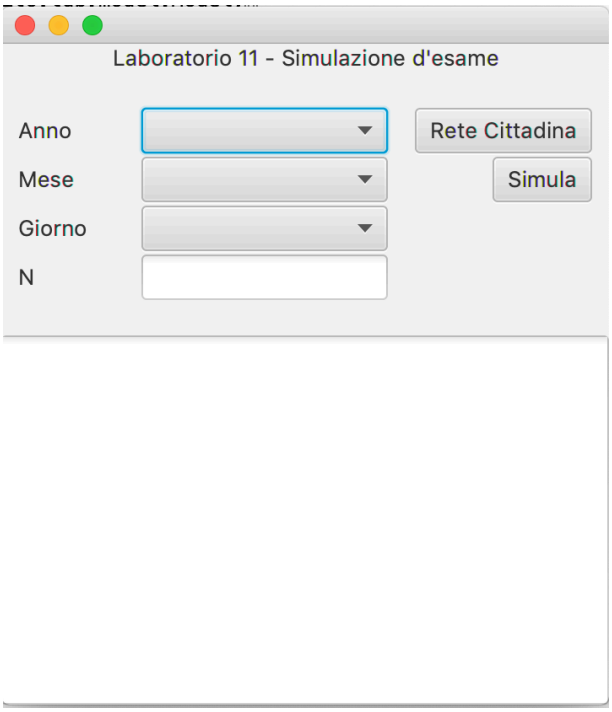
Simulazione d'esame 05/06/2019

Si consideri il data-set "denver_crime" che riporta una serie di oltre 300k eventi criminosi tra il 2014 ed il 2017 nella città di Denver (Colorado, USA). Il data-set contiene un'unica tabella 'events' (vedi la figura nella pagina successiva), ed è tratto dell'originale db costantemente aggiornato e disponibile al sito web https://www.denvergov.org/opendata/dataset/city-and-county-of-denver-crime.

Si intende costruire un'applicazione JavaFX che permetta di interrogare tale base dati e ricavare informazioni utili alla distribuzione delle forze di polizia. L'applicazione dovrà svolgere le seguenti funzioni:

PUNTO 1



a. Si permetta all'utente di selezionare dal menu a tendina un anno tra quelli presenti nel database. Si usi come riferimento la colonna 'reported_date'.
b. Alla pressione del bottone 'rete cittadina' si costruisca un grafo pesato non orientato che connette 7 vertici, ciascuno corrispondente ad uno dei distretti di Denver (campo 'district_id'). Per ogni distretto si considerino tutti gli eventi dell'anno selezionato, e, usando le coordinate 'geo_lon' e 'geo_lat' si determini il centro geografico dei crimini di ogni distretto. Ogni coppia di stretti è collegata da un arco il cui peso sia parti alla distanza tra i centri dei due distretti connessi, espressa in km. Per calcolare tale distanza si faccia uso della libreria *simplelatlng*[1], già inclusa nel progetto.
c. Si visualizzi, per ciascuno dei 7 distretti, l'elenco dei distretti adiacenti, ordinata in base alla distanza crescente.

PUNTO 2

a. Si permetta all'utente di selezionare una specifica data (giorno e mese) dell'anno già considerato ed inserire un numero N (compreso tra 1 e 10) di agenti disponibili. Vogliamo valutare il numero N di agenti utili a gestire gli eventi criminosi di Denver tramite una simulazione dei loro spostamenti durante la giornata selezionata.
b. Si ipotizzi che la centrale di polizia di Denver si trovi nel distretto a minore criminalità dell'anno in corso e che tutti gli agenti siano originariamente in centrale al momento del primo evento del giorno. Al verificarsi di un evento (ordinati per 'reported_date'), la centrale sceglie l'agente da inviare sul posto, selezionando l'agente libero più vicino. Ciascun agente libero può muoversi ad una velocità di 60 km/h, tenendo conto della distanza tra i distretti calcolata nel punto 1. Il tempo necessario a gestire un evento (tempo di permanenza nel distretto con l'evento) è, con uguale probabilità, di 1 o 2 ore per eventi di tipo 'offense_category_id=all_other_crimes', mentre il tempo necessario a gestire qualsiasi altro tipo di 'offense_category_id' è di 2 ore. Una volta che l'evento è stato gestito l'agente rimane sul posto e risulta disponibile ad occuparsi di un nuovo evento. Un evento si considera "mal gestito" se l'agente arriva nel distretto dell'evento oltre 15 minuti dal suo orario in 'reported_date'.

[1] http://javadox.com/com.javadocmd/simplelatlng/1.3.0/com/javadocmd/simplelatlng/package-summary.html

c. Il simulatore dovrà produrre il numero degli eventi “mal gestiti” in funzione della data specificata e del numero di agenti N.

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.