

## ESERCIZIO SULL'EREDITARIETÀ

Si vuole modellare un negozio di computer e stampanti. Ogni prodotto venduto dal negozio è rappresentato da un oggetto della classe *Item*, contenente le seguenti informazioni:

- a) produttore (stringa)
- b) modello (stringa)
- c) prezzo (double)
- d) pezzi disponibili (intero)

Di una stampante (che è un item specializzato) interessano le seguenti informazioni:

- a) pagine al minuto (intero)
- b) stampa a colori (booleano)
- c) presenza scanner (booleano)
- d) tipo di stampa (una stringa tra “Toner” e “IJ”)

Di un computer (anch'esso un item specializzato) interessa:

- a) dimensione schermo in pollici (double)
- b) capacità dell'hardisk SSD (intero)
- c) capacità hardisk standard (intero)
- d) capacità RAM (intero)
- e) scheda video (stringa)

Ogni item ha due funzioni:

- una funzione per stampare tutti i dati (sia quelli generali, sia quelli specifici che dipendono dalla tipologia dell'item)
- una funzione che restituisce il tipo dell'item (“Computer” o “Stampante”)

### Parte I

Creare le classi *Item*, *Stampante*, *Computer*.

### Parte II

Creare una classe *NegozioInformatica*, con i seguenti campi:

- 1) nome (stringa)
- 2) indirizzo (stringa)
- 3) città (stringa)
- 4) lista item (array di Item)

La classe espone i seguenti metodi:

- *public NegozioInformatica(String nome, String indirizzo, String città, int numItemIniziali)*, che crea un nuovo oggetto della classe con le informazioni fornite, inizializzando la lista degli item in modo da avere una lunghezza iniziale pari a *numItemIniziali*
- *public int numItem()*, che restituisce il numero di item attualmente disponibili in negozio, sommando tutte le disponibilità di tutti i prodotti
- *public void aggiungiItem(Item item)* che aggiunge un nuovo item alla lista. In modo particolare, se esiste già uno stesso item con stesso produttore e modello, si aggiorna il numero di pezzi disponibili aggiungendo quelli indicati in *item*. In caso contrario, va aggiunto un nuovo item, eventualmente allungando la lista se piena.
- *public void vendiItem(String produttore, String modello, int quantita)*, che cerca nella lista un item con le caratteristiche indicate e ne decrementa il numero di pezzi disponibili di *quantita*. Se il nuovo numero di pezzi disponibile è uguale a zero, l'item viene eliminato dalla lista, scalando di un posto tutti gli elementi che lo seguivano, in modo da non creare buchi all'interno della lista.  
Nel caso in cui non si trovi nessun item nella lista o il numero di pezzi disponibile è inferiore a *quantita*, si lancia un'eccezione custom.
- *public void stampaStatistiche()*, che stampa il numero di diversi modelli di stampante e computer presenti nella lista di item

### **Parte III**

Definire una classe *ServiziNegozzi*, che implementi il seguente metodo:

- *public static void stampaFattura(Item[] listaItem)*, che stampa su ogni riga nome, modello, quantità e prezzo di ogni item nella lista fornita, mostrando sull'ultima riga anche il totale della spesa.

### **Parte IV**

Scrivere una classe di test che permetta di verificare tutte le funzionalità realizzate ai punti II e III.