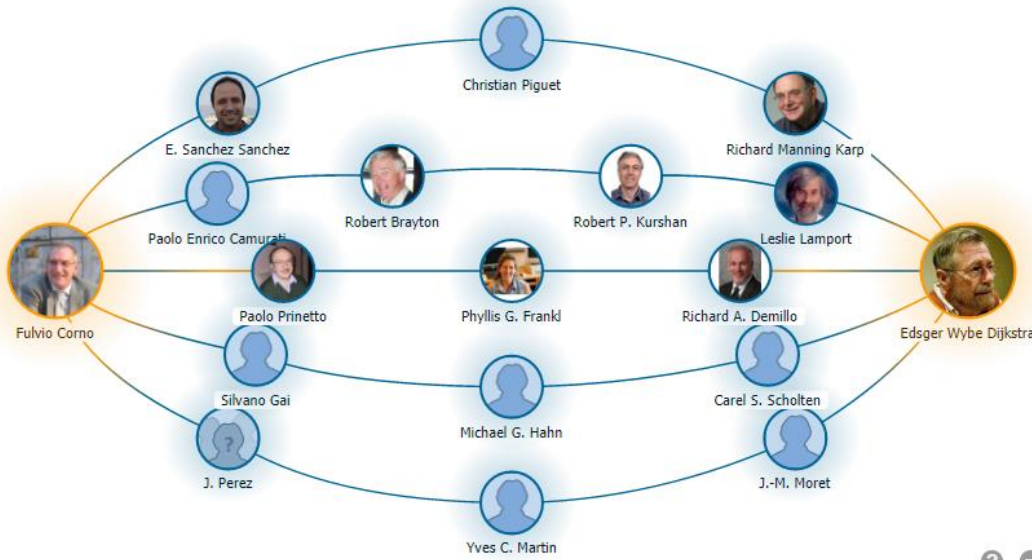


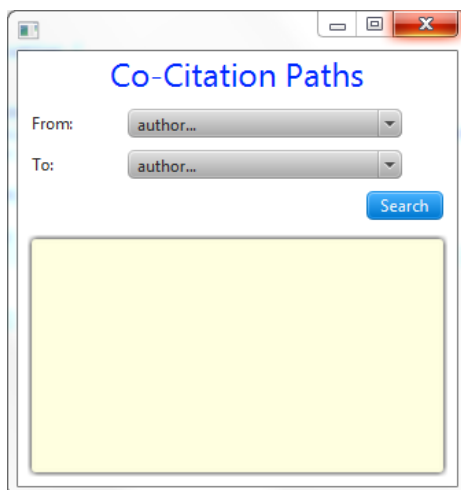
# Simulazione d'esame del 11/06/2013

Nell'ambito delle reti sociali, è particolarmente utile trovare i percorsi che sono in grado di interconnettere due utenti dati, attraverso il minimo numero possibile di intermediari<sup>1</sup>. Lo stesso tipo di analisi si può fare attraverso gli articoli scientifici (oppure gli attori dei film), in cui vi è una connessione ogniqualvolta due persone abbiano scritto un articolo in comune (oppure recitato nello stesso film).

Ad esempio<sup>2</sup>:



Nell'esempio di figura, si vede che il percorso minimo tra F. Corno ed E. Dijkstra è di 4 articoli, in quando Corno ha scritto almeno un articolo insieme a Prinetto, il quale ha scritto almeno un articolo insieme a Frankl, il quale ha scritto almeno un articolo insieme a Demillo, il quale ha scritto almeno un articolo insieme a Dijkstra.



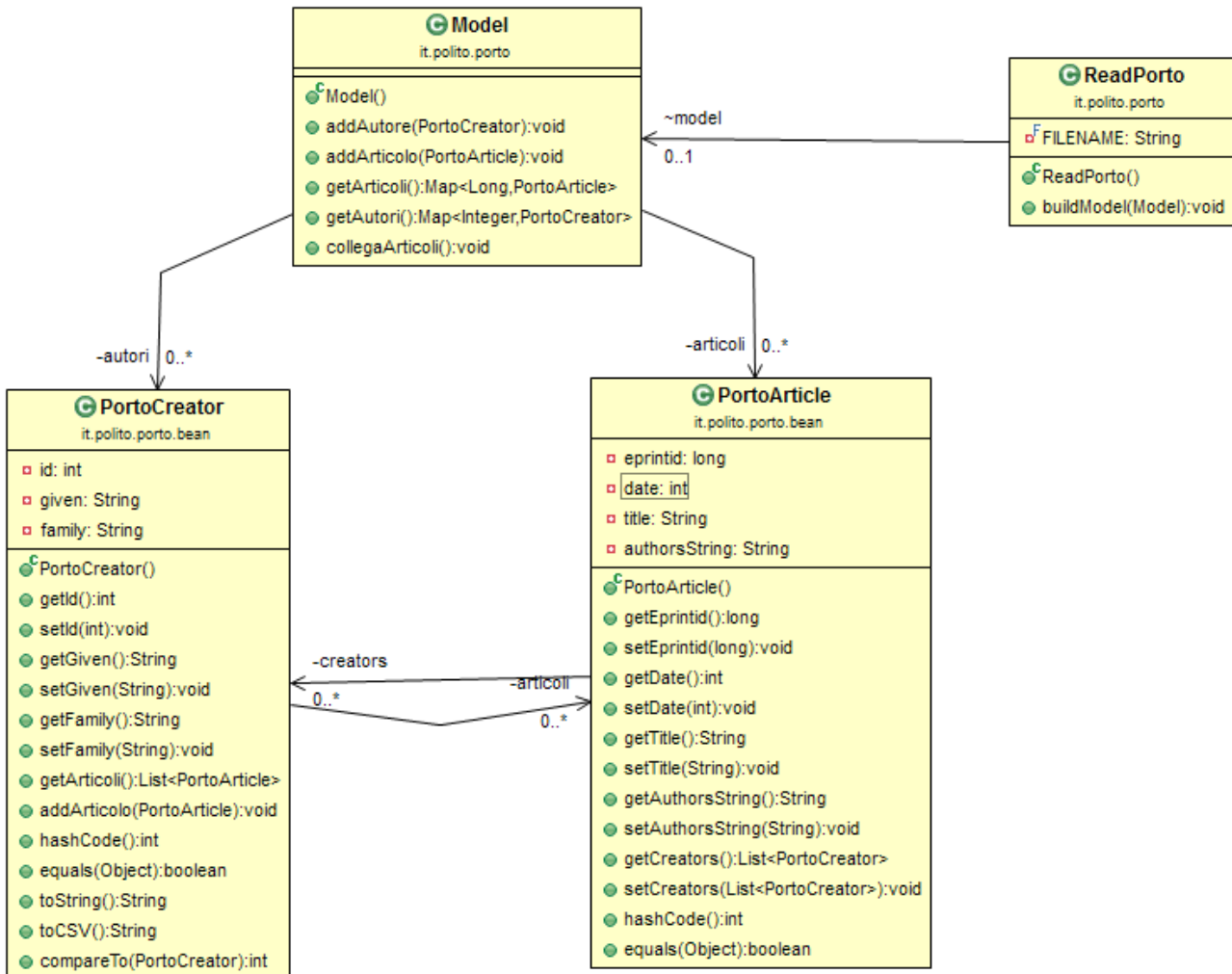
Si vuole realizzare un'applicazione JavaFX la quale, partendo da un elenco di articoli scientifici (e dei rispettivi autori) permetta di trovare il percorso minimo di co-autori in grado di collegare due qualsivoglia autori tra quelli presenti nel database.

Un esempio di interfaccia utente è rappresentato nella figura a fianco.

<sup>1</sup> A questo proposito è famosa la congettura dei "6 gradi di separazione", che si riferirebbe al numero di intermediari sufficienti a collegare due qualsiasi abitanti del pianeta Terra (attraverso la rete "reale", non quella informatica)

<sup>2</sup> Fonte: <http://academic.research.microsoft.com/VisualExplorer#170272&1997914>

Il database su cui lavorare è ricavato dal repository “Porto” delle pubblicazioni del Politecnico di Torino<sup>3</sup>, il cui contenuto è stato esportato in formato testuale JSON<sup>4</sup>. Nel progetto è già presente un metodo ReadPorto.buildModel il quale legge il contenuto del file JSON e popola gli elenchi presenti nella classe Model. Ogni articolo è rappresentato da un oggetto PortoArticle, mentre ogni autore è rappresentato da un PortoCreator.



Nella realizzazione del codice, si utilizzino le classi Bean e Model già presenti nel progetto scaricabile.

<sup>3</sup> PORTO, Publications Open Repository Torino, <http://porto.polito.it/>

<sup>4</sup> JSON (JavaScript Object Notation), <http://json.org/>