

Esercitazione 14

Liste & Alberi

Alberto Marchesi & Giacomo Boracchi
Informatica A – Ingegneria Matematica (M—Z)

25 Novembre 2020 – 30 Novembre 2020

Gli esercizi visti a lezione il 30 Novembre 2020 sono segnalati con (*).

Gli esercizi segnalati con (*) in rosso sono stati svolti durante la lezione del 25 Novembre 2020 tenuta dal Prof. Giacomo Boracchi.

1 Liste

Esercizio 14.1. (*) Definire una funzione che, data una lista opportunamente definita ed un intero n , inserisce n dopo l'ultimo elemento della lista maggiore di n . Se la lista non contiene alcun elemento maggiore di n , la procedura lascia la lista inalterata.

Esercizio 14.2 (TDE 27/2/2013). (*) Si considerino i seguenti tipi di lista:

```
1 #define N 100
2 typedef struct NodeM{
3     int m[N][N];
4     struct NodeM * next;
5 }NodeM;
6 typedef NodeM * ListaM;
7 typedef struct NodeL{
8     int n;
9     struct NodeL * next;
10 }NodeL;
11 typedef NodeL * ListaL;
```

Si scriva una funzione che, ricevuto in ingresso un intero k e una lista di tipo `ListaM`, restituisca una lista di tipo `ListaL` contenente il più piccolo intero maggiore di k contenuto in ognuna delle matrici (se esiste).

Esercizio 14.3 (TDE 29/01/2009). (*) Si scriva una funzione che calcola la differenza simmetrica degli elementi di due liste ordinate in senso crescente e prive di duplicati, restituendola come una nuova lista (allocata allo scopo), anch'essa ordinata. La differenza simmetrica è costituita dagli elementi che appartengono a una delle due liste ma non all'altra lista (contiene cioè tutti gli elementi che non sono in comune alle due liste).

Esercizio 14.4. (*) Si consideri una lista di interi (tutti diversi tra loro), scrivere la funzione *mediana* avente il seguente prototipo: `int mediana(Lista l)`.

2 Alberi

Esercizio 14.5. (*) Scrivere una funzione che dato un albero binario *ordinato* (albero di ricerca) inserisca un nuovo elemento mantenendo l'albero ordinato.

Esercizio 14.6. (*) (*) Sia dato un albero binario etichettato con numeri interi. Si scrivano le seguenti funzioni ricorsive:

- `int somma(tree t);` (la funzione restituisce la somma delle etichette in `t`).
- `int cercaMax(tree t);` (la funzione cerca il valore dell'elemento massimo di `t`).

Esercizio 14.7 (TDE 26/1/2009). (*) Si scriva una funzione che, dati in input due alberi binari con valori interi, restituisce 1 se i due alberi sono identici, 0 altrimenti.

Esercizio 14.8. Un albero binario si dice *isobato* se tutti i cammini dalla radice alle foglie hanno la stessa lunghezza. Si scriva una funzione che, dato in input un albero binario con valori interi, restituisce 1 se l'albero è isobato, 0 altrimenti.