

### Esercizio 1

Si implementi in Java una classe *Sistema* che fornisca metodi per l'analisi dei dati riguardanti messaggi scambiati tra utenti su una piattaforma di messaggistica. Si supponga che le classi *Utente* e *Messaggio* siano già disponibili e forniscano i seguenti metodi:

Classe *Utente*:

- *public String getNome()*, che restituisce il nome che identifica l'utente.
- *public String getCitta()*, che restituisce la città in cui l'utente risiede.
- *public boolean equals(Object o)*
- *public String toString()*

Classe *Messaggio*:

- *public String getNomeMittente()*, che restituisce il nome dell'utente che ha inviato il messaggio.
- *public String getNomeDestinatario()*, che restituisce il nome dell'utente che ha ricevuto il messaggio.
- *public int getData()*, che restituisce la data (rappresentata per semplicità come intero) in cui è stato inviato il messaggio.
- *public boolean letto()*, che restituisce *true* se e solo se il messaggio è stato letto.
- *public boolean equals(Object o)*
- *public String toString()*

La classe *Sistema* contiene le liste delle persone e dei messaggi. Oltre ad eventuali metodi che si ritengano necessari, si includano almeno i seguenti metodi nella classe:

- *public ArrayList<String> nessunaLettura(String destinatario)*. Il metodo restituisce la lista dei nomi degli utenti che hanno inviato qualche messaggio all'utente *destinatario*, ma nessuno di questi messaggi è stato letto.
- *public ArrayList<String> cittaUnica(int dataInizio, int dataFine)*. Il metodo restituisce la lista dei nomi degli utenti per ciascuno dei quali l'insieme dei messaggi inviati nel periodo compreso tra il giorno *dataInizio* e il giorno *dataFine* (estremi inclusi) ha cardinalità maggiore o uguale a 2 e non contiene alcuna coppia di messaggi inviati a destinatari che risiedono in città diverse.
- *public ArrayList<String> mittentiFrequenti(int data, String citta)*. Il metodo restituisce la lista dei nomi degli utenti che hanno inviato il maggior numero di messaggi nel giorno *data* a destinatari che risiedono nella città *citta*.

*Esempio.* Si assuma che i dati a disposizione siano i seguenti:

- Utenti:
  - {"Utente A", "Roma"}
  - {"Utente B", "Roma"}
  - {"Utente C", "Milano"}
  - {"Utente D", "Milano"}
- Messaggi:
  - {"Utente A", "Utente B", 10, false}
  - {"Utente A", "Utente B", 11, false}
  - {"Utente B", "Utente C", 15, true}
  - {"Utente B", "Utente D", 16, false}
  - {"Utente C", "Utente A", 15, true}
  - {"Utente C", "Utente D", 16, false}
  - {"Utente D", "Utente C", 16, true}

Allora:

- *nessunaLettura*("Utente B") restituisce la lista ["Utente A"].
- *cittaUnica*(14, 16) restituisce la lista ["Utente B"].
- *mittentiFrequenti*(15, "Roma") restituisce la lista ["Utente C"].

## **Esercizio 2**

Si arricchisca la classe *ListaConcatenataInt* sviluppata durante il corso con un metodo *stesseSottosequenze* che restituisca *true* se e solo se sono verificate entrambe le seguenti condizioni:

1. la lista non è vuota e la sua lunghezza è pari;
2. se un nodo (ad eccezione degli ultimi due) si trova in posizione  $i$  e contiene il valore  $n$ , allora il nodo che si trova in posizione  $i+2$  contiene lo stesso valore  $n$ .

Ad esempio, la lista [7,2,7,2,7,2] soddisfa entrambe le condizioni di cui sopra. Il metodo *stesseSottosequenze* dovrà essere ricorsivo o invocare un opportuno metodo ricorsivo sulla classe *NodoInt*.