

Esercitazione 10

Argomento: alberi binari

Scaricare la cartella **Esercitazione 10** e completare le definizioni delle funzioni presenti nei file **esercizio.c**. Compilare e testare il tutto tramite il programma main di prova fornito che si trova all'interno del file **test_esercizio.c**.

Esercizio 10.1

Si consideri l'implementazione del tipo astratto albero binario fornita nei file **esercizio.c** e **esercizio.h**, dove si è scelta una rappresentazione mediante SCL.

Implementare la funzione C

int ricercaLivello(TipoAlbero a, TipoInfoAlbero v, int livello);

che, dati un albero binario **a** e un valore **v**, restituisca il livello dell'albero dove si trova **v**.

Esercizio 10.2

Implementare la funzione C

int verificaNodi(TipoAlbero a, int livello);

Che, dato albero binario, restituisca il numero di nodi di valore pari che si trovano nei livelli pari e di tutti i nodi di valore dispari che si trovano nei livelli dispari.

Esercizio 10.3

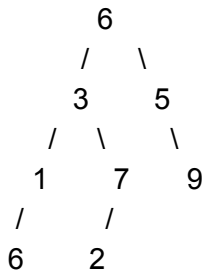
Implementare la seguente funzione C

int singleChildSum(TipoAlbero a);

che, dato in input un albero binario, restituisca la somma dei valori dei nodi che hanno un solo successore.

Esempio

Dato il seguente albero binario **src** in ingresso:



singleChildSum(**src**) dovrà restituire: $1 + 7 + 5 = 13$

Esercizio 10.4

Implementare la seguente funzione C

TipoListaSCL* listaNodiFoglia(TipoAlbero a);

Che, dato un albero binario, restituisca una lista con i valori contenuti nelle foglie (seguendo l'ordine da sinistra a destra).

N.B. TipoListaSCL e le sue funzioni sono disponibili in **esercizio.h** e **esercizio.c**.

Esercizio 10.5

Implementare la seguente funzione C

Coda* codaNodiDueFigli(TipoAlbero a);

Che, dato un albero binario, restituisca una coda contenente i valori dei nodi dell'albero che hanno esattamente due figli.

N.B. Coda e le sue funzioni sono disponibili in **esercizio.h** e **esercizio.c**.

Esercizio 10.6

Implementare la seguente funzione C

TipoListaSCL* listaPercorso(TipoAlbero a);

Che, dato un albero binario, restituisca la lista dei nodi contenuti nel percorso più lungo dalla radice a una delle foglie. Se esistono diversi percorsi di dimensione massima, si restituisca quello più a sinistra.

N.B. Coda e le sue funzioni sono disponibili in **esercizio.h** e **esercizio.c**.