

Fondamenti di Informatica e Laboratorio T-AB ETA
Prova Pratica - 31 Maggio 2011
Compito D

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 **codice** 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*.

L'ente benefico "C'era una Volta" dispone di un avanzato sistema informatizzato che aiuta il personale nella gestione delle donazioni. L'ente si occupa di aiuti in vari ambiti sociali e accetta due tipi di donazioni: le generiche e le specifiche. La donazione generica lascia all'ente decidere in che settore investirla, mentre quella specifica precisa una particolare preferenza. Il sistema mantiene in un file di testo i dati principali dei "clienti" che hanno donato. Ogni riga di tale file corrisponde ad un cliente ed è formalizzata nel seguente modo:

codiceCliente nome cognome annoNascita

dove il codiceCliente è una stringa di 5 caratteri, il nome e cognome sono stringhe (senza spazi) di al massimo 30 caratteri, mentre annoNascita è un valore intero.

Esercizio 1 - Lettura dei dati (rubrica.h/rubrica.c)

Si definisca un'opportuna struttura dati, al fine di rappresentare i dati registrati nel file "**clienti.txt**", come specificato sopra, e si realizzi una funzione:

Cliente* readclienti(char* fileName, int *dim);

che, ricevuto in ingresso il nome di un file contenente i dati dei clienti, legga i dati relativi, allochi la quantità di memoria necessaria a contenerli e ve li memorizzi. Tale funzione deve restituire un riferimento all'area di memoria allocata (array) e la sua dimensione logica (numero di clienti letti) in **dim**. Si noti che non è noto a priori quanti clienti siano presenti nel file: sarà quindi necessario determinare quanti ve ne siano, e poi allocare la memoria necessaria dinamicamente.

Si realizzi in seguito la funzione

void stampaCliente(Cliente c);

che, ricevuto in ingresso un **Cliente**, ne stampi a video i dati.

Si realizzino nel main le opportune istruzioni per verificare il corretto funzionamento delle funzioni realizzate.

Esercizio 2 - interrogazioni del database (rubrica.h/rubrica.c)

Inoltre, il sistema tiene memorizzato in un file unico di testo "**donazioni.txt**" l'elenco delle donazioni. Ogni riga del file (vedi start kit) identifica una donazione ed è composta nel seguente modo:

codiceSettore codiceCliente valore

dove il codiceSettore è un valore numerico intero compreso tra 0 (donazione generica) e 4 ed il valore è l'importo intero della donazione.

Si realizzi poi una funzione

Offerta *leggiOfferte(char* fileName, int *dim);

che, ricevuto in ingresso il nome di un file contenente i dati, legga i dati, allochi la quantità di memoria necessaria a contenerli e ve li memorizzi. Tale funzione deve restituire un riferimento all'area di memoria allocata (array) e la sua dimensione logica (numero di opinioni lette) in **dim**.

Fondamenti di Informatica e Laboratorio T-AB ETA
Prova Pratica - 31 Maggio 2011
Compito D

Si realizzi anche una procedura

void offerta(int settore, Cliente c, int valore);

che inserisca nel file corrispondente un'offerta caratterizzata dai parametri in ingresso alla funzione. L'inserimento deve essere effettuato senza ovviamente sovrascrivere le offerte di altri clienti.

Si realizzi in seguito la procedura

void stampaDonazione(Offerta d, Cliente[] clienti, int dimClienti);

che, ricevuto in ingresso un' **Offerta**, stampi a video i dati dell'offerta. Si richiede, però, che al posto del codice cliente siano stampati a video Nome e cognome dello stesso. Tali dati si ricavano dall'array (la cui dimensione è dimClienti) di clienti passato come parametro.

Si realizzino nel main le opportune istruzioni per verificare il corretto funzionamento delle funzioni realizzate.

Esercizio 3 - Suddivisione Offerte (function.h/function.c)

Si realizzi inoltre una funzione

int donazioniSettore(Offerta[] o,int dimOff, Cliente[] clienti, int dimClienti, int anno);

che prenda in ingresso un array di **Offerta** (di dimensione **dimOff**) , i clienti con la relativa dimensione ed un intero **anno**. Tale funzione deve calcolare inizialmente la somma delle donazioni effettuate da clienti aventi un'età compresa nell'intervallo [anno-5, anno+5]. Se ad esempio l'intero anno avesse valore 1971, sarebbero compresi nell'intervallo tutti i nati tra il 1966 ed il 1976.

Il candidato scriva le opportune istruzioni nel main per testare la funzione e stampi a video l'array ottenuto utilizzando l'opportuna procedura.

Esercizio 4 - premiazione clienti (function.h/function.c)

Per incentivare le offerte, l'ente ha previsto un meccanismo di premiazione al cliente avente il maggior numero di donazioni che superano un certo valore (**soglia**).

Pertanto, il candidato realizzi una procedura:

void premio(Offerta[] o, int dimOff, Cliente[] c, int dimCliente, int soglia);

che, presi l'elenco delle offerte e dei clienti con le relative dimensioni, cerchi e stampi a video il cliente con il numero di offerte (che superano la soglia) più elevato.

Il candidato scriva le opportune istruzioni nel main per testare questa funzione.

Esercizio 5 - Ammontare delle donazioni (function.h/function.c)

Il candidato realizzi una funzione

int ammontareDonazioni(Offerta[] o, int dim);

che prenda in ingresso l'elenco delle donazioni **Offerta** e la relativa dimensione **dim**. La funzione deve restituire la somma delle quantità delle offerte. La procedura deve anche stampare a video la somma delle offerte per ogni differente settore [codiceSettore da 1 a 4]. Si richiede che il candidato implementi tale funzione leggendo l'array **Offerta** il numero minimo di volte necessario.

Il candidato scriva le opportune istruzioni nel main per testare questa funzione.