

Compito di Programmazione
19 Dicembre 2023

Nota Bene. Ogni esercizio deve essere svolto su un foglio diverso.
Scrivere Nome, Cognome e Matricola su ogni foglio.

1. **(punti 8)** Un array memorizza come elementi i nodi di un albero binario non vuoto in cui, nel campo `left` si trova l'indice del sottoalbero di sinistra e nel campo `right` quello del sottoalbero di destra. Quando i valori di questi campi sono `-1` significa che i rispettivi sottoalberi non esistono. I campi `val` dei nodi contengono valori interi. Assumere che la radice si trovi sempre memorizzata ad indice 0.
 - (a) Definire il tipo di dato nodo;
 - (b) definire una funzione che prende in input un albero binario implementato come sopra e ritorna la somma degli interi memorizzati nei nodi;
 - (c) definire una funzione che prende in input un albero binario implementato come sopra e lo modifica sostituendo i campi `val` che contengono un numero pari con il numero dispari successivo.
2. **(punti 8)** Almaesami ha deciso di salvare gli esami degli studenti utilizzando delle liste. Per ogni studente che partecipa ad un esame viene salvato il nome dell'esame (array di `char`), i CFU (`int`) e il voto (`int`). Si implementi la struttura dati necessaria a rappresentare la lista e le seguenti funzioni:
 - (a) `rifiuta_esame` che prende in input la lista di esami e il numero di CFU e cancella dalla lista tutti gli esami che hanno esattamente quei CFU. Non si deve creare una nuova lista, ma va modificata quella passata come parametro. Va restituita la lista modificata.
 - (b) `min_esame` che prende in input la lista di esami e un parametro array di `char` e copia nell'array di `char` passato come parametro il nome dell'esame con il voto più basso.
 - (c) `media` che prende in input la lista di esami e restituisce la media pesata di tutti gli esami. La media pesata viene calcolata come la somma dei prodotti tra voto e CFU di ogni esame diviso per la somma dei CFU di tutti gli esami.

[N.B. È possibile utilizzare le funzioni `strncpy`, `strcpy` e `strcmp` della libreria `cstring`.]

[N.B. Non è possibile restituire puntatori a `char`.]

[N.B. Si possono usare funzioni ausiliarie se definite.]

3. **(punti 8)** Un albergo offre la possibilità di prenotare delle camere da letto. Ogni camera è caratterizzata da un numero identificativo, dal numero di letti e dalla disponibilità.
 - (a) Si rappresenti una camera tramite `struct` e si assuma per semplicità che un albergo abbia N camere.
 - (b) Si implementi la classe `Albergo`, il relativo costruttore e il metodo `prenota()` il quale a seconda del numero di letti richiesti prenota la prima camera libera con esattamente quel numero di letti e restituisce: (i) il numero della camera e (ii) se l'operazione è andata a buon fine.

Esistono anche alberghi che si affacciano sulla spiaggia i quali hanno a disposizione degli ombrelloni. Ogni ombrellone è caratterizzato da un numero identificativo e dalla disponibilità. Si rappresenti un ombrellone tramite `struct` e si assuma per semplicità che un albergo con spiaggia abbia M ombrelloni. Gli ombrelloni sono più delle camere ($M > N$) e possono essere prenotati anche senza camera, però l'albergo garantisce sempre un ombrellone per ogni camera.

- (c) Si implementi la classe `AlbergoConSpiaggia` il relativo costruttore e il metodo `prenota_ombrellone()` il quale prenota un ombrellone se disponibile. Si sfrutti opportunamente l'ereditarietà in modo tale che per ogni camera prenotata di un albergo con spiaggia venga anche prenotato un ombrellone. Per semplicità non è necessario associare il numero della camera al numero dell'ombrellone e/o viceversa.

[N.B. In tutto l'esercizio 3 non è consentito usare le liste.]