

9 settembre 2019

Tempo a disposizione: 2 ore

Cognome e nome: Matricola: Corso: *à A*
à B

Si vuole implementare in Java un sistema che supporti il Parlamento di una certa nazione. Il sistema gestisce informazioni sui parlamentari (istanze della classe *Parlamentare*), sui disegni di legge (istanze della classe *DisegnoDiLegge*) e sulle votazioni (istanze della classe *Votazione*).

Ogni *Parlamentare* è identificato da un *codice fiscale* ed è caratterizzato dalla regione di appartenenza del collegio in cui è stato eletto e dal suo partito politico, rappresentato da una stringa.

Un *DisegnoDiLegge* già sottoposto a votazione da parte del Parlamento è identificato da un *codice* ed è caratterizzato dal nome, dalla lista dei codici fiscali dei parlamentari proponenti, dalla *data* in cui è stato votato (rappresentabile per semplicità con un intero) e dall'esito della votazione (approvazione o non approvazione), rappresentato con un booleano.

Ciascun oggetto della classe *Votazione* è identificato da un *codice* ed è caratterizzato dal codice fiscale del parlamentare, dal codice del *DisegnoDiLegge* e dal voto espresso (favorevole o contrario), rappresentato con un booleano.

Si implementino in Java le classi *Parlamentare*, *DisegnoDiLegge* e *Votazione*. Si implementi inoltre la classe *Sistema*, scegliendo opportunamente le classi Java più adeguate a gestire tutte le informazioni su parlamentari, disegni di legge e votazioni.

Oltre a scrivere eventuali metodi che si ritengono necessari per implementare l'applicazione, occorre fornire almeno i seguenti metodi nella classe *Sistema*:

1. *public ArrayList<DisegnoDiLegge> m1(String r)*. Il metodo restituisce i disegni di legge approvati soltanto con voti favorevoli espressi da parlamentari eletti in collegi della regione *r*.
2. *public ArrayList<Parlamentare> m2(String p)*. Il metodo restituisce l'elenco dei parlamentari del partito politico *p* che hanno espresso voto contrario su qualche disegno di legge proposto da almeno un parlamentare dello stesso partito *p*.
3. *public ArrayList<Parlamentare> m3(Parlamentare p)*. Il metodo restituisce l'elenco dei parlamentari che hanno espresso lo stesso voto di *p* per ogni disegno di legge.

ESEMPIO:

Si supponga che nel sistema siano stati inseriti i seguenti parlamentari:

- *p1* = {codiceFiscale="RSS", regione="Calabria", partito="partito1"}
- *p2* = {codiceFiscale="BNC", regione="Lombardia", partito="partito2"}
- *p3* = {codiceFiscale="GIA", regione="Lombardia", partito="partito2"}
- *p4* = {codiceFiscale="VRD", regione="Calabria", partito="partito3"}
- *p5* = {codiceFiscale="MAR", regione="Calabria", partito="partito1"}

i seguenti disegni di legge:

- *ddl1* = {codice = 1, nome = "DDL1", listaParlamentari= {"RSS", "BNC", "MAR"}, data = 1, esito = True}
- *ddl2* = {codice = 2, nome = "DDL2", listaParlamentari= {"VRD", "MAR"}, data = 2, esito = False}
- *ddl3* = {codice = 3, nome = "DDL3", listaParlamentari= {"BNC", "GIA", "VRD"}, data = 3, esito = True}

le seguenti votazioni:

- {codice= 1, parlamentare="RSS", disegnoDiLegge=1, Voto=True}
- {codice= 2, parlamentare="BNC", disegnoDiLegge=1, Voto=False}
- {codice= 3, parlamentare="MAR", disegnoDiLegge=1, Voto=True}

- {codice= 4, parlamentare= "VRD", disegnoDiLegge=1, Voto=True}
- {codice= 5, parlamentare="GIA", disegnoDiLegge=1, Voto=False}
- {codice= 6, parlamentare= "RSS", disegnoDiLegge=2, Voto=False}
- {codice= 7, parlamentare= "BNC", disegnoDiLegge=2, Voto=False}
- {codice= 8, parlamentare= "MAR", disegnoDiLegge=2, Voto=True}
- {codice= 9, parlamentare="VRD", disegnoDiLegge=2, Voto=True}
- {codice= 10, parlamentare= "GIA", disegnoDiLegge=2, Voto=False}
- {codice= 11, parlamentare= "RSS", disegnoDiLegge=3, Voto=False}
- {codice= 12, parlamentare= "BNC", disegnoDiLegge=3, Voto=True}
- {codice= 13, parlamentare= "MAR", disegnoDiLegge=3, Voto=False}
- {codice= 14, parlamentare= "VRD", disegnoDiLegge=3, Voto=True}
- {codice= 15, parlamentare="GIA", disegnoDiLegge=3, Voto=True}

Allora:

- Se invocato con il parametro $r = \text{"Calabria"}$, il metodo **m1** restituisce la lista $[ddl1]$, infatti il disegno di legge $ddl1$ è stato votato solo dai parlamentari *RSS*, *MAR* e *VRD*, tutti eletti in collegi della regione *Calabria*.
- Se invocato con il parametro $p = \text{"partito1"}$, il metodo **m2** restituisce la lista $[p1]$, infatti il parlamentare $p1$ ha votato contro il disegno di legge $ddl2$ proposto dal collega $p2$, ed entrambi i parlamentari appartengono al partito *partito1*.
- Se invocato con il parametro $d = ddl3$, il metodo **m3** restituisce la lista $[\text{"partito2"}, \text{"partito3"}]$, infatti il disegno di legge $ddl3$ è stato votato dai parlamentari *BNC* e *GIA* del partito *partito2*, e dal parlamentare *VRD* del partito *partito3*.

Esercizio 2

Definiamo picchi della lista quei valori che sono preceduti e seguiti nella lista da valori più piccoli della loro metà. Il primo e l'ultimo elemento della lista non possono essere picchi.

Si arricchisca la classe `ListaConcatenataInt` sviluppata durante il corso con un metodo `picchi()` che restituisce il numero di picchi contenuti nella lista, ad esempio se la lista è

$[4, 9, 12, 36, 16, 23, 87, 34, 18, 64, 33]$ allora la funzione restituisce 2 in quanto 36 e 87 sono picchi (perché $36/2=18$, $18>12$ e $18>16$; $87/2=43,5$, $43,5>23$ e $43,5>34$). Il metodo dovrà essere ricorsivo o invocare un opportuno metodo ricorsivo sulla classe `NodoInt`.