

Reti di Calcolatori T

Reti di Calcolatori L-A

Appello del 15/02/2011

Compito 2

Cognome:
Nome:
Matricola:

Tempo a disposizione: 2h

È obbligatorio inserire **Cognome Nome Matricola e Numero Compito** all'inizio di ogni file sorgente, pena la non valutazione del compito, che viene stampato in modo automatico solo in caso siano presenti gli elementi detti sopra.

Si devono consegnare **tutti i file sorgente e tutti gli eseguibili prodotti singolarmente** (per favore, solo quelli relativi ai file sorgente consegnati!!!).

La prova intende valutare le capacità progettuali e di programmazione sia in **ambiente Java** che in **ambiente C**, pertanto è consigliato sviluppare **entrambe** le soluzioni richieste al meglio.
In entrambi gli esercizi, sia in Java che in C, si effettuino gli opportuni controlli sui parametri della richiesta e si gestiscano le eccezioni, tenendo presente i criteri secondo cui si possa ripristinare il funzionamento del programma oppure si debba forzarne la terminazione.
Leggete con attenzione le specifiche del problema prima di impegnarvi "a testa bassa" nello sviluppo delle singole parti. Naturalmente, ci aspettiamo che i componenti da consegnare siano stati provati e siano funzionanti.

Si richiede il progetto dei servizi **BuonaMusica**, per una gestione dell'inventario e delle recensioni dei clienti relative a CD musicali. I servizi di BuonaMusica utilizzano due strutture dati. La prima struttura dati, **Inventario**, mantiene per ogni CD: un **codice** inteso come **identificatore** unico del CD, il **nome dell'autore**, il **costo** inteso come costo unitario e la **quantità di prodotto rimasta** (fino a un minimo di 0 unità). La seconda struttura dati, **Recensioni**, invece, riporta per ogni recensione le seguenti informazioni: il **codice** del CD, chiave d'accesso alla struttura dati Inventario, l'**opinione** intesa come una stringa e il **voto**, un intero da 1 a 10.

Si vogliono realizzare le funzionalità di gestione:

1. **inserimento di un nuovo CD**: questa operazione richiede il codice e il nome dell'autore, il costo unitario e la quantità, e aggiorna la sola struttura dati Inventario effettuando gli opportuni controlli;
2. **visualizzazione della media dei voti ottenuti da un CD**: questa operazione richiede il codice del CD e restituisce la media dei voti delle recensioni per tale CD;
3. **acquisto di un certo numero di CD**: questa operazione richiede il codice del CD e la quantità di prodotto da acquistare, quindi restituisce se possibile il costo totale dovuto (per un numero di CD pari al minimo fra quelli richiesti e quelli rimasti) aggiornando opportunamente la struttura dati;
4. **visualizzazione delle opinioni relative ad un CD ed il cui voto sia inferiore a una certa soglia**: questa operazione richiede il codice del CD e restituisce l'elenco di tutte le opinioni relative a tale CD con voto inferiore alla soglia data.

Si progettino con particolare attenzione **le due strutture dati (Inventario e Recensioni)** che mantengono lo stato, fino ad un massimo di N elementi ciascuna. Le strutture dati sono da implementare e inizializzare opportunamente nei diversi ambienti richiesti, Java e C.

Si considerino e si segnalino le possibilità di interferenze fra le operazioni, evitandole dove necessario.

Parte Java

Utilizzando **java RMI** sviluppare un'applicazione C/S che consenta di effettuare le operazioni remote per:

- inserire un nuovo CD;
- visualizzare la media dei voti ottenuti da un CD.

Il progetto RMI si basa su un'interfaccia (contenuta nel file *RMI_interfaceFile.java*) in cui vengono definiti i metodi invocabili in remoto dal client:

- Il metodo **inserimento_CD** accetta come parametri d'ingresso il codice del CD, il nome dell'autore, il costo unitario e la quantità; quindi restituisce l'esito dell'operazione, 0 se l'inserimento è andato a buon fine, -1 altrimenti, ad esempio, se il codice è già presente, o se la struttura dati Inventario è piena.
- Il metodo **visualizzazione_media** accetta come parametro d'ingresso una stringa col codice del CD; quindi il server percorre la struttura dati Recensioni valutando la media dei voti ottenuti dal CD e restituisce l'esito dell'operazione, un float positivo pari alla media dei voti ottenuti se la valutazione è andata a buon fine, -1 altrimenti, ad esempio, se il CD non è fra quelli presenti nella struttura Inventario.

Si progettino inoltre i programmi:

- **RMI_Server** (contenuta nel file *RMI_Server.java*), che implementa i metodi del server invocabili in remoto;
- **RMI_Client** (contenuta nel file *RMI_Client.java*), che realizza l'interazione con l'utente proponendo ciclicamente i servizi che utilizzano i due metodi remoti, e stampa a video i risultati, fino alla fine del file di input da tastiera.

Parte C

Progettare un **servitore multiservizio (usando la select)** che consenta di effettuare le operazioni remote per:

- acquistare un certo numero di CD;
- visualizzare le opinioni relative ad un CD il cui voto sia inferiore a una certa soglia.

Più in dettaglio:

- Il **client_datagram** è organizzato come un processo ciclico fino alla fine del file di input e realizza la funzionalità di acquisto di un certo numero di CD utilizzando **socket datagram**. Per ogni richiesta, il client richiede all'utente e invia al server il codice del CD e la quantità di prodotto da acquistare, quindi riceve l'esito della operazione e lo stampa a video.
- Il **client_stream** è organizzato come un processo ciclico fino alla fine del file di input e realizza la funzionalità di visualizzazione delle opinioni relative a un CD il cui voto sia inferiore a una certa soglia utilizzando **socket stream** e un'unica socket per cliente. Per ogni richiesta, il client richiede all'utente e invia al server il codice del CD e la soglia; quindi riceve l'elenco delle opinioni e lo stampa a video.
- Il **server** principale unico discrimina le richieste utilizzando la primitiva select. Il server **gestisce in modo seriale** la funzionalità di acquisto di un certo numero di CD, mentre la funzionalità di visualizzazione delle opinioni relative a un CD il cui voto sia inferiore a una certa soglia **deve essere realizzata in modo parallelo**. Per ogni richiesta di **acquisto di un certo numero di CD**, il server riceve il codice del CD e la quantità di prodotto da acquistare; quindi, se il CD esiste e il numero lo consente, decrementa la quantità di prodotto rimasta (per un numero di CD pari al minimo fra quelli richiesti e quelli rimasti) e spedisce al client il costo totale dei CD acquistati (sottratti), altrimenti spedisce al client un messaggio di errore. Per ogni richiesta di **visualizzazione delle opinioni relative a un CD ed il cui voto sia inferiore a una certa soglia**, il figlio riceve il codice del CD e la soglia e restituisce l'elenco di tutte le opinioni relative a recensione con un voto inferiore alla soglia data.