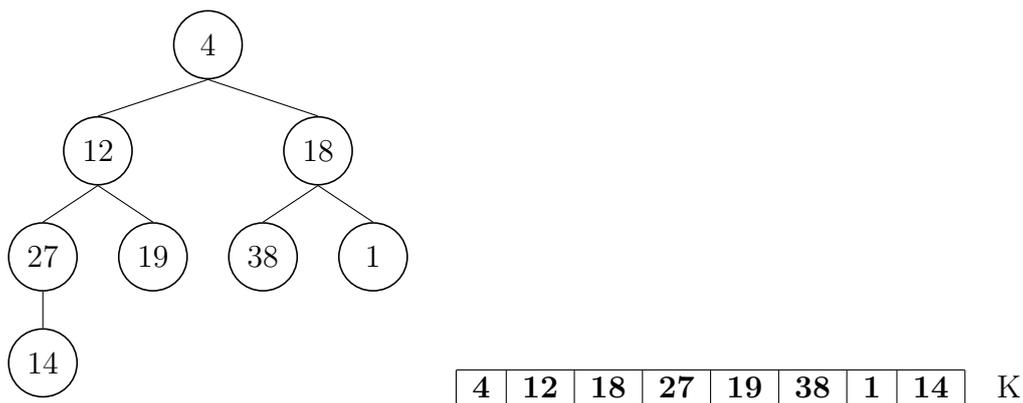


Figura 1: Rappresentazione grafica dell'esempio (soluzione NO)

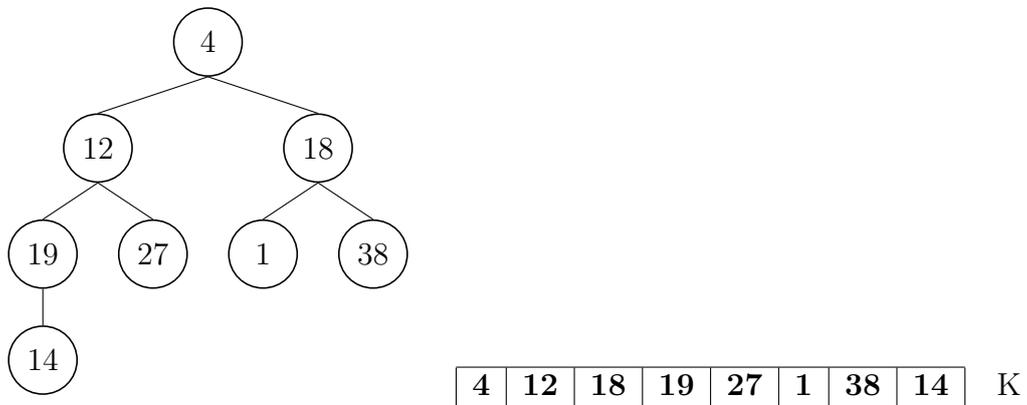


Figura 2: Rappresentazione grafica dell'esempio (soluzione SI)

Input

8
4 12 18 27 19 38 1 14

Output

0

Input

8
4 12 18 19 27 1 38 14

Output

1