Corso di Laurea in Ingegneria Informatica

Prova scritta di *Fondamenti di informatica e laboratorio* – Modulo 2

30 gennaio 2019

**Tempo a disposizione: 2 ore**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| ***Cognome e nome:*** |  | ***Matricola:*** |  | ***Corso:*** | ***◊ A  ◊ B*** |

In base alle specifiche riportate di seguito, si implementi in Java un sistema di supporto alla gestione delle informazioni relative a pubblicazioni scientifiche. In particolare occorre tenere traccia nel sistema della lista degli autori (istanze della classe *Autore*) e della lista delle loro pubblicazioni (istanze della classe *Pubblicazione*).

Ciascun *Autore* è identificato da un *nome* ed è caratterizzato dalla città di residenza.

Ogni *Pubblicazione* è identificata da un codice ed è caratterizzata dal titolo, dalla lista dei nomi degli autori, da una lista di keywords e da una data (rappresentata per comodità con un intero).

Si implementino in Java le classi *Autore*, *Pubblicazione* e *Sistema*. 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<String>* ***pubblicazioniCittà****(String s).* Il metodo restituisce la lista dei codici delle pubblicazioni redatte solo da autori residenti nella città *s*.
  2. *public ArrayList<Autore>* ***individuali****(int d1, int d2).* Il metodo restituisce la lista degli autori di pubblicazioni a singolo nome nel periodo compreso tra la data *d1* e la data *d2*.
  3. *public ArrayList<Pubblicazione>* ***coautori****(Autore a, Autore b).* Il metodo restituisce la lista delle pubblicazioni scritte congiuntamente dagli autori *a* e *b* (eventualmente insieme ad altri), ordinata secondo la data di pubblicazione dei lavori.

Esempio: Si supponga che gli autori del *Sistema* siano i seguenti:

*a1 = {nome = “Rossi”, città = “Roma”}*

*a2 = {nome = “Bianchi”, città = “Torino”}*

*a3 = {nome = “Verdi”, città = “Cosenza”}*

*a4 = {nome = “Neri”, città = “Cosenza”}*

*a5 = {nome = “Marroni”, città = “Milano”}*

*a6 = {nome = “Gialli”, città = “Milano”}*

che la lista delle pubblicazioni sia la seguente:

*p1 = {codice = “pub1”, titolo = ”Titolo di p1”, autori = [“Rossi”, “Bianchi”, “Neri”], keywords = [“kw1”, “kw2”], data = 1}*

*p2 = {codice = “pub2”, titolo = ”Titolo di p2”, autori = [“Bianchi”, “Neri”], keywords = [“kw3”, “kw4”], data = 2}*

*p3 = {codice = “pub3”, titolo = ”Titolo di p3”, autori = [“Gialli”, “Marroni”], keywords = [“kw1”, “kw3”], data = 2}*

*p4 = {codice = “pub4”, titolo = ”Titolo di p4”, autori = [“Gialli”], keywords = [“kw1”, “kw2”, “kw4”], data = 2}*

*p5 = {codice = “pub5”, titolo = ”Titolo di p5”, autori = [“Neri”, “Verdi”], keywords = [“kw3”, “kw5”], data = 1}*

*p6 = {codice = “pub6”, titolo = ”Titolo di p6”, autori = [“Bianchi”], keywords = [“kw6”], data = 1}*

*p7 = {codice = “pub7”, titolo = ”Titolo di p7”, autori = [“Gialli”, “Rossi”, “Bianchi”, “Neri”], keywords = [“kw1”, “kw6”], data = 2}*

*p8 = {codice = “pub8”, titolo = ”Titolo di p8”, autori = [“Verdi”], keywords = [“kw2”, “kw5”], data = 3}*

Allora:

1. se *città =* **Milano**, il metodo ***pubblicazioniCittà*** restituisce la lista: [“pub3”, “pub4”].
2. se *d1*=**2** e *d2*=**3**,il metodo ***individuali*** restituisce la lista [a3, a6].
3. se *a=***a2**e *b=***a4**, il metodo ***coautori*** restituisce la lista [p1, p2, p7].

**Esercizio 2**

Si arricchisca la classe ListaConcatenataInt sviluppata durante il corso con un metodo verificaLista() che riceve in un’altra ListaConcatenataInt l1 che contiene gli indici di tutti gli elementi nulli nella lista originale e restituisce True se e solo se effettivamente tutti e soli gli elementi puntati da l1 sono nulli, ad esempio se la lista è

[20, 0, 10, 62, 0, 0, 12, 36, 6, 0] e l1=[ 1, 4, 5, 9] allora la funzione restituisce True. Il metodo dovrà essere ricorsivo o invocare un opportuno metodo ricorsivo sulla classe NodoInt.