

Esercitazione Corso di laurea

La segreteria di un corso di laurea desidera sviluppare un archivio elettronico per la gestione di tutti gli studenti, di tutti gli appelli d'esame, e delle sessioni di laurea.

Implementare le seguenti classi con i relativi metodi prescritti.

Attenzione: non esistono esami diversi con lo stesso nome. Potete definire dei metodi ausiliari non indicati, se lo ritenete opportuno.

Classe Esame

```
public class Esame {
    private String nome;
    private int crediti;
    private int voto;
    public Esame(String unNomeEsame, int numCrediti){...}
    public String getNome(){...}
    public String getCrediti(){...}
    public String getVoto(){...}
    public void setVoto(int unVoto){...}
    public String toString(){...}
}
```

Classe Studente

```
public class Studente {
    private String nome;
    private int matricola;
    private ArrayList<Esame> esamiSostenuti;
    private ArrayList<Esame> pianoStudio;
    public Studente(String unNomeStudente, int unaMatricola){...}
    public String getNome(){...}
    public String getMatricola(){...}
    public void aggiungiEsame(String unNomeEsame, int numCrediti){...}
    public void promosso(String unNomeEsame, int unVoto){...}
    public void cancella(String unNomeEsame){...}
    public double media(){...}
    public double crediti(){...}
    public String toString(){...}
}
```

La classe contiene:

- due ArrayList di oggetti di tipo **Esame**:

- **PianoStudi** contenente il **piano di studi dello studente**, cioè il numero di esami che uno studente deve sostenere per conseguire la laurea,
- **EsamiSostenuti**, cioè il numero di esami il cui voto è superiore a 18.
- Si creano dei **metodi ausiliari**
 - `cercaIndiceEsamePiano(String unNomeEsame)`
 - `cercaIndiceEsamiSostenuti(String unNomeEsame)`,
 - entrambi che verificano la presenza di un dato esame nelle ArrayList
- metodo `promosso(String unNomeEsame, int unVoto)`: se l'esito della ricerca di un dato esame è stato positivo,
- metodo `aggiungiEsame(String unNomeEsame)`, aggiunge un nuovo oggetto nell'ArrayList **EsamiSostenuti**, fissando un voto, ed eliminando il corrispondente dall'ArrayList **PianoStudi**.
- metodo `cancella(String unNomeEsame)`

Classe Archivio:

```
public class CorsoLaurea {
    private String nome;
    private ArrayList<Studente> immatricolati;
    public static int prossimaMatricola;
    public static final int MIN_CREDITI;
    public CorsoLaurea(String unNomeStudente){...}
    public String getNome(){...}
    public void iscrivi(String unNomeStudente){...}
    public void ritira(String unNomeStudente){...}
    public void appello(String unNomeEsame, ArrayList<Studente> iscritti)
{...}
    public void laurea(){...}
}
```

- è presente una **variabile di classe (static) prossimaMatricola**, che assegna un numero ad ogni studente, e che essa deve essere incrementata o decrementata in base alle necessità (aggiungi o rimuovi un nuovo studente)
- nel metodo `appello(String unNomeEsame, ArrayList<Studente> iscritti)`, è necessario creare un oggetto di classe **java.util.Random**, che assegna un voto **casuale** ad ogni studente (un intero non superiore a 30).
- Gli studenti che totalizzano un punteggio superiore a 18, sono promossi, altrimenti dovranno sostenere nuovamente l'esame.

Test delle funzionalità

Implementare una classe **CorsoTester**, con metodo **main()** per testare le funzionalità dell'applicativo inserendo opportuni dati di prova (studenti, esami, ...).

Non occorre implementare l'interfaccia grafica o testuale.

Esempio esami CDL Informatica

I ANNO

SEMESTRE	CORSO	CFU
1°	Matematica discreta	9
1°	Fondamenti di informatica	6
1°	Programmazione 1	12
1°- 2°	Abilità linguistiche (Inglese)	3
2°	Calcolo differenziale e integrale	9
2°	Algoritmi e strutture dati 1	9
2°	Architettura degli elaboratori 1	6
2°	Fisica e metodo scientifico	6

II ANNO

SEMESTRE	CORSO	CFU
1°	Calcolo scientifico e metodi numerici	6
1°	Sistemi operativi 1	12
1°	Automi e linguaggi formali	6
1°	Elementi di economia e diritto per informatici	6
2°	Reti di calcolatori	9
2°	Programmazione 2	9
2°	Dati e Modelli	6
2°	Attività formative a scelta	6

III ANNO

SEMESTRE	CORSO	CFU
1°	Basi di dati	9
1°	Linguaggi di programmazione	9
1°	Interazione uomo-macchina	6
1°	Attività formative a scelta	6
2°	Ingegneria del software	9
2°	Tirocinio	15
2°	Prova finale	6