

Esercizio 1

Si implementi in Java una classe *Sistema* che fornisca metodi per l'analisi dei dati riguardanti acquisti fatti presso una catena di negozi. Si supponga che le classi *Negozi* e *Acquisto* siano già disponibili e forniscano i seguenti metodi:

Classe *Negozi*:

- *public String getNome()*, che restituisce il nome che identifica il negozio.
- *public String getCitta()*, che restituisce la città in cui è ubicato il negozio.
- *public boolean equals(Object o)*
- *public String toString()*

Classe *Acquisto*:

- *public String getNomeNegozio()*, che restituisce il nome del negozio in cui è stato effettuato l'acquisto.
- *public int getData()*, che restituisce la data (rappresentata per semplicità come intero) in cui è stato effettuato l'acquisto.
- *public String getCFCliente()*, che restituisce il codice fiscale del cliente che ha effettuato l'acquisto.
- *public String getCittaCliente()*, che restituisce la città di residenza del cliente che ha effettuato l'acquisto.
- *public boolean equals(Object o)*
- *public String toString()*

La classe *Sistema* contiene le liste dei negozi e degli acquisti effettuati. Oltre ad eventuali metodi che si ritengano necessari, si includano almeno i seguenti metodi nella classe:

- *public ArrayList<String> negoziPreferiti(int data)*. Il metodo restituisce la lista dei nomi dei negozi presso i quali, nel giorno *data*, sono stati effettuati il maggior numero di acquisti.
- *public ArrayList<String> clientiEsterniPeriodo(int dataInizio, int dataFine)*. Il metodo restituisce la lista dei codici fiscali dei clienti che, nel periodo compreso tra il giorno *dataInizio* e il giorno *dataFine* (estremi inclusi), hanno effettuato acquisti solo in negozi ubicati in una città diversa da quella in cui risiedono.
- *public ArrayList<String> clientiCittaDiverse()*. Il metodo restituisce la lista dei codici fiscali dei clienti che hanno effettuato acquisti presso almeno 2 città diverse (un acquisto si intende effettuato presso una città se è effettuato presso un negozio di tale città).

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

- Negozi:
 - {"Negozio A", "Roma"}
 - {"Negozio B", "Roma"}
 - {"Negozio C", "Milano"}
- Acquisti:
 - {"Negozio A", 10, "ABCDEF", "Roma"}
 - {"Negozio A", 10, "GHIJKL", "Napoli"}
 - {"Negozio A", 10, "MNOPQR", "Palermo"}
 - {"Negozio B", 10, "MNOPQR", "Palermo"}
 - {"Negozio B", 20, "GHIJKL", "Napoli"}
 - {"Negozio C", 20, "MNOPQR", "Palermo"}
 - {"Negozio C", 20, "ABCDEF", "Roma"}

Allora:

- *negoziPreferiti(10)* restituisce la lista ["Negozio A"].
- *clientiEsterniPeriodo(10, 20)* restituisce la lista ["GHIJKL", "MNOPQR"].

- *clientiCittaDiverse()* restituisce la lista ["ABCDEF", "MNOPQR"].

Esercizio 2

Si arricchisca la classe *ListaConcatenataInt* sviluppata durante il corso con un metodo *alternati* che restituisca *true* se e solo se, scorrendo la lista dall'inizio alla fine, si incontrano valori alternati positivi e negativi. In altri termini, per ogni nodo deve valere che: se esso contiene un valore positivo, allora il successivo (se esiste) contiene un valore negativo, e viceversa. Si assuma inoltre che una lista vuota soddisfi la proprietà verificata da *alternati*. Il metodo *alternati* dovrà essere ricorsivo o invocare un opportuno metodo ricorsivo sulla classe *NodoInt*.