

## Esame di Programmazione II, 18 febbraio 2019

*Si crei un progetto Eclipse e, nella directory dei sorgenti, si crei il package `it.univr.library`. Si copino al suo interno le classi del compito, tranne `Main.java` che va copiato dentro il package di default. Se si realizzano nuove classi, le si creino dentro il package `it.univr.library`. Non si modifichino le dichiarazioni dei metodi. Si possono definire altri campi, metodi, costruttori e classi, ma devono essere **private**. La consegna fornita compila a meno di due costruttori che non compilano. La soluzione che verrà consegnata dovrà compilare, altrimenti non verrà corretta.*

Lo scopo del compito è di realizzare un catalogo dei libri di una biblioteca. I libri possono essere cartacei (**PaperBook**) oppure audio (**AudioBook**). Un catalogo contiene vari libri, senza ripetizioni, e permette di specificare l'ordinamento con cui vanno tenuti dentro al catalogo stesso. I generi dei libri (storia, informatica, narrativa...) sono enumerati dentro **Genre**, che è già completa e non deve venire modificata.

**Esercizio 1 [4 punti]** La classe astratta **Book** rappresenta un libro, genericamente, senza ancora distinguere fra libri cartacei e audio. La si completi dove indicato con **TODO**.

**Esercizio 2 [6 punti]** Le sottoclassi concrete **PaperBook** e **AudioBook** di **Book** implementano i due diversi tipi di libro. Le si completino dove indicato con **TODO**.

**Esercizio 3 [7 punti]** La classe **Catalog** implementa il catalogo dei libri di una biblioteca. Quindi la sua costruzione richiede di specificare i libri che devono essere contenuti dentro il catalogo. Tali libri sono tenuti ordinati. Questo significa che stampando un catalogo si vedranno i libri in ordine crescente e che iterando su un catalogo si otterranno i libri in ordine crescente. L'ordine dei libri non è necessariamente quello del `compareTo` fra i libri: è possibile infatti specificare un ordinamento diverso al momento della costruzione del catalogo, passando un oggetto di tipo `java.util.Comparator`. Si completi la classe dove indicato con **TODO**.

**Esercizio 4 [5 punti]** La classe **CatalogWithStatistics** si comporta come la sua superclasse **Catalog** ma ha un `toString` che aggiunge, dopo la lista dei libri, una riga che riporta il numero totale di pagine dei libri cartacei e il numero totale di minuti degli audio-libri. Si completi la classe dove indicato con **TODO**.

**Esercizio 5 [9 punti]** Si completi la classe di partenza **Main**, dove indicato con **TODO**. In particolare, si dovrà creare tre cataloghi e stamparli. Tutti e tre i cataloghi contengono gli stessi libri, ma usano tre ordinamenti distinti. Inoltre il primo riporta anche il numero di pagine e minuti totali dei libri.

*Se tutto è corretto, l'esecuzione del **Main** (già completo, non lo si modifichi) stamperà quanto riportato nel file di testo `risultato_main.txt`.*