

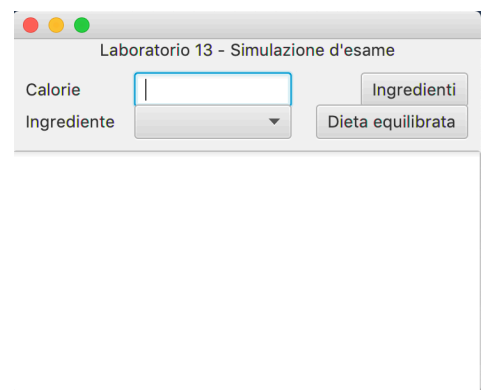
# Simulazione d'esame 12/06/2019

Si consideri il data-set "food\_pyramid" che riporta le qualità nutrizionali di una serie di cibi ed ingredienti. Il data-set (estratto dal sito <https://catalog.data.gov/dataset/mypyramid-food-raw-data-f9ed6>) contiene una tabella 'food' con vari tipi di cibi, ed una tabella 'condiment' con gli ingredienti inclusi in tali cibi, connessi mediante la relazione 'food\_condiment' (vedi la figura nella pagina successiva), che indica quali ingredienti siano compresi all'interno dei cibi. Si noti che nella tabella food lo stesso food\_code può comparire più volte (in corrispondenza di porzioni di dimensione diversa). Si noti anche che alcuni cibi particolarmente semplici (ad esempio: Whole milk, food\_code 11111000) possono comparire sia come cibo, che come ingrediente.

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

## PUNTO 1

- Si permetta all'utente di inserire un numero reale nel campo 'calorie', per selezionare gli ingredienti che forniscano un numero di calorie inferiore a quello specificato (colonna 'condiment\_calories').
- Alla pressione del bottone 'Ingredienti' si costruisca un grafo pesato non orientato, i cui vertici rappresentino gli ingredienti sopra selezionati, ed i cui archi rappresentino il fatto che due ingredienti siano presenti nella composizione di almeno un cibo. Il peso dell'arco deve corrispondere al numero di cibi diversi (colonna 'food\_code') che contengono entrambi gli ingredienti adiacenti all'arco stesso.
- Si visualizzi, per ciascuno degli ingredienti (ordinati per calorie decrescenti), il numero di calorie corrispondente ed il numero totale di cibi che lo contengono (tenendo conto del peso di tutti gli archi incidenti).



## PUNTO 2

- Si permetta all'utente di selezionare un ingrediente tra quelli selezionati.
- Alla pressione del bottone "Dieta equilibrata", si calcoli un insieme di ingredienti *indipendenti tra loro* che apportino il massimo numero di calorie, e che contengano il vertice selezionato. Gli ingredienti dell'insieme devono essere a due a due *indipendenti*, cioè non devono mai comparire nello stesso cibo (cioè non devono essere adiacenti). Nota: nel calcolo dell'insieme di vertici indipendenti, si lavori partendo da grafi di piccole dimensioni.
- Si visualizzi l'insieme di ingredienti così ottenuto.

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.**

