

Fondamenti di Informatica T1

Esempio Prova Pratica

Compito

Prima di cominciare: si scarichi il file **StartKit.zip** contenente i file di esempio.

Avvertenze per la consegna: nominare i file sorgenti come richiesto nel testo del compito, apporre all'inizio di ogni file sorgente un commento contenente i propri dati (**cognome, nome, numero di matricola**) e il **numero** della prova d'esame. Al termine, **consegnare tutti i file sorgente ed i file contenuti nello StartKit.**

Rispettare le specifiche, in particolare inserire le funzioni nei file specificati fra parentesi dopo il nome della funzione. Chi non rispetta le specifiche sarà opportunamente penalizzato. **NON SARANNO CORRETTI** gli elaborati che presenteranno un numero "non ragionevole" di errori di compilazione.

Consiglio: per verificare l'assenza di *warnings*, effettuare di tanto in tanto un *Rebuild All*.

La biblioteca *SoulFood* ha deciso di servirsi di un sistema informatizzato per semplificare la gestione dei libri in suo possesso e dei prestiti fatti ai suoi clienti.

A tal proposito, in un file di testo chiamato "*libri.txt*" sono memorizzati tutti i libri di proprietà della biblioteca, dove ogni riga corrisponde ad un singolo libro ed è formata dai seguenti campi:

codice titolo autore genere anno copie

dove codice è una stringa di 4 caratteri che indica in maniera univoca un libro, titolo è una stringa di al massimo 40 caratteri senza spazi, l'autore è un'altra stringa senza spazi di al massimo 30 caratteri, il genere è un singolo carattere che può assumere i valori 'P', 'N' e 'T' (rispettivamente Poesia, Narrativa e Teatro), l'anno è un intero che rappresenta l'anno di pubblicazione del libro in questione e copie è un altro intero che indica il numero di copie totali in possesso della biblioteca (e non il numero di copie effettivamente disponibili al pubblico, poiché questo varia in funzione anche di eventuali prestiti). Nel file "*libri.txt*" sono presenti al massimo 30 libri.

In un altro file di testo denominato "*prestiti.txt*" sono invece salvate le informazioni relative ai libri attualmente prestati – si suppone che ogni volta che un libro viene restituito da questo file sia automaticamente rimossa la riga relativa – formato in questo modo:

codice cliente giorno mese anno

dove il codice è ancora una stringa di 4 caratteri che identifica un libro in maniera univoca, cliente indica il nome del cliente che ha effettuato un prestito (massimo 20 caratteri, senza spazi), mentre giorno, mese e anno sono tre interi che indicano la data entro la quale il libro dovrebbe essere riconsegnato (si suppone che un sia possibile prendere in prestito una sola copia di un determinato libro per volta); un utente può chiedere in prestito al massimo tre libri. Nel file "*prestiti.txt*" sono presenti al massimo 20 prestiti.

Esercizio 1 - Lettura da file (*biblioteca.h/biblioteca.c*)

Si definisca un'opportuna struttura dati **Libro** al fine di rappresentare i libri messi a disposizione dalla biblioteca tenendo traccia delle necessarie informazioni.

Il candidato realizzi poi la funzione:

Fondamenti di Informatica T1

Esempio Prova Pratica

Compito

int leggiLibri (char* nomefile, Libro libri[])

che, ricevuto come parametro di ingresso il nome di un file, provveda ad aprirlo e ne legga il contenuto, ovvero i libri messi presenti nella biblioteca. Non è noto a priori quanti libri siano registrati nel file, ma è noto il numero massimo di libri che possono esservi (30). Tale funzione deve restituire la dimensione logica dell'array.

Si realizzi inoltre una procedura:

void stampaLibro (Libro l)

che stampi a video (in maniera leggibile – usare spazi, separatori, fine linea, etc.) le informazioni relative ad un singolo libro preso come ingresso dalla procedura.

Procedendo in maniera simile al punto precedente, definire un'opportuna struttura dati **Prestito** per rappresentare i prestiti fatti ai clienti della biblioteca, struttura che ovviamente contenga tutte le informazioni descritte in precedenza.

Realizzare quindi la funzione:

int leggiPrestiti (char* nomefile, Prestito prestiti[])

che provveda a leggere da un file di testo (il cui nome è passato come parametro di ingresso) i prestiti attualmente attivi; anche in questo caso non è dato sapere a priori il numero di righe presenti nel file di testo. Nuovamente, restituire la dimensione logica del vettore tramite il valore di ritorno.

Analogamente al caso precedente, definire una procedura che stampi a video tutte le informazioni relativa ad un prestito:

void stampaPrestito (Prestito p)

Il candidato abbia cura di realizzare nel main opportuni test al fine di verificare il corretto funzionamento delle funzioni di cui sopra, sfruttando il file di testo "*libri.txt*" e "*prestiti.txt*" (presenti nello **StartKit.zip**). Una volta verificato il corretto funzionamento delle funzioni, il candidato non cancelli il codice nel main ma si limiti a commentarlo.

Esercizio 2 – Calcola prestiti in ritardo ([biblioteca.h](#)/[biblioteca.c](#))

Il candidato definisca una procedura:

int calcolaRitardi(Prestito p[], int dimP)

che preso in ingresso un vettore **p** di strutture dati **Prestito** e la sua dimensione logica, stampi a video i prestiti in ritardo, ovvero quelli che dovrebbero essere stati già riconsegnati in data odierna ma risultano invece ancora in possesso dei clienti.

La data odierna deve essere letta come input da tastiera (occorre cioè chiedere all'utente i tre interi corrispondenti a giorno, mese e anno corrente).

La funzione deve anche restituire in numero di prestiti in ritardo trovati.

Si realizzino nel main le opportune istruzioni per verificare il corretto funzionamento della funzione.

Fondamenti di Informatica T1

Esempio Prova Pratica

Compito

Esercizio 3 – Ordinamento Libri ([biblioteca.h/biblioteca.c](#))

Si realizzi una procedura:

void ordina (Libro libri[], int dimL)

che preso in ingresso un vettore di libri e relativa dimensione logica ordini i libri contenuti nel vettore in ordine crescente di anno di pubblicazione. Si usi uno degli algoritmi di sorting visti a lezione.

Come sempre, si realizzino nel main le opportune istruzioni per verificare il corretto funzionamento della funzione.

Esercizio 4 – Nuovo prestito ([biblioteca.h/biblioteca.c](#))

Si realizzi una funzione:

int inserisciPrestito (FILE * fp , Libro l[], int dimL, Prestito p[], int dimP)

che, ricevuto in ingresso il puntatore ad un file già opportunamente aperto, provveda a creare un nuovo prestito prendendo in input da tastiera il titolo del libro (non il codice) e il nome del cliente e successivamente scriva le informazioni relative a questo prestito nel file indicato dal parametro di ingresso (mantenendone il formato visto in precedenza: codice libro, cliente, giorno e mese della scadenza del prestito); la funzione dovrà ritornare un intero che può assumere due valori, '0' nel caso in cui la scrittura sia terminata con successo e '-1' in caso contrario.

Occorre anche leggere da tastiera la data entro la quale il libro dovrà essere restituito – necessaria per creare il nuovo **Prestito**.

La funzione deve controllare che ci siano delle copie disponibili per il libro in questione e che il cliente non abbia già preso in prestito altri tre libri; per questo motivo essa prende in ingresso anche un vettore di strutture di tipo **Libro** e uno di strutture di tipo **Prestito**, con le rispettive dimensioni, i quali rappresentano lo stato del sistema prima dell'inserimento di un nuovo prestito (non è necessario aggiornare anche il vettore dei prestiti preso in ingresso, ma esclusivamente il file di testo).

Il candidato realizzi nel main opportuni test per verificare il corretto funzionamento delle funzioni implementate, scrivendo il nuovo prestito nel file di testo "*prestiti.txt*" (quello usato anche in fase di lettura).

Esercizio 5 – Main ([main.c](#))

Il candidato realizzi un programma in grado di:

- 1) Leggere i libri offerti dalla biblioteca e i prestiti attivi al momento dai rispettivi

Fondamenti di Informatica T1

Esempio Prova Pratica

Compito

file creando due array.

- 2) Calcolare e mostrare a video tutti i prestiti in ritardo.
- 3) Stampare a video i libri ordinati in base all'anno di pubblicazione.
- 4) Inserire nel file dei prestiti un nuovo prestito.