

## 03FYZ TECNICHE DI PROGRAMMAZIONE

Esercitazione di laboratorio 09 – 13 maggio 2020

Obiettivi dell'esercitazione:

- Utilizzo della libreria JGraphT
- Visite e ricerca delle componenti connesse

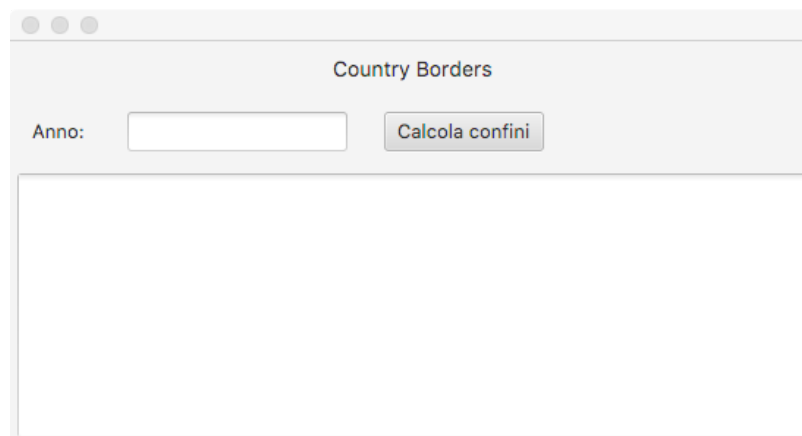
### ESERCIZIO 1

Dopo aver effettuato il fork del progetto relativo al nono laboratorio (Lab09), scrivere un'applicazione JavaFX che permetta di interrogare il database *countries.sql* e restituire informazioni sui confini via terra tra stati. L'interfaccia grafica, *Fig. 1*, viene fornita nel progetto di base. La struttura del database viene analizzata nella pagina seguente.

L'applicazione dovrà svolgere le seguenti funzioni:

- Permettere all'utente di inserire un anno, nell'intervallo 1816 – 2016, nell'apposita casella di testo e di richiedere il calcolo dei confini facendo click sul bottone *Calcola confini*
- Creare un grafo che rappresenti la situazione dei confini mondiali nell'anno specificato dall'utente. Le relazioni di confine da considerare sono tutte quelle in cui campo 'year' sia minore o uguale dell'anno specificato. I vertici del grafo rappresentano tutte le nazioni tra le quali esiste un confine nell'arco di tempo specificato, e gli archi (non orientati e non pesati) rappresentano un confine via terra ('conttype'=1)
- Stampare l'elenco degli stati, indicando per ciascuno il numero di stati confinanti (grado del vertice)
- Stampare il numero di componenti connesse nel grafo.

**Nota:** per trovare il numero di componenti connesse di un grafo si può sfruttare la classe di JGraphT *ConnectivityInspector*. <http://jgrapht.org/javadoc/org/jgrapht/alg/ConnectivityInspector.html>



*Fig. 1*

**Continua nella pagina seguente**

## ESERCIZIO 2

Partendo dal programma sviluppato nell'esercizio 1, aggiungere la funzionalità "Trova tutti i vicini" a partire da uno stato selezionato. Modificare l'interfaccia grafica, inserendo una nuova *ComboBox*, per selezionare uno stato, ed un *Button*, per la ricerca degli stati raggiungibili a partire da quello selezionato.

Facendo click sul bottone "Trova tutti i vicini" viene visualizzata la lista di tutti i nodi raggiungibili nel grafo a partire da un vertice selezionato, che coincide con la componente connessa del grafo relativa allo stato scelto.

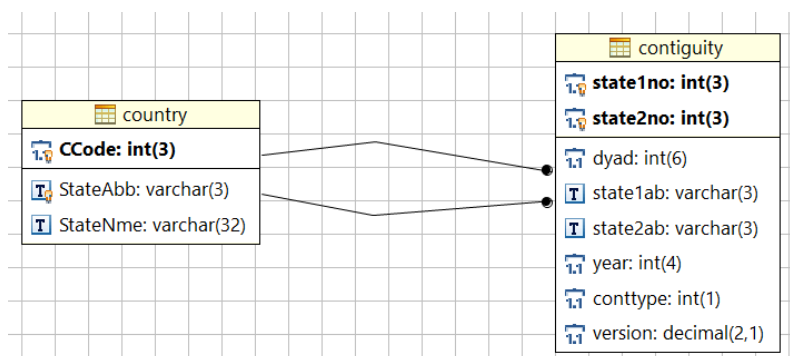
**Nota:** per trovare la lista di tutti i vicini di un vertice (quindi la componente connessa relativa al vertice selezionato) è necessario effettuare una visita del grafo utilizzando una delle seguenti tecniche:

- Sfruttando i metodi esposti dalla libreria JGraphT (*BreadthFirstIterator*, *DeepFirstIterator*)
- Implementando manualmente un algoritmo ricorsivo (per la visita in profondità)
- Implementando manualmente un algoritmo iterativo

La versione iterativa utilizza due liste, quella dei nodi *Visitati* e quella dei nodi *daVisitare*. Si inizia inserendo lo stato scelto nella lista *daVisitare*. L'algoritmo continua fino a quando la lista dei nodi *daVisitare* non si svuota. Ad ogni passo si estrae un nodo dalla lista *daVisitare* e si inseriscono tutti i nodi vicini a quello estratto (a meno di quelli già visitati) nella lista *daVisitare*. Infine, il nodo estratto viene inserito nella lista dei *Visitati*.

Si consiglia di provare almeno due metodi alternativi.

### Descrizione del database "country borders":



La tabella *country* riporta la lista di tutte le nazioni, ciascuna identificata da un numero, da un'abbreviazione univoca di 3 lettere e dal nome completo. La tabella *contiguity* rappresenta la presenza di un confine, per ogni coppia di stati ('state1no', 'state2no'), a partire dall'anno 'year' (anni compresi tra il 1816 ed il 2006). Esistono tipi diversi di confine, ma considerare esclusivamente il confine via terra ('conttype'=1).  
[Nota: la tabella *contiguity2006* non deve essere utilizzata]

I dati sono estratti dal progetto "The Correlates of War" <http://www.correlatesofwar.org/>