

**Laboratorio di Ingegneria del Software**  
**AA 2021/2022**  
**docente: Chiara Braghin**

**Assignment 3**  
**10/11/2021**

**Entro quando e dove consegnare:** entro la mezzanotte del giorno 20/11/2021 all'url <https://upload.di.unimi.it>, alla voce *Assignment3* del corso di Ingegneria del SW - Lab.

**Cosa consegnare:** una cartella zippata denominata `cognome_nome_matricola_ass3.zip` (dove ovviamente dovere sostituire a `cognome`, `nome` e `matricola` il vostro cognome, nome e numero di matricola). Nel caso il lavoro sia stato svolto in coppia, il nome del file deve essere `gruppo_cognome1_cognome2_ass3.zip` (dove la parola `gruppo` non viene sostituita). Mi raccomando, ricordatevi di dare sempre un titolo al vostro documento e di specificare il nome del/degli autore/i.

**NB:** La persona/il gruppo che consegna deve essere lo stesso degli assignment precedenti.

**Testo dell'assignment:**

Definire una classe `Elettore` che contenga i seguenti attributi e metodi:

- nome e cognome
- data di nascita (per comodità potete esprimere la data con 3 interi, uno per esprimere il giorno, uno per il mese e uno per l'anno)
- luogo (comune) di nascita
- nazione di nascita
- sesso
- codice fiscale (espresso come array di caratteri)
- voto, un booleano che ha valore `true` nel caso in cui l'elettore abbia espresso il suo voto, `false` nel caso in cui l'elettore non abbia ancora espresso il voto
- metodo `esprimi_voto`, di cui non è necessario dare l'implementazione completa (non mi interessa sapere cosa abbia votato l'elettore, ma che abbia votato. Serve solo per poter specificare alcune condizioni JML)

Esprimere in JML le seguenti proprietà (sta a voi decidere se come invarianti, pre- o post- condizioni):

- il nome e il cognome non possono essere nulli
- un elettore può essere o maschio o femmina
- un elettore può votare solo se maggiorenne
- un elettore può votare al più una volta
- se l'elettore è nato in Italia, allora deve essere indicato anche il comune di nascita
- la data di nascita deve essere valida (ovvero non successiva alla data corrente)
- il codice fiscale deve essere valido (nel vostro caso potete fare un controllo semplificato degli ultimi 5 caratteri alfanumerici del codice fiscale, controllando solo che (i) lettere e numeri compaiano nella giusta posizione e che (ii) per chi è nato all'estero la lettera sia Z).

Le proprietà espresse in JML vanno verificate. Va quindi allegato anche il metodo `main` utilizzato per testare le condizioni espresse in JML.

Il codice fiscale per le persone fisiche è formato da 16 caratteri alfanumerici e la normativa che regola come calcolare il codice fiscale è definita nel decreto del Ministero delle finanze del 23 dicembre 1976.

La struttura del codice fiscale è la seguente (<https://www.codicefiscale.com/>):

1. le prime 3 lettere contengono le consonanti del cognome, prese nel loro ordine. (Se le consonanti non bastano si prendono anche le vocali, sempre nel loro ordine. Nel caso che un cognome abbia meno di tre lettere, la parte di codice viene completata aggiungendo la lettera X.)
2. le 3 lettere successive contengono le consonanti del nome - o dei nomi, se sono più di uno. Nel caso di più nomi separati da virgola, verrà considerato solo il primo nome - prese nel loro ordine. (In caso non ce ne siano a sufficienza, funziona come per il cognome.)
3. 2 numeri che corrispondono alle ultime due cifre dell'anno di nascita.
4. una lettera che indica il mese di nascita (a=gennaio, b=febbraio, c=marzo, d=aprile, ecc.).
5. due numeri indicano il giorno di nascita (se il giorno è compreso tra 1 e 9 occorre metterci uno 0 davanti). Per le donne bisogna aggiungere 40, quindi se il giorno di nascita è l'8, le due cifre saranno 48.
6. una lettera e tre numeri indicano il Comune di nascita, per chi è nato all'estero la sigla inizia per Z ed è seguita dal numero che identifica lo Stato di nascita.
7. un carattere alfabetico di controllo generato attraverso uno speciale algoritmo e che chiude ogni codice fiscale valido

**Hint:** se volete potete utilizzare variabili/metodi di appoggio.