

Corso di laurea in Matematica**Insegnamento di Informatica generale****Canale I – Z****Docente: G. Bongiovanni****Secondo esonero****18 giugno 2014****ESERCIZIO 1. (10 punti)**

Progettare una funzione C che, presa in input una lista a chiavi intere, restituisca 1 se esiste un elemento della lista tale che il numero delle chiavi con valore dispari contenute negli elementi in posizione precedente sia uguale al numero delle chiavi con valore dispari contenute negli elementi in posizione successiva, 0 altrimenti.

Nota: il primo elemento della lista si considera preceduto da zero elementi dispari; l'ultimo elemento della lista si considera seguito da zero elementi dispari.

Ad esempio, data la lista:

$8 \rightarrow -2 \rightarrow 5 \rightarrow 6 \rightarrow 7 \rightarrow 8 \rightarrow 1 \rightarrow 4 \rightarrow -||$

la funzione deve restituire 1, in quanto il numero degli elementi con chiave dispari che precedono l'elemento con chiave 7 è uguale al numero degli elementi con chiave dispari che lo seguono.

Dell'algoritmo progettato:

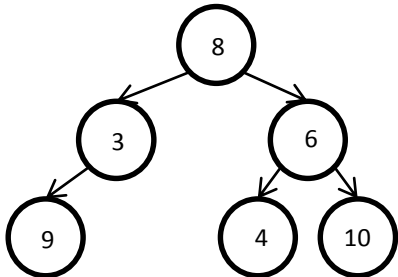
- (2 punti)** si dia la descrizione a parole;
- (6 punti)** si dia il codice C ;
- (2 punti)** si valuti la complessità della soluzione proposta.

ESERCIZIO 2. (10 punti)

Progettare una funzione C ricorsiva che, preso un albero binario non vuoto - implementato tramite puntatori - i cui nodi contengono chiavi intere, ritorni:

- 1 se esiste un cammino dalla radice ad una foglia tale che, lungo tale cammino, la chiave di ogni nodo (esclusa la foglia) sia strettamente maggiore di quella del figlio;
- 0 altrimenti.

Ad esempio, dato l'albero:



La funzione deve restituire 1, perché il cammino $8 \rightarrow 6 \rightarrow 4$ soddisfa la condizione richiesta.

Dell'algoritmo progettato:

- (2 punti) si dia la descrizione a parole;
- (6 punti) si dia il codice C;
- (2 punti) si valuti la complessità della soluzione proposta.

ESERCIZIO 3. (10 punti)

Progettare una funzione C che, presi:

- un vettore A di $n > 0$ numeri interi, tutti diversi fra loro;
- un albero binario di ricerca (ABR) implementato tramite puntatori, completo o quasi completo, contenente $m > 0$ nodi a chiavi intere tutte diverse fra loro;

restituisca 1 se non più della metà dei valori contenuti nell'ABR sono presenti anche nel vettore e zero altrimenti.

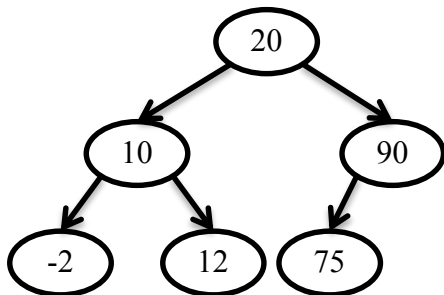
Possono essere definite, se ritenute utili, delle funzioni C ausiliarie.

Possono essere utilizzate, senza bisogno di riportarne il codice C, le funzioni di base viste a lezione per gli ABR.

NOTE:

- il numero m di nodi dell'ABR non è noto a priori;
- la complessità della soluzione proposta non deve essere superiore a $O(m) + O(n \log m)$.

Ad esempio, consideriamo un vettore di $n = 4$ elementi ed il seguente ABR di $m = 6$ nodi:



Se $A = [20, 75, -2, 12]$, la funzione deve restituire 0

Se $A = [20, 12, 4, 1]$, la funzione deve restituire 1

Se $A = [20, 12, -2, 30]$, la funzione deve restituire 1

Dell'algoritmo progettato:

- (2 punti) si dia la descrizione a parole;
- (6 punti) si dia il codice C;
- (2 punti) si valuti la complessità della soluzione proposta.