

Prova d'esame del 24/01/2019 – Turno A

Si consideri il database “extflightdelays”, contenente informazioni su aeroporti, compagnie aeree, orari di partenza e di arrivo dei voli effettuati negli Stati Uniti durante il 2015. Il database (disponibile su Kaggle all'indirizzo: <https://www.kaggle.com/usdot/flight-delays/data>) è strutturato secondo il diagramma ER della pagina seguente.

Si intende costruire un'applicazione JavaFX che permetta di interrogare tale base dati. L'applicazione dovrà svolgere le seguenti funzioni:

PUNTO 1

- Facendo click sul bottone *Crea Grafo*, creare un grafo semplice, pesato e orientato, i cui vertici siano tutti gli stati USA presenti nella tabella ‘airports’ (colonna STATE). Gli archi del grafo devono rappresentare le coppie di STATE tra i cui aeroporti esiste almeno un volo.
- Il peso di un arco orientato va calcolato come **il numero di velivoli distinti** che, partendo da un qualsiasi aeroporto del primo stato, atterrano in un qualsiasi aeroporto del secondo stato. Il numero di velivoli distinti si può ottenere cercando i valori distinti tra loro della colonna TAIL_NUMBER nella tabella ‘flights’. L'arco è orientato (il numero di velivoli che viaggiano in un senso può essere diverso da quelli che viaggiano in verso opposto), ed ogni nodo può avere un arco di tipo “cappio” (che rappresenta i velivoli che viaggiano tra due aeroporti dello stesso stato).
- Popolare il menù a tendina con la lista degli stati (i vertici del grafo) in ordine alfabetico. Al click del bottone *Visualizza Velivoli*, stampare per lo stato selezionato l'elenco di tutti gli stati ad esso direttamente collegati da un arco uscente, ed il peso dell'arco che li collega, ordinati per numero di velivoli (cioè peso) decrescente.

The screenshot shows a JavaFX application window titled "Esame 24-01-2019". The interface includes a "Crea Grafo" button, a dropdown menu, a "Visualizza Velivoli" button, input fields for "T" and "G", and a "Simula" button. Below these controls is a large empty rectangular area, likely for displaying the results of the simulation or graph creation.

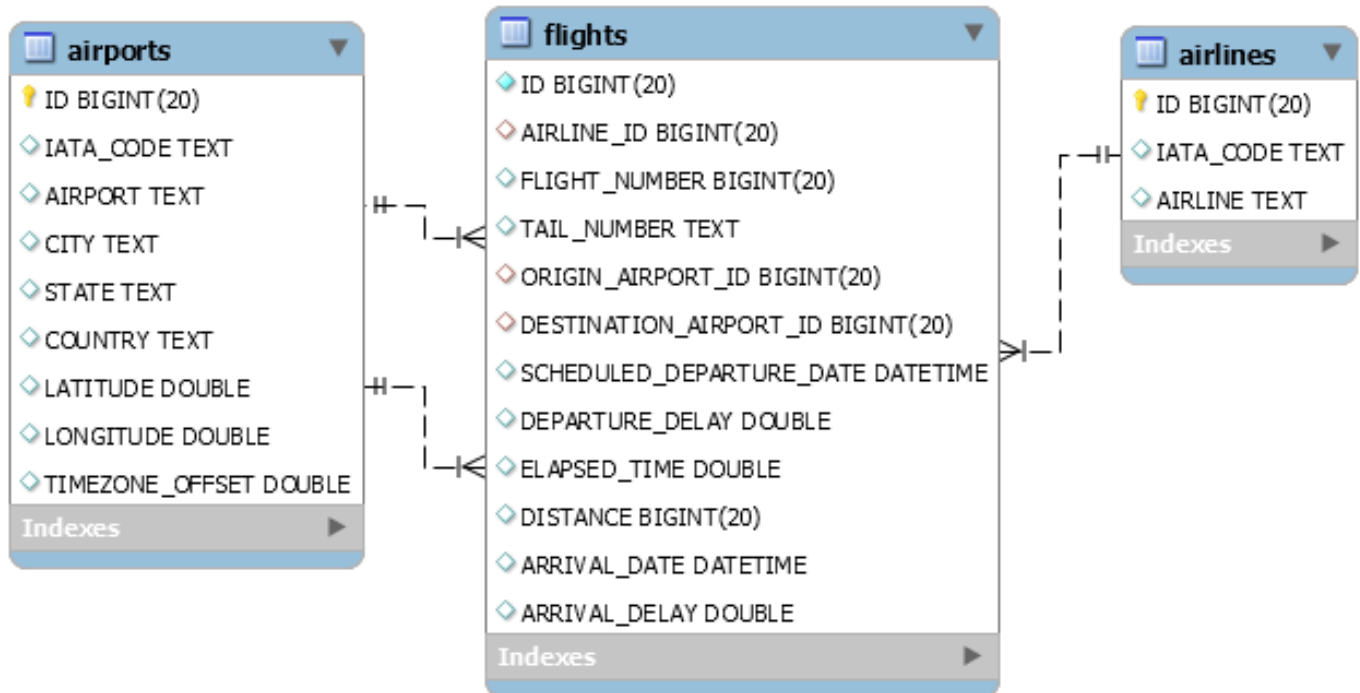
PUNTO 2

- Effettuare una simulazione degli spostamenti di un numero T di turisti che visitano gli USA spostandosi in modo casuale tra i diversi stati.
- Tutti i T turisti al tempo iniziale si trovano tutti nello stato selezionato dall'utente al punto precedente.
- Si assume che ogni turista prenda un volo al giorno. Ogni giorno, il turista (che si trova in un certo stato) prenderà un volo uscente, scegliendolo casualmente tra tutti i voli uscenti dallo stato. La probabilità di scegliere un certo volo dipende dal peso dell'arco uscente. In altre parole, se mi trovo nello stato CA, avrò una probabilità di volare verso NY data dal peso dell'arco CA->NY divisa per la somma dei pesi degli archi uscenti da CA). Tra i possibili voli, è possibile anche considerare voli che rimangano nello stesso stato (arco CA->CA).
- La simulazione verrà condotta simulando G giorni.
- I parametri T e G sono inseriti dall'utente.

- f. Al termine della simulazione stampare, per ciascuno stato, il numero i turisti in esso presenti.

Nella realizzazione del codice, si lavori a partire dalle classi (Bean e DAO, FXML) e dal database contenuti nel progetto di base. È ovviamente permesso aggiungere o modificare classi e metodi.

Tutti i possibili errori di immissione, validazione dati, accesso al database, ed algoritmici devono essere gestiti, non sono ammesse eccezioni generate dal programma.



La tabella *airports* riporta tutti gli aeroporti presenti sul territorio statunitense, la tabella *airlines* tutte le compagnie aeree operanti, mentre la tabella *flights* contiene informazioni sui voli effettuati da una specifica compagnia aerea tra una coppia di aeroporti, riportando le date di partenza, di arrivo ed eventuali ritardi.