

Esercizi sugli Array

(Fondamenti di Informatica 1 – Walter Didimo)

Esercizio 1 Un oggetto della classe `InsiemeDiStringhe` consente di rappresentare un qualunque insieme finito di stringhe. La classe `InsiemeDiStringhe` ha il seguente scheletro.

```
class InsiemeDiStringhe {
    /* variabile che memorizza l'insieme delle stringhe rappresentate
       dall'oggetto */
    private String[] insieme;

    /* costruttore: permette di creare un oggetto che rappresenta l'insieme
       delle stringhe contenute nell'array ins */
    public InsiemeDiStringhe (String[] ins){...}

    /* restituisce la stringa di lunghezza massima dell'insieme; se ci sono
       due stringhe di lunghezza massima, ne restituisce una arbitrariamente
       PRE: l'insieme non deve essere vuoto */
    public String stringaPiùLunga () {...}

    /* restituisce la stringa dell'insieme che è lessicografic. più grande
       PRE: l'insieme non deve essere vuoto */
    public String stringaPiùGrande () {...}

    /* visualizza sullo standard output tutte le stringhe dell'insieme che
       hanno il prefisso specificato */
    public void stringheConPrefisso (String prefisso){...}
}
```

Scrivere il corpo dei metodi della classe `InsiemeDiStringhe` e definire inoltre una classe di prova il cui metodo speciale `main` svolga le seguenti funzioni.

- Fa inserire all'utente una sequenza di stringhe a sua scelta.
- Visualizza all'utente la stringa più lunga e quella lessicograficamente più grande della sequenza inserita.
- Fa inserire all'utente una stringa `s` e visualizza all'utente tutte le stringhe della sequenza precedentemente inserita che hanno `s` come prefisso.

Esercizio 2 La classe `Esame` permette di gestire un appello di esame universitario. Precisamente, un oggetto `Esame` rappresenta ha i seguenti campi:

- Un nome (il nome della materia di esame).
- Una data di esame (espressa come stringa).
- Un numero intero che specifica il numero massimo di studenti che possono sostenere l'esame.
- Un numero intero che specifica il numero effettivo di studenti che hanno sostenuto l'esame.
- Una lista di studenti, dove ogni studente ha una stringa che definisce il suo nome e cognome.
- Una lista di voti, associati agli studenti (i voti sono numeri interi).

La classe `Esame` ha i seguenti costruttori e metodi:

- Un costruttore che consente di creare un appello di esame, specificandone il nome, la data ed il numero massimo di studenti.
- Un metodo di istanza che consente di aggiungere nome e cognome di un nuovo studente ed il relativo voto. Se sono già stati immessi più del numero massimo di studenti consentiti, allora il metodo non inserisce lo studente e restituisce false, in caso contrario il metodo inserisce correttamente lo studente e restituisce true.
- Un metodo di istanza che restituisce sotto forma di unica stringa, il nome dell'esame, la data dell'esame, e la lista dei promossi (con relativo voto); uno studente è promosso se ha un voto superiore ai 2/3 della media aritmetica dei voti.

Scrivere il codice della classe `Esame`, e scrivere inoltre il codice di una classe `ProvaEsame` che consente, attraverso il suo metodo `main`, di inserire un appello di esame e tutti gli studenti che hanno sostenuto l'esame, con i relativi voti. Il metodo `main` deve poi visualizzare la lista dei promossi.

Esercizio 3 Un oggetto della classe `AgendaTelefonica` permette di gestire una semplice agenda telefonica. Ogni utenza dell'agenda si compone di un nome, di un cognome e di un numero di telefono. L'agenda può gestire un numero limitato di utenze. Tale numero deve essere scelto all'atto della creazione di un oggetto della classe `AgendaTelefonica`. La classe ha il seguente scheletro.

```
class AgendaTelefonica{

    /* variabili di istanza*/
    private String[] nome;      // nome[i] è il nome dell'i-esima utenza
    private String[] cognome;  // cognome[i] è il cognome dell'i-esima utenza
    private String[] numTel;   // numTel[i] è il num. di telef. dell'i-esima utenza

    /* costruttore: crea una agenda capace di gestire dim utenze */
    public AgendaTelefonica (int dim)

    /* inserisce nell'agenda una nuova utenza con i dati specificati; se l'agenda
       è piena non inserisce l'utenza e restituisce false, altrimenti
       restituisce true. */
    public boolean inserisci (String n, String c, String numero)

    /* restituisce il numero di telefono dell'utenza che ha il nome e cognome
       specificati se l'utenza è presente nell'agenda, e null altrimenti */
    public String trova (String nome, String cognome)

    /* visualizza sullo standard output la lista completa delle utenze
       nell'agenda */
    public void listaUtenze ()
}
```

Scrivere il corpo dei metodi della classe `AgendaTelefonica`. Scrivere inoltre una classe che:

- Permette all'utente di creare una agenda di dimensione a sua scelta.
- Offre all'utente un menù dal quale sia possibile scegliere ripetutamente una tra le seguenti opzioni:
 1. inserimento di una nuova utenza nell'agenda
 2. ricerca di una nuova utenza nell'agenda
 3. visualizzazione di tutte le utenze dell'agenda sullo standard output
 4. uscita dal menù

Provare infine ad aggiungere alla classe AgendaTelefonica un metodo che permetta di cancellare una utenza con nome e cognome specificati, ed aggiungere l'operazione di rimozione al menù della classe di test.