

## Esame di Programmazione II, 13 settembre 2024

(si consegna Corso.java, Esame.java, MainEsame.java, Studente.java, StudenteIllegaleException.java e StudenteLavoratore.java)

*Si crei un progetto Eclipse e il package `it.univr.corso`. Si copino al suo interno le classi del compito. Non si modifichino le dichiarazioni dei metodi e delle classi. Si possono definire altri campi, metodi o costruttori non richiesti dal compito, ma devono essere `private`. Si possono definire altre classi, che in tal caso vanno consegnate. La soluzione che verrà consegnata dovrà compilare, altrimenti non verrà corretta.*

**Esercizio 1 (3 punti).** Si completi la classe `Corso.java`, che rappresenta un corso di laurea (per esempio, *Informatica*), con un nome e una durata in anni.

**Esercizio 2 (2 punti).** Si crei la classe di eccezione controllata `StudenteIllegaleException.java`, che ha solo un costruttore, il quale riceve il messaggio di eccezione e lo passa alla superclasse.

**Esercizio 3 (8 punti).** Si completi la classe `Studente.java` che rappresenta uno studente, con un nome, cognome, matricola e anno di iscrizione. Non si dimentichi di implementare il lancio dell'eccezione dentro il costruttore.

**Esercizio 4 (4 punti).** Si completi la classe `StudenteLavoratore.java`, sottoclasse di `Studente.java`. Rispetto a uno studente *normale*, uno studente lavoratore impiega il doppio degli anni per finire fuori corso (per esempio, sei anni per un corso che abbia una durata di tre anni). Serve forse ridefinire qualche metodo della superclasse?

**Esercizio 5 (7 punti).** Si completi la classe `Esame.java`, che rappresenta un esame di un corso di laurea, a cui si possono iscrivere degli studenti (per esempio, *Programmazione Quantistica*). Non si dimentichi di implementare il lancio dell'eccezione dentro il metodo `iscrivi`.

**Esercizio 6 (7 punti).** Si completi la classe di prova `MainEsame.java`, nei tre punti indicati. Nel primo va letto uno studente da tastiera, nel secondo e terzo vanno selezionati solo alcuni degli studenti iscritti all'esame e poi ne fanno stampare le matricole (secondo punto) o l'intero studente (terzo punto).

**Suggerimento 1:** per conoscere l'anno in cui ci troviamo, si può usare `Year.now().getValue()`, dove `Year` è la classe di libreria `java.time.Year`.

**Suggerimento 2:** in questo compito non serve usare né lambda espressioni né stream, ma è ovviamente possibile farlo se ritenete che così si semplifichi il codice. In tal caso, è possibile concatenare le stringhe di uno stream di stringhe `s`, andando a capo tra una e l'altra stringa, scrivendo `s.collect(Collectors.joining("\n"))`, dove `Collectors` è la classe di libreria `java.util.stream.Collectors`.

---

ESEMPIO DI ESECUZIONE ALLA PAGINA SEGUENTE ⇒

Se tutto è corretto, una esecuzione di MainEsame.java potrebbe essere:

```
Nome: Alessandro  
Cognome: Manzoni  
Matricola: 123456  
Anno di immatricolazione: 2025
```

```
Studente illegale, riprova
```

```
Nome: Alessandro  
Cognome: Manzoni  
Matricola: 123456  
Anno di immatricolazione: 2024
```

```
Esame di Programmazione Quantistica del corso di Informatica:  
34555 Giulio Andreotti immatricolato nel 2017  
98034 Giordano Bruni immatricolato nel 2018  
111564 Giulio Rossi immatricolato nel 2021  
123456 Alessandro Manzoni immatricolato nel 2024  
151535 Antonietta Reale immatricolato nel 2020  
178066 Alessandra Allegri immatricolato nel 2024
```

```
Matricole degli studenti fuori corso:  
34555  
151535
```

```
Studenti lavoratori:  
34555 Giulio Andreotti immatricolato nel 2017  
98034 Giordano Bruni immatricolato nel 2018
```

Si noti che l'inserimento di Alessandro Manzoni avviene da tastiera e che la prima volta tale inserimento viene rifiutato perché non è possibile creare studenti immatricolati nel 2025, visto che al momento siamo nel 2024.