

Linguaggi

*Corso M-Z - Laurea in Ingegneria Informatica
A.A. 2008-2009*

Alessandro Longheu

<http://www.dit.unict.it/users/alongheu>

alessandro.longheu@dit.unict.it

Homeworks

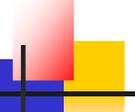
1

A. Longheu – Linguaggi M-Z – Ing. Inf. 2008-2009

Regole

- Gli homework che seguono sono tre, tutti facoltativi ed indipendenti (se ne possono effettuare 1, 2 o tutti e 3)
- Gli homework sono svolti da un gruppo di 2 persone, costante per i 3 testi
- La scadenza della consegna e' il 20 gennaio 2009; la consegna viene effettuata da uno dei due componenti del gruppo sul sito <https://giove.dit.unict.it/linguaggi>, indicando come nome (e formato) del file <cognome1>_<cognome2>_<#homework>.rar
- Ogni homework deve avere una documentazione esplicativa allegata
- Gli homework consentono di ottenere da 0 a 3 punti sulla valutazione finale, e sono validi per gli appelli fino al mese di luglio (sessioni invernale ed estiva dell'A.A. in cui sono svolti)

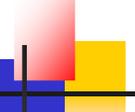
2



Homework 1

- Occorre realizzare un sistema di prenotazione per un'agenzia di viaggi.
- Il cliente, che può essere standard o business (diversificati in base alla scontistica effettuabile, creare le opportune classi), deve potere gestire le prenotazioni dei propri viaggi tramite una data agenzia
 - il "viaggio" è costituito da una sola tratta, effettuata in aereo
 - Per "gestione" si intende inserimento, modifica, cancellazione e ricerca di prenotazioni; per l'inserimento, il sistema deve verificare che sia possibile (posti disponibili sul volo, volo esistente ecc.) e fornire un messaggio corrispondente, altrettanto per la modifica (della partenza, destinazione, giorno od ora del volo). La cancellazione viene effettuata utilizzando giorno ed ora come chiave, supponendo che lo stesso cliente non abbia voli distinti prenotati con quella chiave. La ricerca deve avvenire tramite qualsiasi campo
 - Il cliente ha le sue prenotazioni, memorizzate in una opportuna struttura dati
- L'agenzia detiene gli archivi di tutte le prenotazioni di tutti i clienti ed anche una struttura dati (condivisa) per tutti i voli disponibili, con le relative informazioni (partenza, destinazione, giorno ed ora, posti residui disponibili), utilizzata per le prenotazioni; ad ogni prenotazione da soddisfare o annullare, il numero di posti residui su quel volo decrementa o incrementa
- ...

3

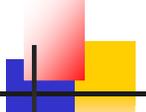


Homework 1

Prevedere inoltre tre delle seguenti funzionalità per l'agenzia:

0. Annullamento di un volo, che di conseguenza congela tutte le prenotazioni ad esso associate (modellare con opportuni campi, in modo che il cliente possa accorgersi dello status delle prenotazioni)
1. Ripristino di un volo annullato
2. Creare due file con i clienti presenti nel sistema, divisi per categoria
3. Determinare e stampare il volo corrente con più posti disponibili
4. Determinare e stampare le informazioni sui voli pieni
5. Determinare e stampare il cliente business con più prenotazioni
6. Determinare e stampare il cliente standard con meno prenotazioni
7. Determinare e stampare la località di partenza con il maggior numero di voli
8. Determinare e stampare la località di arrivo con il minor numero di voli
9. Determinare e stampare il costo totale delle prenotazioni di un dato cliente
10. Applicare uno sconto dato ai clienti, riducendo le tariffe per ogni singola prenotazione e applicando uno sconto del 20% superiore di quello dato per i clienti business
11. Permettere il trasferimento di tutti i dati in un db sottostante
12. Riallocare una prenotazione di un cliente (verificare cioè se esiste un volo sulla stessa tratta con altra compagnia per una data successiva ad una fornita in ingresso)
13. Creare un file di testo con tutte le prenotazioni di un dato cliente
14. Creare un file di testo con tutti i voli di una data compagnia

4



Homework 1

- La scelta delle tre funzioni deve essere effettuata come segue:
 - Siano N1 ed N2 il numero di lettere del cognome dei componenti del gruppo
 - Il numero della prima funzione è dato da $N1\%15$, per la 2 da $N2\%15$, la terza infine da $(N1+N2)\%15$. Se $N1=N2\%15$ (p.es. $N1=3$, $N2=3$ o $N2=18$), porre $N2=(N2+4)\%15$
 - Le funzioni devono essere tutte distinte, in caso contrario, contattare il docente per una riassegnazione
- Realizzare un'applicazione che soddisfi tutti i requisiti indicati, e permetta a più clienti di connettersi e ricevere una risposta alle richieste effettuate.
- L'agenzia, all'uscita del programma deve serializzare tutte le strutture dati su file, e all'avvio deve caricare tutte le informazioni.
- Rilasciare una documentazione in javadoc completa con tutte le spiegazioni e scelte progettuali effettuate (architettura del sistema, scelta ruoli client e server, flusso delle informazioni, strutture dati utilizzate, ipotesi aggiuntive...)

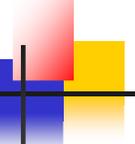
5



Homework 2

- Realizzare un programma che legga da un file di testo i dati relativi ai clienti di una libreria e ne effettui il caricamento in memoria centrale in un'apposita struttura dati. In particolare:
 - nel file viene indicato l'ID del cliente (una riga con prefisso "#"), il nome e cognome (entrambi su una seconda riga), il numero dei titoli che seguono (su una sola riga) e la lista di titoli acquistati (titoli ognuno su una riga)
 - La struttura dati in memoria centrale deve essere una tabella hash di clienti, ognuno con la lista concatenata dei propri titoli. Scrivere la classe che modella la struttura così definita e che fornisca dei metodi per avere:
 - Il nome e cognome di un cliente dato l'ID
 - La lista dei nomi e cognomi di tutti i clienti con ID entro un range assegnato
 - La lista (senza ripetizioni) dei titoli acquistati da un gruppo di utenti aventi l'ID entro un range specificato
 - La stampa dell'intera struttura dati
 - ...

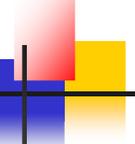
6



Homework 2

- ...gestire inoltre tramite le eccezioni (esistenti o personalizzate) le seguenti condizioni:
 - In caso di file non trovato, si deve usare un nome di default e riprovare; se neanche questo file di default e' presente, uscire con un apposito messaggio
 - In caso di mancata lettura della riga in ingresso dal file, spostarsi al cliente successivo, segnalare che alcuni dati del cliente corrente non possono essere letti e proseguire
 - qualora nel file dei clienti si riscontrasse un id duplicato, consentire il cambio dell'id inserendolo da tastiera e proseguire con il caricamento in memoria
 - Qualora il numero di titoli indicato nel file fosse discorde con il numero reale di titoli letti per quel cliente, prendere quest'ultimo come valore e segnalare l'errore del file a video
 - Qualora uno stesso titolo fosse presente due o piu' volte per uno stesso cliente, segnalarlo a video e scartare i duplicati nell'inserimento in memoria
- Documentare (anche in formato solo testo) tutto il lavoro svolto

7



Homework 3

- Realizzare e documentare un programma per la gestione di un insieme di richieste provenienti dalla rete e dirette verso 4 possibili servizi implementati sullo stesso server.
 - Ogni richiesta è modellata come una terna di stringhe <#id_client, #id_servizio_richiesto, richiesta> e viene inviata con un socket datagram da un numero generico di client
 - Ogni servizio contiene una coda di richieste con dimensione fissa (LinkedListDeque) in cui sono inserite le richieste dei vari client
 - Periodicamente, un thread seleziona un elemento da una della quattro code e invia una risposta al richiedente tramite un datagram
 - Gestire tramite le opportune eccezioni (NoSuchElementException, IllegalStateException) le situazioni di coda piena e vuota, in particolare visualizzando il messaggio "nessun richiedente per il servizio x" in caso di coda vuota, o inviando immediatamente un datagram al client richiedente in caso di coda piena indicando "server occupato"
 - L'applicazione deve avere una GUI per visualizzare le code sullo schermo (status servizi), e i client che hanno fatto richiesta (storico)

8