

Comuni

Obiettivi

Il file "comuni_prov_mi.csv" contiene le seguenti informazioni separate da `;`:

- nome
- indirizzo
- cap
- superficie
- asl
- codice_istat
- distanza_da_milano

dei 135 comuni che fanno parte della provincia di Milano (una riga per comune). Si vuole scrivere un programma che legga questi dati e possa fare delle interrogazioni sui campi (filtraggi). Ad esempio si potrebbe volere stampare la lista di tutti i comuni che fanno parte dell' "ASL MI1" oppure quelli che hanno una superficie compresa tra due valori.

Requisiti

1. Definire una entità '**comune**' che deve contenere tutte le informazioni sopra elencate. Potete usare `std::string` per le stringhe e `int` per i dati numerici.
 - a. Implementate i metodi fondamentali nell'entità '**comune**'
 - b. Implementare, tramite `operator<<`, la stampa del contenuto di un '**comune**'.
2. Definire una entità '**comuni**' che deve contenere un insieme di oggetti '**comune**'. Potete appoggiarvi alle classi container della STL per la memorizzazione.
 - a. Implementate i metodi fondamentali nell'entità '**comuni**'
 - b. Implementate i metodi accessori per poter aggiungere e accedere le singole entità '**comune**'
 - c. Implementare, tramite `operator<<`, la stampa del contenuto di un oggetto '**comuni**'.
3. Definire nel `main.cpp` delle funzioni di lettura del file `comuni_prov_mi.csv` per riempire con le informazioni lette un oggetto '**comuni**'.
 - a. Separate in due distinte operazioni la lettura: una funzione che legge riga per riga il file, e una seconda funzione che, data la riga letta, riempie un oggetto di tipo '**comune**' con le relative informazioni (suggerimento, potrebbero servire le seguenti funzioni: `std::ifstream`, `std::stringstream`, `std::getline` e `atoi`.)
4. Scrivere nel `main.cpp` una funzione generica `find_all_if` che, dato una istanza di '**comuni**' riempita e un predicato, ritorna l'insieme delle entità '**comune**' che soddisfano il predicato. L'insieme è un oggetto di tipo '**comuni**'.
 - a. Scrivete diversi predicati (i.e. funtori) da usarsi con la funzione e testateli.
 - b. Scrivete dei funtori con stato e senza stato.

5. Scrivete altre funzioni che manipolino la lista dei comuni usando anche gli algoritmi della libreria STL in `<algorithm>`.
 - a. Per questo punto è necessario che l'entità 'comuni' esponga degli iteratori (suggerimento: le classi container della STL espongono già gli iteratori)