

Fondamenti di Informatica T1
Prova Pratica – 31/05/2017
Compito A

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 stazione radiofonica *IRadio* sfrutta un sistema informatizzato per automatizzare la programmazione giornaliera.

A tal proposito, in un file di testo chiamato "*canzoni.txt*" sono memorizzati tutte le canzoni disponibili negli archivi; ogni riga corrisponde ad una canzone ed è formata dai seguenti campi:

codice titolo artista genere anno minuti secondi

dove codice è una stringa di 5 caratteri che indica in maniera univoca una canzone, titolo è una stringa di al massimo 25 caratteri senza spazi, l'artista è una stringa senza spazi di al massimo 40 caratteri, il genere è un singolo carattere che può assumere i valori 'R', 'E', 'J', 'C' (rispettivamente Rock, Elettronica, Jazz e Classica), l'anno è un intero che rappresenta l'anno di uscita della canzone in questione; minuti e secondi sono due interi che indicano la durata della canzone. Nel file "*canzoni.txt*" sono presenti al massimo 40 canzoni.

Gli ascoltatori della radio possono anche richiedere particolari canzoni, sempre tramite il sistema informatizzato. Ogni ascoltatore può effettuare più di una richiesta. In un altro file di testo denominato "*richieste.txt*" sono quindi salvate le informazioni relative alle richieste effettuate dai radioascoltatori – ogni riga del file è composta dai seguenti campi:

nome_ascoltatore titolo dedica

dove il titolo è nuovamente una stringa di al massimo 25 caratteri senza spazi che indica la canzone richiesta, nome_ascoltatore è il nome dell'ascoltatore di massimo 40 caratteri e dedica è una stringa di al massimo 50 caratteri senza spazi. Ogni richiesta è identificata in maniera univoca dalla coppia <nome_ascoltatore, titolo>: un ascoltatore non può richiedere la stessa canzone più di una volta. Non è detto che tutte le canzoni richieste dagli utenti siano disponibili nella playlist della radio. Nel file "*richieste.txt*" sono presenti al massimo 30 richieste.

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

Si definisca un'opportuna struttura dati **Canzone** che rappresenti una canzone.

Il candidato realizzi poi la funzione:

int leggiCanzoni (char* nomefile, Canzone playlist[])

che, ricevuto come parametro di ingresso il nome di un file, provveda ad aprirlo e ne

Fondamenti di Informatica T1
Prova Pratica – 31/05/2017
Compito A

legga il contenuto, ovvero le canzoni a disposizione della radio. Non è noto a priori il numero esatto di canzoni nel file, ma è noto il loro numero massimo (40). Tale funzione deve restituire la dimensione logica dell'array.

Si realizzi inoltre una procedura:

void stampaCanzone (Canzone c)

che stampi a video (in maniera leggibile – usare spazi, separatori, fine linea, etc.) le informazioni relative ad una canzone presa come ingresso dalla procedura.

Procedendo in maniera simile al punto precedente, definire un'opportuna struttura dati **Richiesta** per rappresentare una richiesta fatta dagli ascoltatori, struttura che ovviamente contenga tutte le informazioni descritte in precedenza.

Realizzare quindi la funzione:

int leggiRichieste (char* nomefile, Richiesta richieste[])

che provveda a leggere da un file di testo (il cui nome è passato come parametro di ingresso) le richieste fatte fino ad ora; anche in questo caso non è dato sapere a priori il numero di righe presenti nel file di testo ma il loro numero massimo (e quindi la dimensione massima dell'array di richieste) è noto e pari a 30. 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 relative ad una richiesta:

void stampaRichiesta (Richiesta r)

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 "*canzoni.txt*" e "*richieste.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 – Richieste Ascoltatori (radio.h/radio.c)

Il candidato definisca una procedura:

**int richAscoltatore(
Canzone c[], int dimC,
Richiesta r[], int dimR,
char * ascoltatore)**

che preso in ingresso un vettore **c** di strutture dati **Canzone** e la sua dimensione logica, un vettore **r** di strutture dati **Richiesta** e la sua dimensione logica ed il nome di un ascoltatore della radio calcoli se è possibile soddisfare TUTTE le richieste dell'ascoltatore passato in come argomento alla funzione. Si supponga, per semplicità, che l'ascoltatore preso come argomento abbia fatto almeno una richiesta. La funzione deve restituire il valore 0 se non è possibile soddisfare le richieste dell'ascoltatore oppure 1 in caso contrario. Le condizioni per soddisfare un utente sono le seguenti: 1) tutte le canzoni richieste devono essere presenti nella playlist della stazione radiofonica; 2) la somma delle durate delle canzoni richieste da un utente non può essere superiore a 15 minuti; 3) La somma (il numero di canzoni) delle canzoni Jazz e Rock non può superare 5

Fondamenti di Informatica T1
Prova Pratica – 31/05/2017
Compito A

(Il candidato si ricordi che 1 minuto equivale a 60 secondi!).

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

Esercizio 3 – Ordinamento Canzoni (radio.h/radio.c)

Si realizzi una procedura:

void ordina (Canzone playlist[], int dimC)

che preso in ingresso un vettore di canzoni e la relativa dimensione logica ordini le canzoni nel vettore in ordine crescente di anno di uscita. Si usi uno degli algoritmi di sorting visti a lezione.

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

Esercizio 4 – Nuova richiesta (radio.h/radio.c)

Si realizzi una funzione:

int inserisciRichiesta (FILE * fp , Canzone c[], int dimC, Richiesta r[], int dimR)

che, ricevuto in ingresso il puntatore ad un file già opportunamente aperto, crei una nuova richiesta (prendendo in input da tastiera le informazioni necessarie) e la scriva nel file indicato dal parametro di ingresso (mantenendone il formato visto in precedenza. La funzione dovrà ritornare un intero che può assumere due valori, 1 nel caso in cui la scrittura sia terminata con successo e 0 in caso contrario (il successo della scrittura dipende dal fatto che la richiesta sia accettata oppure no, come spiegato in seguito). La funzione dovrà leggere come input le informazioni relative alla nuova richiesta, cioè nome dell'ascoltatore, titolo della canzone e la dedica. La funzione provvederà poi a controllare che la richiesta sia "permessa" oppure no. Affinché una richiesta sia permessa devono essere soddisfatte queste condizioni: 1) la canzone deve essere disponibile nella playlist della stazione radio; 2) la somma delle durate delle canzoni richieste non deve essere superiore a 15 minuti; 3) La somma (il numero di canzoni) delle canzoni Jazz e Rock non può superare 5. Per controllare la validità di una nuova richiesta è possibile usare la funzione implementata nell'Esercizio 2.

Nel caso in cui la richiesta sia permessa la funzione provvederà a scriverla sul file passato come argomento - non è necessario aggiornare anche in vettore delle richieste preso in ingresso.

Il candidato realizzi nel main opportuni test per verificare il corretto funzionamento delle funzioni implementate, scrivendo il nuovo prestito nel file di testo "*richieste.txt*" (quello usato anche in fase di lettura). Per controllare il corretto funzionamento si consiglia di effettuare più di un test, ovvero provare ad inserire sia richieste permesse che richieste

Fondamenti di Informatica T1
Prova Pratica – 31/05/2017
Compito A

non permesse.

Esercizio 5 – Main (main.c)

Il candidato realizzi un programma in grado di:

- 1) Leggere le canzoni presenti nella playlist della stazione radio e le richieste fatte dagli ascoltatori dai rispettivi file creando due array.
- 2) Calcolare e mostrare a video le richieste soddisfacibili e quelle non soddisfacibili (non è un problema se per uno stesso ascoltatore vengono mostrate molteplici richieste)
- 3) Stampare le canzoni disponibili e ordinate in base all'anno di uscita (ordine crescente).
- 4) Inserire nel file delle richieste una nuova richiesta.